Saul I. Gass
Michael C. Fu
Editors

Encyclopedia of Operations Research and Management Science

Third Edition

Springer Reference

Encyclopedia of Operations Research and Management Science

Saul I. Gass • Michael C. Fu
Editors

Encyclopedia of Operations Research and Management Science

Third Edition

A–I

With 231 Figures and 59 Tables

Springer Reference

Editors
Saul I. Gass
Robert H. Smith School of Business
University of Maryland
College Park, MD, USA

Michael C. Fu
Robert H. Smith School of Business and Institute for Systems Research
University of Maryland
College Park, MD, USA

ISBN 978-1-4419-1137-7 ISBN 978-1-4419-1153-7 (eBook)
ISBN 978-1-4419-1154-4 (print and electronic bundle)
DOI 10.1007/978-1-4419-1153-7
Springer New York Heidelberg Dordrecht London

Library of Congress Control Number: 2013949762

Dedication: In Memoriam

To the memory of my very dear colleague Saul Irving Gass, an OR pioneer, practitioner, and statesman, and a true scholar and friend. Saul passed away on March 17, 2013, as this edition of the Encyclopedia *was going to press.*

Preface

The goal of the *Encyclopedia of Operations Research and Management Science* is to provide decision makers and problem solvers in business, industry, government, and academia a comprehensive overview of the wide range of ideas, methodologies, and synergistic forces that combine to form the preeminent decision-aiding fields of operations research and management science (OR/MS). The impact of OR/MS on people's quality of life and economic well-being is a story that deserves to be told in its full detail and glory. The *Encyclopedia of Operations Research and Management Science* is the prologue to that story.

The editors, working with the *Encyclopedia*'s Editorial Advisory Board, surveyed and divided OR/MS into specific topics that collectively encompass the foundations, applications, and emerging elements of this ever-changing field. We also wanted to establish the close associations that OR/MS has maintained with other scientific endeavors, with special emphasis on its symbiotic relationships with computer science, information systems, and mathematics. Based on our broad view of OR/MS, we enlisted a distinguished international group of academics and practitioners to contribute entries on subjects for which they are renowned. We commissioned over 200 major expository entries and complemented them by numerous descriptions, discussions, definitions, and abbreviations. The connections between topics are highlighted by an entry's final "See" statement, as appropriate. Each entry provides a background or history of the topic, describes relevant applications, overviews present and future trends, and lists seminal and current references. To allow for variety in exposition, the authors were instructed to present their material from their research and applied perspectives. In particular, the authors, each of whom is a leading authority on the particular subject, were allowed to use whatever mathematical notation they felt was standard for their topics.

The *Encyclopedia*'s intended audience is technically diverse and wide; it includes anyone concerned with the science, techniques, and ideas of how one makes decisions. As this audience encompasses many professions, educational background, and skills, we were attentive to the form, format, and scope of the entries. Thus, the entries are designed to serve as initial sources of information for all such readers, with special emphasis on the needs of students.

What are Operations Research and Management Science?

Operations research and management science are often equated to one another. If one defines them by the methodologies they employ, the equation would probably stand

inspection. If, however, one defines them by their historical developments and the classes of problems they encompass, the equation becomes fuzzy. OR grew out of the operational problems of the British and U.S. military efforts in World War II. It was augmented methodologically and computationally by the postwar developments of linear programming, game theory, dynamic programming, discrete-event simulation (among others), and the digital computer. A number of additional ideas and problem types from the pre-war years were incorporated into the field as well, including inventory and queueing theories, Markov modeling, and the basic methods of optimization. Early (1950s) practitioners of OR applied its philosophy and techniques to the solution of industrial and business operational problems with great success. It was soon recognized that whatever OR was as a scientific field, it could be used to study and solve the broader planning and strategic issues of organizational management, financial planning, and public policy. From this observation, MS began and flourished in a similar and somewhat overlapping manner to OR.

More formal definitions of OR and MS are readily available. OR can be defined as: (1) the application of the methods of science to complex problems arising in the direction and management of large systems of men, machine, materials, and money in industry, business, government, and defense; (2) the science of deciding how to best design and operate man-machine systems; (3) a scientific method for providing executive departments with a quantitative basis for decision making. MS can be defined as: (1) the application of scientific methodology or principles to management decisions; (2) the use of quantitative methods for solving management and organizational decision problems. Together, OR and MS may be thought of as the science of operational processes, decision making, and management. However, to our minds, the definition of OR/MS is really given by the coverage of the material in this *Encyclopedia*.

Second Edition

The second edition aimed to capture the advances since the 1996 first edition, especially the relationships between OR/MS and information technology, and to update, expand, and correct the original material. With respect to OR/MS advances, new material ranged from the Analytic Network Process to Data Mining to Electronic Commerce to the Theory of Constraints, among the 28 new entries added at that time. First edition entries were updated or rewritten by the original and/or new authors. Based on suggestions from the readers of the first edition, more material on the history of OR/MS was added, including emphasis of its origins by an article on early British OR.

Third Edition

This third edition of the *Encyclopedia of Operations Research and Management Science* moves us closer to our goal of providing a comprehensive overview of the theoretical and applied subject matter that forms the ever-expanding field of OR/MS. Again, many of the second edition entries have been updated by the original authors,

with a few being totally rewritten by new authors. In addition, the following completely new entries have been added, each of which either describes a new topic or replaces a previous short entry:

- Agent-Based Simulation
- Air Traffic Management
- Approximate Dynamic Programming
- Business Intelligence
- Closed-loop Supply Chains
- Combinatorial Auction Theory
- Community OR
- Complementarity Applications
- Computational Biology
- Conditional Value at Risk
- Convex Optimization
- Critical Systems Thinking
- Data Warehousing
- Decision Analysis Practice
- Deep Uncertainty
- Differential Games
- Digital Music
- Disaster Management
- Disease Prevention, Detection, and Treatment
- Financial Engineering
- Flexible Manufacturing Systems
- Fuzzy Sets, Systems, and Applications
- Global Optimization
- Health Care Management
- Health Care Strategic Decision Making
- Heuristics
- Hit and Run Methods
- Influence Diagrams
- Knowledge Management
- Lagrangian Relaxation
- Markov Chain Monte Carlo
- Metaheuristics
- Open Source Software (and COIN-OR)
- Operational Research Society
- Petroleum Refining
- Quadratic Assignment Problem
- Rare Event Simulation
- Regenerative Simulation
- Response Surface Methodology
- Revenue Management
- Sample Average Approximation
- Sensitivity Analysis
- Service Science
- Simulated Annealing
- Societal Complexity
- Statistical Ranking and Selection
- Stochastic Approximation
- Stochastic Input Model Selection

We want to emphasize that the *Encyclopedia of Operations Research and Management Science* is the responsibility of the editors. We made the final determination of the scope, topics, and material. Any shortcoming (editorial, inclusion, omission, emphasis, factual) that the reader may perceive rests with us. Hence, we sincerely welcome comments and feedback on all aspects of the *Encyclopedia*.

Acknowledgments

We wish to thank all the contributors to the *Encyclopedia of Operations Research and Management Science* for their individual efforts and for their cooperation, support, and patience. We would also like to thank the members of the Editorial Advisory Board, past and present, for their help in the initial formulation of the *Encyclopedia* and their subsequent efforts in reviewing the articles. Last but not least, we are grateful to Springer for their help throughout the process of preparing this third edition, and would specifically like to thank the editorial support provided by Jennifer Carlson, Elizabeth Ferrell, Julia Koerting, and Annalea Manalili.

About the Editors

Saul I. Gass received his B.S. in Education and M.A. in Mathematics from Boston University in 1949, and his Ph.D. in Engineering Science/Operations Research from the University of California, Berkeley, in 1965. At the time of his passing, he was Professor Emeritus at the Robert H. Smith School of Business, University of Maryland, College Park, after serving as the Dean's Lifetime Achievement Professor and having been named a University of Maryland Distinguished Scholar-Teacher. He served over 2 years in the Army during World War II prior to entering college. His first job in 1949 was as a mathematician for the Aberdeen Bombing Mission, U.S. Air Force, but shortly thereafter he transferred to Air Force Headquarters where he began his career in operations research when he joined Project SCOOP at the Pentagon-based Directorate of Management Analysis, the organization in which linear programming was first developed. He left Project SCOOP in 1955 to join IBM, where he served as an Applied Science Representative, Manager of the Project Mercury Man-in-Space Program, and Manager of IBM's Federal Civil Programs. He also served on the Science and Technology Task Force of the President's Commission on Law Enforcement. Other positions included Director of Operations Research for the Corporation for Economic and Industrial Research (CEIR), Senior Vice-President of World Systems Laboratories, and Vice-President of Mathematica. He has served as a consultant to the U.S. General Accounting Office, Congressional Budget Office, the National Institute of Standards and Technology, and other operations research and systems analysis organizations. Included in his many publications are the textbooks *Linear Programming* (the first textbook on the subject, now in its fifth edition) and *Decision Making, Models and Algorithms*, and the books *An Illustrated Guide to Linear Programming*, *An Annotated Timeline of Operations Research: An Informal History*, and *Profiles in Operations Research: Pioneers and Innovators*. He served as President of the Operations Research Society of America (ORSA) and Omega Rho and as Vice-President for international activities of the Institute of Operations Research and the Management Sciences (INFORMS) and the International Federation of Operational Research Societies (IFORS). He is a recipient of the ORSA Kimball Medal for distinguished service to the society and the profession, the INFORMS Expository Writing Award, and the Military Operations Research Society's Jacinto Steinhardt Memorial. He is a Fellow of INFORMS and was a Fulbright Research Scholar at the Computer and Automation Research Institution, Hungarian Academy of Sciences.

Michael C. Fu received his S.B. in Mathematics and S.M/S.B. in Electrical Engineering and Computer Science from MIT in 1985, and his Ph.D. in Applied Mathematics from Harvard University in 1989. Since 1989, he has been at the University of Maryland in the Robert H. Smith School of Business, where he is

currently Ralph J. Tyser Professor of Management Science, with a joint appointment in the Institute for Systems Research, A. James Clark School of Engineering. At the University of Maryland, he was named a Distinguished Scholar-Teacher and received the Institute for System Research's Outstanding Systems Engineering Faculty Award and the Business School's Allen J. Krowe Award for Teaching Excellence. He served as Simulation Area Editor of *Operations Research* from 2000 to 2005 and as the Stochastic Models and Simulation Department Editor of *Management Science* from 2006 to 2008, and on the Editorial Boards of the *INFORMS Journal on Computing*, *Mathematics of Operations Research*, *Production and Operations Management*, and *IIE Transactions*. He was Program Chair of the 2011 Winter Simulation Conference. His coauthored book, *Conditional Monte Carlo: Gradient Estimation and Optimization Applications*, received the INFORMS College on Simulation Outstanding Publication Award. He also coauthored the research monograph *Simulation-based Algorithms for Markov Decision Processes*, and coedited the books *Perspectives in Operations Research* and *Advances in Mathematical Finance*. He is a Fellow of IEEE and INFORMS.

Advisory Board Members

Contributors

Leonard Adelman George Mason University, Fairfax, VA, USA

Vijay K. Agrawal The University of Nebraska at Kearney, Kearney, NE, USA

Alp Akcay Carnegie Mellon University, Pittsburgh, PA, USA

Susan Albin Rutgers, The State University of New Jersey, Piscataway, NJ, USA

Abeer A. Al-Hassan Kuwait University, Kuwait City, Kuwait

Francis B. Alt University of Maryland, College Park, MD, USA

Robert L. Armacost University of Central Florida, Orlando, FL, USA

J. Scott Armstrong University of Pennsylvania, Philadelphia, PA, USA

Jay E. Aronson The University of Georgia, Athens, USA

Søren Asmussen Aarhus University, Aarhus, Denmark

Osman Balci Virginia Polytechnic Institute & State University, Blacksburg, VA, USA

Michael Ball University of Maryland, College Park, MD, USA

Stephen J. Balut Institute for Defense Analyses, Alexandria, VA, USA

Steve Bankes BAE Systems, Arlington, VA, USA

Arnold Barnett Massachusetts Institute of Technology, Cambridge, MA, USA

David J. Bartholomew The London School of Economics and Political Science, London, UK

Russell R. Barton The Pennsylvania State University, University Park, PA, USA

Frank M. Bass The University of Texas at Dallas, Richardson, TX, USA

Rajan Batta University at Buffalo, The State University of New York, Buffalo, NY, USA

Paul van Beek Wageningen University, Wageningen, The Netherlands

Isabel M. Beichl National Institute of Standards & Technology, Gaithersburg, MD, USA

Peter C. Bell University of Western Ontario, London, Ontario, Canada

Filmore Bender University of Maryland, College Park, MD, USA

Javier Bernal National Institute of Standards & Technology, Gaithersburg, MD, USA

Bahar Biller Carnegie Mellon University, Pittsburgh, PA, USA

John R. Birge The University of Chicago, Chicago, IL, USA

Gabriel R. Bitran Massachusetts Institute of Technology, Cambridge, MA, USA

John L. G. Board Henley Business School, University of Reading, Reading, UK

Lawrence Bodin University of Maryland, College Park, MD, USA

Paul T. Boggs Sandia National Laboratories, Livermore, CA, USA

National Institute of Standards and Technology, Gaithersburg, MD, USA

Eric T. Bradlow University of Pennsylvania, Philadelphia, PA, USA

Kurt M. Bretthauer Texas A&M University, College Station, TX, USA

Indiana University Bloomington, Bloomington, IN, USA

Percy H. Brill University of Windsor, Windsor, Ontario, Canada

Robert G. Brown Materials Management Systems, Thetford Center, VT, USA

James R. Buck The University of Iowa, Iowa City, IA, USA

Dennis M. Buede Innovative Decisions, Inc., Vienna, VA, USA

George Mason University, Fairfax, VA, USA

Richard M. Burton Duke University, Durham, NC, USA

Kathleen M. Carley Carnegie Mellon University, Pittsburgh, PA, USA

Michael W. Carter University of Toronto, Toronto, Ontario, Canada

Jonathan P. Caulkins Carnegie Mellon University, Pittsburgh, PA, USA

Kaushal Chari University of South Florida, Tampa, FL, USA

Peter Checkland Lancaster University, Lancaster, UK

Kenneth Chelst Wayne State University, Detroit, MI, USA

Elaine Chew Queen Mary, University of London, London, UK

Dilip Chhajed University of Illinois at Urbana-Champaign, Champaign, IL, USA

John W. Chinneck Carleton University, Ottawa, Ontario, Canada

Clyde G. Chittister Carnegie Mellon University, Pittsburgh, PA, USA

Michael D. D. Clarke Sabre Research, Southlake, TX, USA

Izack Cohen Technion – Israel Institute of Technology, Haifa, Israel

Nastaran Coleman Federal Aviation Administration, Washington, DC, USA

Sue A. Conger University of Dallas, Irving, TX, USA

William W. Cooper The University of Texas at Austin, Austin, TX, USA

Richard W. Cottle Stanford University, Stanford, CA, USA

Peter I. Cowling University of Bradford, Bradford, UK

Thomas K. Dasaklis University of Piraeus, Piraeus, Greece

Sriram Dasu University of Southern California, Los Angeles, CA, USA

Brian T. Denton University of Michigan, Ann Arbor, MI, USA

Dorien J. DeTombe International Research Society on Methodology of Societal Complexity, Amsterdam, The Netherlands

James A. Dewar RAND Corporation, Santa Monica, CA, USA

Roberto Diéguez Galvão Federal University of Rio de Janeiro, Brazil

David Dobrzykowski Eastern Michigan University, Ypsilanti, MI, USA

Paul Dupuis Brown University, Providence, RI, USA

James S. Dyer The University of Texas at Austin, Austin, TX, USA

Joseph G. Ecker Rensselaer Polytechnic Institute, Troy, NY, USA

Jonathan Eckstein Rutgers, The State University of New Jersey, Livingston Campus, New Burnswick, NJ, USA

Richard W. Eglese Lancaster University, Lancaster, UK

Jehoshua Eliashberg University of Pennsylvania, Philadelphia, PA, USA

Joseph H. Engel Bethesda, MD, USA

Yariv Ephraim George Mason University, Fairfax, VA, USA

Gary M. Erickson University of Washington, Seattle, WA, USA

Stuart Eriksen Santa Ana, CA, USA

Gerald W. Evans University of Louisville, Louisville, KY, USA

Anthony V. Fiacco The George Washington University, Washington, DC, USA

Peter Fishburn AT&T Bell Laboratories, Murray Hill, NJ, USA

Gene H. Fisher RAND Corporation, Santa Monica, CA, USA

Leonard Fortuin Eindhoven University of Technology, Eindhoven, The Netherlands

Robert Fourer Northwestern University, Evanston, IL, USA

Richard L. Francis University of Florida, Gainesville, FL, USA

Terrill L. Frantz Peking University, Shenzhen, Guangdong, China

Hershey H. Friedman City University of New York, Brooklyn, NY, USA

Linda Weiser Friedman Baruch College, City University of New York, New York, NY, USA

John A. Friel RAND Corporation, Santa Monica, CA, USA

Michael C. Fu University of Maryland, College Park, MD, USA

Steven A. Gabriel University of Maryland, College Park, MD, USA

Tomas Gal Fern Universität in Hagen, Hagen, Germany

Gina M. Galindo Pacheco University at Buffalo, The State University of New York, Buffalo, NY, USA

Universidad del Norte, Barranquilla, Colombia

Mark A. Gallagher Air Force Studies and Analyses, Assessments, and Lessons Learned, Washington, DC, USA

Saul I. Gass University of Maryland, College Park, MD, USA

Denos C. Gazis PASHA Industries, Katonah, NY, USA

Arthur M. Geoffrion University of California, Los Angeles, CA, USA

Theodoros Gevezes Aristotle University of Thessaloniki, Thessaloniki, Greece

Fred W. Glover OptTek Systems, Inc., Boulder, CO, USA

University of Colorado Boulder, Boulder, CO, USA

Hans W. Gottinger International Institute for Technology Management and Economics, Bad Waldsee, Germany

Andreas Graefe LMU Munich, Munich, Germany

Paul Gray Claremont Graduate University, Claremont, CA, USA

Kesten C. Green University of South Australia, Adelaide, South Australia, Australia

Harvey J. Greenberg University of Colorado-Denver, Denver, CO, USA

Irwin Greenberg George Mason University, Fairfax, VA, USA

Scott D. Grimshaw Brigham Young University, Provo, UT, USA

Donald Gross George Mason University, Fairfax, VA, USA

Thomas A. Grossman University of San Francisco, San Francisco, CA, USA

Monique Guignard University of Pennsylvania, Philadelphia, PA, USA

Thomas R. Gulledge George Mason University, Fairfax, VA, USA

Jeffery L. Guyse University of California, Irvine, CA, USA

Peter J. Haas IBM Almaden Research Center, San Jose, CA, USA

Robert W. Haessler University of Michigan, Ann Arbor, MI, USA

Yacov Y. Haimes University of Virginia, Charlottesville, VA, USA

John R. Hall Jr. National Fire Protection Association, Quincy, MA, USA

Leslie Hall The Johns Hopkins University, Baltimore, MD, USA

Nicholas G. Hall The Ohio State University, Columbus, OH, USA

Dominique M. Hanssens University of California, Los Angeles, CA, USA

Carl M. Harris George Mason University, Fairfax, VA, USA

Ronald M. Harstad University of Missouri-Columbia, Columbia, MO, USA

Dean S. Hartley III Oak Ridge National Laboratory, Oak Ridge, TN, USA

Arnoldo C. Hax Massachusetts Institute of Technology, Cambridge, MA, USA

James C. Hearn University of Georgia, Athens, GA, USA

Willy S. Herroelen Katholieke Universiteit Leuven, Leuven, Belgium

Rebecca Herron University of Lincoln, Lincoln, UK

Sidney W. Hess Chadds Ford, Philadelphia, PA, USA

Daniel P. Heyman Lincroft, NJ, USA

Frederick S. Hillier Stanford University, Stanford, CA, USA

David S. Hirshfeld MathPro Inc., Bethesda, MD, USA

James K. Ho University of Illinois at Chicago, Chicago, IL, USA

Karla L. Hoffman George Mason University, Fairfax, VA, USA

Allen G. Holder Rose-Hulman Institute of Technology, Terre Haute, IN, USA

Clyde W. Holsapple University of Kentucky, Lexington, KY, USA

Thomas A. Horan Claremont Graduate University, Claremont, CA, USA

David W. Hutchison RAND Corporation, Santa Monica, CA, USA

Candice H. Huynh University of California, Irvine, CA, USA

Lakshmi S. Iyer The University of North Carolina at Greensboro, Greensboro, NC, USA

Richard H. F. Jackson National Institute of Standards and Technology, Gaithersburg, MD, USA

Jianmin Jia The Chinese University of Hong Kong, Shatin, NT, Hong Kong, China

M. Eric Johnson Dartmouth College, Hanover, NH, USA

Sharon A. Johnson Worcester Polytechnic Institute, Worcester, MA, USA

Albert Jones National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA

Kenneth De Jong George Mason University, Fairfax, VA, USA

Gerald Kahan McCormick and Company, Sparks, MA, USA

Bharat K. Kaku Georgetown University, Washington, DC, USA

Gurumurthy Kalyanaram The University of Texas at Dallas, Richardson, TX, USA

Alla Kammerdiner New Mexico State University, Las Cruces, NM, USA

Edward H. Kaplan Yale University, New Haven, CT, USA

Harry H. Kelejian University of Maryland, College Park, MD, USA

L. Robin Keller University of California, Irvine, CA, USA

Seong-Hee Kim Georgia Institute of Technology, Atlanta, GA, USA

Maurice W. Kirby Lancaster University, Lancaster, UK

Jack P. C. Kleijnen Tilburg University, Tilburg, The Netherlands

Howard W. Kreiner Center for Naval Analyses, Alexandria, VA, USA

Ramayya Krishnan Carnegie Mellon University, Pittsburgh, PA, USA

Dirk P. Kroese The University of Queensland, Brisbane, Australia

Roman Krzysztofowicz University of Virginia, Charlottesville, VA, USA

Jan H. Kwakkel Delft University of Technology, Delft, The Netherlands

Pierre L'Ecuyer Université de Montréal, Montréal, Québec, Canada

Shaul P. Ladany Ben-Gurion University of the Negev, Beer Sheva, Israel

Manuel Laguna University of Colorado Boulder, Boulder, CO, USA

Sarah S. Lam Binghamton University, Binghamton, NY, USA

Richard C. Larson Massachusetts Institute of Technology, Cambridge, MA, USA

Kathryn Blackmond Laskey George Mason University, Fairfax, VA, USA

Eugene L. Lawler

Robert J. Lempert RAND Corporation, Santa Monica, CA, USA

Peter Lenk University of Michigan, Ann Arbor, MI, USA

Benjamin Lev Drexel University, Philadelphia, PA, USA

Reuven R. Levary Saint Louis University, St. Louis, MO, USA

Matthew J. Liberatore Villanova University, Villanova, PA, USA

Gary L. Lilien The Pennsylvania State University, University Park, PA, USA

Andrew G. Loerch Center for Army Analysis, Fort Belvoir, VA, USA

Timothy J. Lowe University of Iowa, Iowa City, IA, USA

William F. Lucas Claremont Graduate University, Claremont, CA, USA

Roice D. Luke Virginia Commonwealth University, Richmond, VA, USA

Irvin Lustig IBM, Somers, NY, USA

Charles M. Macal Argonne National Laboratory, Argonne, IL, USA

Michael Magazine University of Cincinnati, Cincinnati, OH, USA

Costis Maglaras Columbia University, New York, NY, USA

Thomas L. Magnanti Massachusetts Institute of Technology, Cambridge, MA, USA

Nicolas S. Majluf Pontificia Universidad Católica de Chile, Santiago, Chile

Michael D. Maltz University of Illinois at Chicago, Chicago, IL, USA

Andre Z. Manitius George Mason University, Fairfax, VA, USA

William G. Marchal The University of Toledo, Toledo, OH, USA

Vladimir Marianov Pontificia Universidad Católica de Chile, Santiago, Chile

Rafael Martí University of Valencia, Valencia, Spain

Carl D. Martland Massachusetts Institute of Technology, Cambridge, MA, USA

Jennifer E. Mason University of Virginia, Charlottesville, VA, USA

Richard O. Mason Southern Methodist University, Dallas, TX, USA

Christina M. Mastrangelo University of Virginia, Charlottesville, VA, USA

James E. Matheson SmartOrg, Inc., Menlo Park, CA, USA

Brian R. McEnany Military Operations Research Society (MORS), Alexandria, VA, USA

Vijay Mehrotra University of San Francisco, San Francisco, CA, USA

Syam Menon The University of Texas at Dallas, Richardson, TX, USA

Zbigniew Michalewicz The University of Adelaide, Adelaide, South Australia, Australia

Douglas R. Miller George Mason University, Fairfax, VA, USA

Hugh J. Miser Farmington, CT, USA

Heiner Müller-Merbach Technische Universität Kaiserslautern, Kaiserslautern, Germany

Mansooreh Mollaghasemi University of Central Florida, Orlando, FL, USA

Douglas C. Montgomery Arizona State University, Tempe, AZ, USA

James W. Morrison University of Notre Dame, Notre Dame, IN, USA

Frederic H. Murphy Temple University, Philadelphia, PA, USA

Wendy M. Murphy IBM Corporation, Armonk, NY, USA

Katta G. Murty University of Michigan, Ann Arbor, MI, USA

Kaing Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

Steven Nahmias Santa Clara University, Santa Clara, CA, USA

Stephen G. Nash George Mason University, Fairfax, VA, USA

Yurii Nesterov Université Catholique de Louvain (UCL), Louvain-la-Neuve, Belgium

Marcel F. Neuts The University of Arizona, Tucson, AZ, USA

Michael J. North Argonne National Laboratory, Argonne, IL, USA

Børge Obel Aarhus University, Aarhus, Denmark

Yasar A. Ozcan Virginia Commonwealth University, Richmond, VA, USA

Manfred Padberg New York University, New York, NY, USA

John C. Papageorgiou Wellesley, MA, USA

Costas P. Pappis University of Piraeus, Piraeus, Greece

Panos M. Pardalos University of Florida, Gainesville, FL, USA

Eduardo Pasiliao Air Force Research Laboratory (AFRL) Munitions Directorate, Eglin Air Force Base, FL, USA

Chad Perry Queensland University of Technology, Brisbane, Australia

Julia Pet-Armacost University of Central Florida, Orlando, FL, USA

James B. Pick University of Redlands, Redlands, CA, USA

William P. Pierskalla University of California, Los Angeles, CA, USA

Leonidas Pitsoulis Aristotle University of Thessaloniki, Thessaloniki, Greece

Donald R. Plane Rollins College, Winter Park, FL, USA

Roman A. Polyak George Mason University, Fairfax, VA, USA

Sergey Porotsky A.L.D. Ltd., Tel-Aviv, Israel

Ingmar R. Prucha University of Maryland, College Park, MD, USA

Jean-Francois Puget IBM, Valbonne, France

David F. Pyke University of San Diego, San Diego, CA, USA

Luis C. Rabelo Universito of Central Florida, Orlando, FL, USA

Cliff T. Ragsdale Virginia Polytechnic Institute and State University, Blacksburg, VA, USA

Jagmohan Raju University of Pennsylvania, Philadelphia, PA, USA

Ted K. Ralphs Lehigh University, Bethlehem, PA, USA

John S. Ramberg Pagosa Springs, CO, USA

Ramaswamy Ramesh University at Buffalo, The State University of New York, Buffalo, NY, USA

Graham K. Rand Lancaster University, Lancaster, UK

Arvind Rangaswamy The Pennsylvania State University, University Park, PA, USA

Steffen Rebennack Colorado School of Mines, Golden, CO, USA

Andres Redchuk Universidad Rey Juan Carlos, Mostoles, Madrid, Spain

Universidad Autónoma de Chile, Santiago, Chile

Arnold Reisman Reisman and Associates, Shaker Heights, OH, USA

Charles ReVelle The Johns Hopkins University, Baltimore, MD, USA

Michael Rich RAND Corporation, Santa Monica, CA, USA

George P. Richardson University at Albany, State University of New York, Albany, NY, USA

Ad A. N. Ridder Vrije University, Amsterdam, The Netherlands

Giovanni Rinaldi CNR – Istituto di Analisi dei Sistemi ed Informatica (IASI), Rome, Italy

David Ríos Insua Spanish Royal Academy of Sciences, Madrid, Spain

Meir J. Rosenblatt Washington University in St. Louis, St. Louis, MO, USA

Technion – Israel Institute of Technology, Haifa, Israel

Jonathan Rosenhead The London School of Economics and Political Science, London, UK

Reuven Y. Rubinstein Technion – Israel Institute of Technology, Haifa, Israel

David M. Ryan The University of Auckland, Auckland, New Zealand

Thomas L. Saaty University of Pittsburgh, Pittsburgh, PA, USA

Andrew P. Sage George Mason University, Fairfax, VA, USA

Matthew J. Saltzman Clemson University, Clemson, SC, USA

The COIN–OR Foundation, Inc., Towson, MD, USA

Douglas A. Samuelson Infologix, Inc., Annandale, VA, USA

Hiroaki Sandoh Osaka University, Toyonaka, Osaka, Japan

Rakesh K. Sarin University of California, Los Angeles, CA, USA

Siegfried Schaible University of California, Riverside, CA, USA

Marc J. Schniederjans University of Nebraska-Lincoln, Lincoln, NE, USA

Marc Schoenauer INRIA Saclay – Île-de-France, Orsay cedex, France

David A. Schum George Mason University, Fairfax, VA, USA

William Schwabe RAND Corporation, Santa Monica, CA, USA

Suvrajeet Sen The University of Arizona, Tucson, AZ, USA

University of Southern California, Los Angeles, CA, USA

Richard F. Serfozo Georgia Institute of Technology, Atlanta, GA, USA

Gaia Serraino American Optimal Decisions, Gainesville, FL, USA

L. D. Servi The MITRE Corporation, Bedford, MA, USA

Alexander Shapiro Georgia Institute of Technology, Atlanta, GA, USA

Ramesh Sharda Oklahoma State University, Stillwater, OK, USA

Robert S. Sheldon Military Operations Research Society (MORS), Alexandria, VA, USA

Bala Shetty Texas A&M University, College Station, TX, USA

Douglas R. Shier Clemson University, Clemson, SC, USA

Edgar H. Sibley George Mason University, Fairfax, VA, USA

Sauleh A. Siddiqui University of Maryland, College Park, MD, USA

Constantinos I. Siettos National Technical University of Athens, Athens, Greece

Edward A. Silver University of Calgary, Calgary, Alberta, Canada

James R. Simpson Florida State University, Tallahassee, FL, USA

Jaya Singhal University of Baltimore, Baltimore, MD, USA

Kalyan Singhal University of Baltimore, Baltimore, MD, USA

Alice E. Smith Auburn University, Auburn, AL, USA

Robert L. Smith University of Michigan, Ann Arbor, MI, USA

Ariela Sofer George Mason University, Fairfax, VA, USA

Marius M. Solomon Northeastern University, Boston, MA, USA

Kenneth Sörensen University of Antwerp, Antwerp, Belgium

Gilvan C. Souza Indiana University Bloomington, Bloomington, IN, USA

James C. Spall The Johns Hopkins University, Applied Physics Laboratory, Laurel, MD, USA

James C. Spohrer Almaden Research Center, San Jose, CA, USA

Kathryn E. Stecke The University of Texas at Dallas, Richardson, TX, USA

Ralph E. Steuer University of Georgia, Athens, GA, USA

William R. Stewart Jr. College of William and Mary, Williamsburg, VA, USA

William J. Stewart North Carolina State University, Raleigh, NC, USA

Lawrence D. Stone Metron Inc., Reston, VA, USA

Todd Strauss Yale University, New Haven, CT, USA

Francis Sullivan Supercomputing Research Center, Bowie, MD, USA

Balram Suman Energy Technology Company, Chevron Corporation, Houston, TX, USA

Charles M. S. Sutcliffe University of Reading, Reading, UK

Edward A. Sykes Make Systems, Inc., Carey, NC, USA

Thomas Taimre The University of Queensland, Brisbane, Australia

Tamás Terlaky Lehigh University, Bethlehem, PA, USA

Clayton J. Thomas Air Force Studies and Analyses Agency (AFSAA), Washington, DC, USA

Kaoru Tone National Graduate Institute for Policy Studies, Minato-ku, Tokyo, Japan

Alan Tucker The State University of New York at Stony Brook, Stony Brook, NY, USA

Hoang Tuy Vietnam Academy of Science and Technology, Hanoi, Vietnam

Werner Ulrich University of Fribourg, Fribourg, Switzerland

The Open University, Milton Keynes, UK

Stanislav Uryasev University of Florida, Gainesville, FL, USA

Igor Ushakov Qualcomm Inc., San Diego, CA, USA

Andrew Vazsonyi University of San Francisco, San Francisco, CA, USA

Eugene P. Visco Silver Spring, MD, USA

Mark A. Vonderembse The University of Toledo, Toledo, OH, USA

Warren E. Walker Delft University of Technology, Delft, The Netherlands

William A. Wallace Rensselaer Polytechnic Institute, Troy, NY, USA

Hui Wang Brown University, Providence, RI, USA

Pearl Wang George Mason University, Fairfax, VA, USA

Luk Van Wassenhove INSEAD, Fontainebleau, France

John M. Watts Jr. Fire Safety Institute, Middlebury, VT, USA

Michel Wedel University of Maryland, College Park, MD, USA

Andrés Weintraub University of Chile, Santiago, Chile

Mark Westcombe Lancaster University, Lancaster, UK

Andrew B. Whinston The University of Texas at Austin, Austin, TX, USA

Chelsea C. White III Georgia Institute of Technology, Atlanta, GA, USA

Berend Wierenga Erasmus University Rotterdam, Rotterdam, The Netherlands

Yoram (Jerry) Wind University of Pennsylvania, Philadelphia, PA, USA

Christoph Witzgall National Institute of Standards & Technology, Gaithersburg, MD, USA

Norman Keith Womer University of Missouri-St Louis, St. Louis, MO, USA

Carlos G. Wong-Martinez Woosong University, Daejeon, Korea

R. E. D. Woolsey Colorado School of Mines, Golden, CO, USA

Xiaomei Xu Cleveland, OH, USA

Oliver S. Yu Star Strategy Group, Los Altos Hills, CA, USA

Zelda B. Zabinsky University of Washington, Seattle, WA, USA

Fatemeh Mariam Zahedi University of Wisconsin-Milwaukee, Milwaukee, WI, USA

Stavros A. Zenios University of Cyprus, Nicosia, Cyprus

University of Pennsylvania, Pennsylvania, PA, USA

Jingyu Zhang Philips Research North America, Briarcliff Manor, NY, USA

William T. Ziemba University of British Columbia, Vancouver, British Columbia, Canada

Oxford University, Oxford, UK

Stanley Zionts University at Buffalo, The State University of New York, Buffalo, NY, USA

A

A* Algorithm

A heuristic search procedure that selects a node in its search tree for expansion such that the selected node has minimum value of the sum of the cost to reach the node plus a heuristic cost value for that node, where the heuristic cost underestimates the true minimum cost of completion.

See

► Artificial Intelligence

References

Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics, 4*, 100–107.

Pearl, J. (1984). *Heuristics*. Reading, MA: Addison-Wesley.

Acceptance Sampling

► Quality Control

Acceptance-Rejection Method

In stochastic or Monte Carlo simulation, a method for sampling from a given difficult target probability distribution by sampling from a distribution that is close to the target distribution and relatively easy to sample but possibly rejecting the generated output. Sometimes just called the rejection method.

See

► Monte Carlo Methods
► Monte Carlo Simulation
► Simulation of Stochastic Discrete-Event Systems

Accounting Prices

► Shadow Prices

Accreditation

► Model Accreditation

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Active Constraint

A constraint in an optimization problem that is satisfied exactly by a solution.

See

- ▶ Inactive Constraint
- ▶ Slack Variable
- ▶ Surplus Variable

Active Set Methods

- ▶ Quadratic Programming

Activity

(1) A structural variable whose value (level) is to be computed in a linear programming problem.
(2) Project work items having specific beginning and completion points and durations.

See

- ▶ Network Planning
- ▶ Project Management
- ▶ Structural Variables

Activity Level

The value taken by a structural variable in an intermediate or final solution to a linear programming problem.

See

- ▶ Structural Variables

Activity-Analysis Problem

A linear-programming problem of the form Maximize $\boldsymbol{cx}$, subject to $\boldsymbol{Ax} \leq \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$. The variables x_j of the vector $\boldsymbol{x}$ are quantities of products to be produced. The b_i coefficients of the resource vector $\boldsymbol{b}$ represent the amount of resource i that is available for production, the c_j coefficients represent the value (profit) of one unit of output x_j, and the coefficients a_{ij} of the technological matrix $\boldsymbol{A}$ represent the amount of resource i required to produce one unit of product j. The a_{ij} are termed technological or input-output coefficients. The objective function $\boldsymbol{cx}$ represents some measure of value of the total production.

See

- ▶ Input–Output Analysis
- ▶ Input–Output Coefficients
- ▶ Linear Programming

Acyclic Network

A network that contains no cycles.

See

- ▶ Graph Theory
- ▶ Network Optimization

Adjacent

Nodes of a graph or network are adjacent if they are joined by an edge; edges are adjacent if they share a common node.

See

- ▶ Graph Theory
- ▶ Network Optimization

Adjacent (Neighboring) Extreme Points

Two extreme points of a polyhedron that are connected by an edge of the polyhedron.

Advertising

Gurumurthy Kalyanaram[1], Frank M. Bass[1] and Dominique M. Hanssens[2]
[1]The University of Texas at Dallas, Richardson, TX, USA
[2]University of California, Los Angeles, CA, USA

Introduction

Advertising research has focused on three substantive areas: sales response to advertising, optimal advertising policy (constant spending or pulsing), competitive reactions and over-time effects. The research has employed econometric, time-serioes, optimization and game theoretic analytical techniques to address the issues. The advent of enormous amounts of scanner panel and internet data has led to some fruitful modeling at the individual household level. Contributions in each one of the three areas are discussed. A thorough review of optimal control advertising models is given in Feichtinger, Hartl, and Sethi (1994). Mathematical programming also has been a useful technology. Since some early successful applications of this technology for media planning, the progress has been limited because of measurement problems relating to advertising response function (Little and Lodish 1969). Advances in research, however, provide reasons for optimism in identifying the response function (Little 1979; Eastlack and Rao 1986). Heuristic approaches have been developed to estimate the media characteristics of reach and frequency (Rust and Eechambadi 1989).

Sales-Advertising Relationship

The first generally recognized model of importance was proposed by Vidale and Wolfe (1957). Building on a diffusion modeling framework, the Vidale-Wolfe model proposed that advertising directly persuades potential customers not currently buying from the firm, while those who are buying tend to forget (buy less) over time. Formally, the model is represented as follows:

$$x' = ru(1 - x) - kx,\ x(0) = x_0$$

where x is the market share, u is the level of advertising expenditure at time t, and k is the decay constant. The model suggests an exponential reach and decay phenomena with r and k rate parameters. Bass and Parsons (1969), following Bass (1969), developed a dynamic simultaneous equation model of sales and advertising and estimated this model on data for a frequently purchased consumer product. The empirical results from this analysis suggest that the advertising elasticity for the brand is small and the advertising expenditures are responsive to sales increases of other brands. A very interesting feature of this model is that it has good forecasting properties.

As far as estimation technology is concerned, there are three works that have provided insightful results. Bass and Clarke (1972) showed that statistical models of sales-advertising relationship need not be limited to the Koyck (1954) model. For example, nonmonotonic lag distributions are more appropriate for monthly data. Bass and Leone (1986) further examined the data interval issue. Rao (1986) has suggested that one should recognize the role of unobservable advertising expenditures in estimating the parameters of sales-advertising relationship associated with different data intervals.

With the evolution of a better appreciation of the advertising effects, the focus has shifted to models at the individual level. Blattberg and Jeuland (1981) postulated a micromodel that incorporated two well established advertising mechanisms, reach and decay. They assumed that the exposure of an individual to an advertisement can be characterized as a Bernoulli process, and the decay (forgetting) as an exponential process. These assumptions lead to a saw-tooth description of advertising effectiveness. The micromodel is aggregated to derive a model of advertising effects on the firm's sales. The model, while fairly flexible and general, provides insightful interpretations. The advent and explosion of scanner panel data over the last decade has accelerated efforts

to model at the individual level. The work by Pedrick and Zufryden (1991) was representative of this effort. They proposed a nonstationary, integrative, stochastic model approach that melds brand choice, purchase incidence and exposure behavior components. The integrated model, calibrated on scanner panel data, provided good fit and fairly accurate forecasts. Jedidi, Mela, and Gupta (1999) also used the scanner panel data and study the tradeoff between advertising and sales promotion for long-run profitability employing a heteroscedastic, varying parameter joint probit choice model. They show that advertising has a positive effect on brand equity, while sales promotions have a negative effect.

Optimal Advertising Policy

Researchers have been engaged in the examining what might be the optimal advertising policy given a budget constraint. Some have argued that constant advertising or chattering (vacillate between two levels of spending with infinite frequency) is probably the optimal policy (Sasieni 1971; Sethi 1973). However, others have found pulsing (Hahn and Hyun 1990; Feinberg 1992) to be optimal or better. Sasieni (1971) formulated the problem as follows:

$$\max \int_0^\infty [\pi x(t) - u(t)]e^{-rt}dt, \quad \text{with}$$
$$x' = g(x,u),\ x(0) = x_0$$

where $x(t)$ and $u(t)$ refer to sales and advertising at time t and r is the discount rate. Now, let g_u and g_x be the first partial derivatives and g_{uu} the second partial derivative. Then, employing Bellman's approach to dynamic programming and the classical Poincare-Bendixson theorem for phase space of differential equations, Sasieni showed that the optimal advertising policy is constant spending when it is assumed that for a given sales and advertising (1) the sales response would be the same or more positive if advertising level were higher ($g_u \geq 0$), (2) the sales response would be same or more positive if sales were at a lower level ($g_x \leq 0$), and (3) the sales response exhibits diminishing returns to increases in advertising level that so that the response curve is concave ($g_{uu} \leq 0$). The optimal policy, however, becomes chattering when the assumption of concavity of the response function is violated. Clearly, therefore, the shape of the response curve has become a matter of debate. There are many who find evidence for an S-shaped response curve (Little 1979; Eastlack and Rao 1986). A more definitive conclusion on the shape of the response curve would enhance the ability to model advertising better. This is, of course, a question for empirical examination.

Hahn and Hyun (1990) showed that when transaction costs above the ordinary media costs are included in Mahajan and Muller's model (1986), pulsing is the optimal policy. Feinberg (1992) introduced the concept of a filter and modifies the Sasieni model as follows:

$$x' = g(x,z), z' = G(u-z),$$

where z characterizes the filter. The only way to produce something constant in the Sasieni formulation is to fluctuate very rapidly, that is, to chatter. Since chattering is impossible in principle and constant spending is impossible in practice, they are two unrealizable ends of a frequency spectrum and are, in a sense, perceptually equivalent. The introduction of a filter allows Feinberg to mathematically equate the two. The filter exponentially smooths out the input. If the input is constant or chattering advertising, the filter yields a constant output. The filter output, however, is a pulsing policy for any nonconstant periodic input. Feinberg (1992) showed numerically that pulsing is a better policy than constant spending.

Research continues to demonstrate that pulsing is an optimal strategy. Bronnenberg (1998) has shown that under the assumptions of a discrete and interpretable Markov process and a constrained budget, a pulsing strategy is optimal, and the advertising effects on switching or repeat purchase affect both the length of the pulse and the optimal level of advertising. Adapting the Nerlove and Arrow (1962) model of advertising, where advertising is formulated as a function of awareness, Naik, Mantrala and Sawyer (1998) showed that pulsing strategies can generate greater total awareness than continuous advertising when the effectiveness of advertising varies over time.

Competition

This issue has received considerable attention. Three kinds of competitive models — differential game models, hazard rate models and competitive reaction models — have emerged.

The differential game models have been built on the Vidale-Wolfe or Nerlove and Arrow models, and employ either open-loop (Rao 1984) or closed-loop (Erickson 1991) deterministic games to solve them. Rao (1984) casts the model in the standard format. The value to a firm is the discounted profit which is defined as sales minus the advertising cost. Sales and advertising response functions are assumed to be strictly concave and convex, respectively. Further, sales at any time period are expressed as a geometric decay of last period's sales and this guarantees an upper bound for the value to the firm. The context is oligopolistic competition. In this setting, Rao demonstrates an industry open-loop Nash equilibrium. Most of the open-loop differential games confirm Rao's analysis. There have been efforts to compare the open-loop and closed-loop solutions to differential games. It has been found that closed-loop equilibrium strategies provide a better fit of the data, that is, actual spending levels in the market (Erickson 1991).

The work by Bourguignon and Sethi (1981) is a good representation of the hazard rate models applied to the study of competition in the context of advertising. These models are useful in situations where a firm must advertise to deal with the threat of entry by another firm. Bourguignon and Sethi characterize a special class of hazard rates given by $h(p, u) = (1 - F)^{n-1}$ where p and u represent price and advertising, $F(t)$ is the probability that the entry of the firm has occurred in the time interval $[0, t)$, and n is a parameter depicting the nature of potential entrants. Employing Pontryagin's maximum principle, the researchers show that, for certain conditions, the optimal policy for a firm is to forbid the entry of any competitor by setting p and u aggressively.

Competitive reaction to advertising may be assessed by multivariate time-series models. Steenkamp, Nijs, Dekimpe & Hanssens (2005) examined competitive reactions to advertising and promotion in over 400 consumer product categories, using vector-autoregressive models. They found that the predominant competitive reaction to advertising is no reaction at all, and that, when reaction does occur, it is often ineffective. Thus the ultimate impact of most promotion and advertising campaigns depends primarily on the nature of consumer response, not the vigilance of competitors.

Over-Time Effects

Many authors have argued that the effect of advertising extends beyond the period in which the expense is incurred. This raises questions about the duration length, i.e., how long does advertising have an impact, and the combined effect across the various periods during which advertising has an impact.

Leone (1995) reports the empirical generalization that the average advertising duration interval on sales is short, typically between 6 and 9 months. As a consequence, managers should not expect the tangible impact of an individual advertising campaign to last for years. In this respect, one should be aware of the well-documented data aggregation bias, in that the coarser the level of temporal aggregation in the data, the longer the duration interval one tends to obtain; see Russell (1988) for diagnosing and correcting such data aggregation bias.

Dekimpe and Hanssens (1995a) introduced persistence modeling, and demonstrated empirically that it is possible that advertising has a permanent effect on sales. Central in their conceptualisation is that one should take into account various channels through which advertising may impact (subsequent) sales. They identified six such channels: instantaneous effects, delayed response, purchase reinforcement, performance feedback, decision rules, and competitive reactions; see Hanssens (2011) for several business illustrations of the impact of these channels. As shown in Dekimpe and Hanssens (1995a), persistent marketing (advertising) effects can only occur in evolving environments, i.e., where the performance fluctuations are not just temporary deviations from a pre-determined (i.e., deterministic) level. Subsequent research (Nijs et al. 2001; Steenkamp et al. 2005) across many product categories and brands in the CPG (Consumer Packaged Goods) sector has established that such evolution is rare in both primary and secondary demand, and even more so when looking at market shares (Dekimpe and Hanssens 1995b). Evolution in

sales is most common in the early stages of the product life cycle. In such cases, advertising may (but need not) have persistent effects on the brand's future performance evolution. Pauwels and Hanssens (2007) find that, even in overall stable markets, there are selective and usually brief time windows that offer an opportunity for brands to benefit from long-term advertising effects. Still, most evidence to date supports the notion of business-as-usual scenarios, where the actions of brands only lead to temporary performance changes that eventually return to the mean.

In the absence of evolution and, hence, the absence of permanent effects, the question centers on the combined, or cumulative, effect of advertising. Lodish et al. (1995) report in this respect that if increased TV advertising has a significant impact on sales during the year of its expenditure, the sales impact is approximately doubled in the following 2 years. They also report, however, that if there is no significant effect in the short run, there will not be any significant long-term effect either. This finding is echoed in Hanssens (2011), who concludes that "consistently, it has been found that a short-run impact on consumer purchasing (sales) is a prerequisite for a long-term effect" (p. 3). Given that many advertising campaigns have no significant immediate effect, this is quite worrisome. On the other hand, when short-term effects are positive, they can be amplified up to five-fold by brand actions that involve other aspects of marketing, such as product-line extensions and sales promotions, as shown by Pauwels (2004). Similar results are reported in Srinivasan, Vanhuele and Pauwels (2010).

In sum, even when accounting for the fact that advertising's impact may extend beyond the week (month) in which it is spent, fairly small elasticities are typically found. These do not imply that advertising is wasteful, but instead that a tight relationship exists between sales revenue, advertising spending and profitabilty. Firms are well advised to study the effectiveness of their advertising as they make resource allocation decisions.

Concluding Remarks

The models reviewed suggest several conclusions relating to advertising. First, there are two mechanisms – growth and decay – in the advertising process. Second, the sales-advertising response curve is either concave or S-shaped. Third, pulsing is better (and may even be optimal) advertising policy. Fourth, competition matters and can be modeled in several ways. Fifth, the over-time effects of advertising are important; advertising's long-term effect is typically a multiple of its short-term impact; see Tellis and Ambler (2007) for an in-depth discussion of these issues.

See

- ▶ Game Theory
- ▶ Marketing

References

Bass, F. M. (1969). A simultaneous equation regression study of advertising and sales of cigarettes. *Journal of Marketing Research, 6*, 291–300.

Bass, F. M., & Clarke, D. G. (1972). Testing distributed lag models of advertising effect. *Journal of Marketing Research, 9*, 298–308.

Bass, F. M., & Leone, R. P. (1986). Estimating micro relationships from macro data: A comparative study of two approximations of the brand loyal model under temporal aggregation. *Journal of Marketing Research, 23*, 291–297.

Bass, F. M., & Parsons, L. J. (1969). Simultaneous-equation regression analysis of sales and advertising. *Applied Economics, 1*, 103–124.

Blattberg, R. C., & Jeuland, A. P. (1981). A micro-modeling approach to investigate the advertising-sales relationship. *Management Science, 27*, 988–1005.

Bourguignon, F., & Sethi, S. P. (1981). Dynamic optimal pricing and (possibly) advertising in the face of various kinds of potential entrants. *Journal of Economic Dynamics and Control, 3*, 119–140.

Bronnenberg, B. J. (1998). Advertising frequency decisions in a discrete markov process under a budget constraint. *Journal of Marketing Research, 35*, 399–406.

Dekimpe, M. G., & Hanssens, D. M. (1995a). The persistence of marketing effects on sales. *Marketing Science, 14*(Winter), 1–21.

Dekimpe, M. G., & Hanssens, D. M. (1995b). Empirical generalization about market evolution and stationarity. *Marketing Science, 14*, 109–121.

Eastlack, J. O., & Rao, A. (1986). Modeling response to advertising and pricing changes for V8 cocktail vegetable juice. *Marketing Science, 5*, 245–259.

Erickson, G. M. (1991). *Empirical analysis of closed-loop duopoly advertising strategies*. Working Paper. Seattle: University of Washington.

Feichtinger, G., Hartl, R. F., & Sethi, S. P. (1994). Dynamic optimal control models in advertising: Recent developments. *Management Science, 40*, 195–226.

Feinberg, F. (1992). Pulsing policies for aggregate advertising models. *Marketing Science, 11*, 221–234.
Hahn, M., & Hyun, J. S. (1990). Advertising cost interpretations and the optimality of pulsing. *Management Science, 37*, 157–169.
Hanssens, D.M. (2011). *What is known about measuring (forecasting & improving) the long-term impact of advertising*. Marketing Accountability Standards Board, Practitioner Paper 2011-01.
Jedidi, K., Mela, C. F., & Gupta, S. (1999). Managing advertising and promotion for long-run profitability. *Marketing Science, 18*, 1–22.
Koyck, L. M. (1954). *Distributed lags and investment analysis*. Amsterdam: North Holland.
Leone, R. P. (1995). Generalizing what is known of temporal aggregation and advertising carryover. *Marketing Science, 14*(3), G141–150.
Little, J. D. C. (1979). Aggregate advertising models, the state of the art. *Operations Research, 27*, 629–667.
Little, J. D. C., & Lodish, L. M. (1969). A media planning calculus. *Operations Research, 17*, 1–35.
Lodish, L. M., Abraham, M., Kalmenson, S., Livelsberger, J., Lubetkin, B., Richardson, B., et al. (1995). How TV advertising works: A meta-analysis of 389 real world split cable TV advertising experiments. *Journal of Marketing Research, 32*(May), 125–39.
Mahajan, V., & Muller, E. (1986). Advertising pulsing policies for generating awareness for new products. *Marketing Science, 5*, 89–106.
Naik, P. A., Mantrala, M. K., & Sawyer, A. F. (1998). Planning media schedules in the presence of dynamic advertising quality. *Marketing Science, 17*, 1–35.
Nerlove, M., & Arrow, K. (1962). Optimal advertising policy under dynamic conditions. *Economica, 29*, 129–142.
Nijs, V., Dekimpe, M. G., Steenkamp, J.-B., & Hanssens, D. M. (2001). The category-demand effects of price promotions. *Marketing Science, 20*(1), 1–22.
Pauwels, K. (2004). How dynamic consumer response, competitor response, company support, and company inertia shape long-term marketing effectiveness. *Marketing Science, 23*(4), 596–610.
Pauwels, K., & Hanssens, D. M. (2007). Performance regimes and marketing policy shifts. *Marketing Science, 26*(3), 293–311.
Pedrick, J. H., & Zufryden, F. S. (1991). Evaluating the impact of advertising media plans: A model of consumer purchase dynamics using single-source data. *Marketing Science, 10*, 111–130.
Rao, R. C. (1984). Advertising decisions in oligopoly: An industry equilibrium analysis. *Optimal Control Applications and Methods, 5*, 331–344.
Rao, R. C. (1986). Estimating continuous time advertising-sales models. *Marketing Science, 5*, 125–142.
Russell, G. J. (1988). Recovering measures of advertising carryover from aggregate data: The role of the firm's decision behavior. *Marketing Science, 7*(Summer), 252–70.
Rust, R. T., & Eechambadi, N. (1989). Scheduling network television programs: A heuristic audience flow approach to maximizing audience sharel. *Journal of Advertising, 18*(2), 11–18.
Sasieni, M. W. (1971). Optimal advertising expenditures. *Management Science, 18*, 64–72.
Sethi, S. P. (1973). Optimal control of the Vidale-Wolfe advertising model. *Operations Research, 21*, 998–1013.
Srinivasan, S., Vanhuele, M., & Pauwels, K. (2010). Mind-set metrics in market response models: An integrative approach. *Journal of Marketing Research, 47*(August), 672–684.
Steenkamp, J.-B. E. M., Nijs, V. R., Hanssens, D. M., & Dekimpe, M. G. (2005). Competitive reactions and the cross-sales effects of advertising and promotion. *Marketing Science, 24*, 35–54.
Tellis, G. J., & Ambler, T. (Eds.). (2007). *The SAGE handbook of advertising*. London: Sage Publications.
Vidale, M. L., & Wolfe, H. B. (1957). An operations research study of sales response to advertising. *Operations Research, 5*, 370–381.

Affiliated Values Bidding Model

A bidding model in which a bidder, upon learning that a competitor's valuation for what is being sold is higher than previously thought, will raise (or at least not lower) the bidder's own valuation. Affiliated values models include common value models and independent private value models as limiting cases.

See

► Bidding Models

Affine Transformation

A shifted linear transformation. An affine transformation on an n-dimensional vector space assigns to any point $\boldsymbol{x}$ the point $\boldsymbol{Ax} + \boldsymbol{c}$, where $\boldsymbol{A}$ is an $n \times n$ matrix and $\boldsymbol{c}$ is an n-dimensional vector.

Affine-Scaling Algorithm

An interior point method for linear programming based on affine transformations. In the primal affine-scaling algorithm, a problem in standard form is transformed so that the current solution estimate is mapped to the point (1, 1,. . ., 1). A movement is then made in the transformed space in the direction of the negative

projected gradient. The inverse affine transformation is applied to the resulting point, to obtain a new solution estimate in the original space. In the dual affine-scaling algorithm, similar ideas are used to solve the dual problem, with the affine transformations applied to the dual slack variables.

See

- ▶ Interior-Point Methods for Conic-Linear Optimization
- ▶ Nonlinear Programming

Agency Theory

- ▶ Organization

Agent

An agent is an autonomous decision-making entity that receives sensor information from an environment and acts based on that information. Agents may be humans or computer software, and may include hardware elements (e.g., robots). In agent-based simulation, agents interact within the environment to generate system behavior; and in artificial intelligence, the behavior of an agent is generally directed towards achieving certain goals.

See

- ▶ Agent-Based Simulation
- ▶ Artificial Intelligence

References

Macal, C. M., & North, M. J. (2011). Introductory tutorial: Agent-based modeling and simulation. In S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, & M. Fu (Eds.), *Proceedings of the 2011 winter simulation conference* (pp. 1456–1469).

Russell, S., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). New York: Prentice Hall.

Agent-Based Simulation

Charles M. Macal[1], Michael J. North[1] and Douglas A. Samuelson[2]
[1]Argonne National Laboratory, Argonne, IL, USA
[2]Infologix, Inc., Annandale, VA, USA

Introduction

Agent-based simulation (ABS) is a computational framework for simulating dynamic processes that involve autonomous agents. An autonomous agent acts on its own without external direction in response to situations the agent encounters during the simulation. Modeling a population of autonomous and heterogeneous agents that extensively interact is a defining feature of an ABS. ABS is a simulation approach made possible by advances in computational modeling and software. The agent perspective is unique among simulation approaches, unlike the process or activity perspectives of discrete-event simulation (DES), or the dynamical systems approach of system dynamics (SD). Agent-based simulation is most commonly used to model individual decision making and social and organizational behavior (Bonabeau 2001). Samuelson (2000) provides a brief overview of the early history of agent-based modeling, especially as applied to studying how organizations work, while Samuelson and Macal (2006) trace subsequent developments.

An agent is a general concept having broad applicability. Agents often represent people, or groups of people. Agent relationships represent processes of social interaction (Gilbert and Troitzsch 1999). For example, an individual's daily activities are explicitly modeled in an ABS of infectious disease transmission to understand infection patterns arising from contact patterns of individuals. In a supply chain ABS, agents are firms with decision-making behaviors about material sourcing and ordering, stocking, and shipping. In an ABS composed of artificial agents, collaborating robots search the landscape and communicate their findings to collectively accomplish a task. ABS models have also been developed that extend the notion of agents. Various types of animals, bacterial cells, cells composing the human immune system, and even molecules have been modeled as

individual, interacting agents. By modeling individual agents and their interactions, emergent system behaviors are often observed that were not explicitly programmed into the models.

The notions of behavior, decision making, and interaction apply to modeling many kinds of system. A common reason for modeling a system as an ABS is to consider agent learning and adaptation. At the individual level, learning and adaptation can be modeled as agent behaviors. At the population level, adaptation can be modeled by allowing agents to enter and leave the population, with the more successful agents increasing their relative numbers in the population over time.

The development of agent-based modeling tools, the availability of micro-data on agent transactions and interactions, and advances in computation have made possible a growing number of ABS applications across a variety of domains and disciplines.

Agent-based simulation has historical ties to complexity theory and the field of complex adaptive systems (CAS). CAS concerns itself with questions about how complex systems observed in nature, which are composed of autonomous agents with limited cognitive and perceptual abilities, can self-organize themselves to be better suited to their environment (Holland 1975). The first agent-based modeling software tool, Swarm, was developed to model complex adaptive systems, specifically to investigate aspects of Artificial Life (ALife) (Langton 1989). Holland and Miller (1991) appear to be the first to use the term agent in models of this type.

Agent-based simulation is also closely related to the field of multi-agent systems (MAS), but MAS has a somewhat different focus and legacy; MAS is a subfield of distributed artificial intelligence. Individual-Based Modeling (IBM) is another field related to ABS. IBM has a history associated with ecological modeling, where it was important to model heterogeneous populations of agents, but agent interaction was less important of a consideration. Agent-based modeling (ABM) is often used synonymously for ABS; sometimes the term ABMS is used (Agent-based Modeling and Simulation) to refer to the entire field.

Many models now thought of as agent-based simulations were originally developed in the form of cellular automata (CA). CA were originated by John von Neumann to investigate the theory of machine self-replication. Cellular automata use a grid divided into cells as the environment. The cells immediately surrounding an agent are its neighborhood for cell interaction. For example, a cell's von Neumann neighborhood consists of the four cells immediately above, below, and on either side of the cell; the Moore neighborhood consists of the eight cells completely surrounding the cell. Sakoda's checkerboard model was essentially a cellular automaton that simulated the dynamic process of social interaction in one of the first recognizable examples of an ABS (Sakoda 1971). Schelling used a checkerboard framework to study housing segregation patterns (Schelling 1971). Schelling's model was not computerized, and agents were represented as coins moving on a checkerboard. A key finding by Schelling was that patterns emerge from agents interacting that are not necessarily implied or even consistent with the objectives of the individual agents. Analytical results are seldom available for such simple, yet complex systems. Computer simulation is necessary to determine system behaviors that result from the micro-level agent interactions. Axelrod and Hamilton (1981) studied the emergence of cooperation and reciprocation strategies among agents in an evolutionary game set on a cellular automaton grid (Axelrod and Hamilton 1981). Epstein and Axtell introduced artificial societies in their SugarScape model that represents an entire society "from the ground up" by modeling its individuals and their interactions (Epstein and Axtell 1996). These initial ABS models have been blueprints for agent-based models for many years, and their influence in form and approach can be seen in ongoing ABS developments.

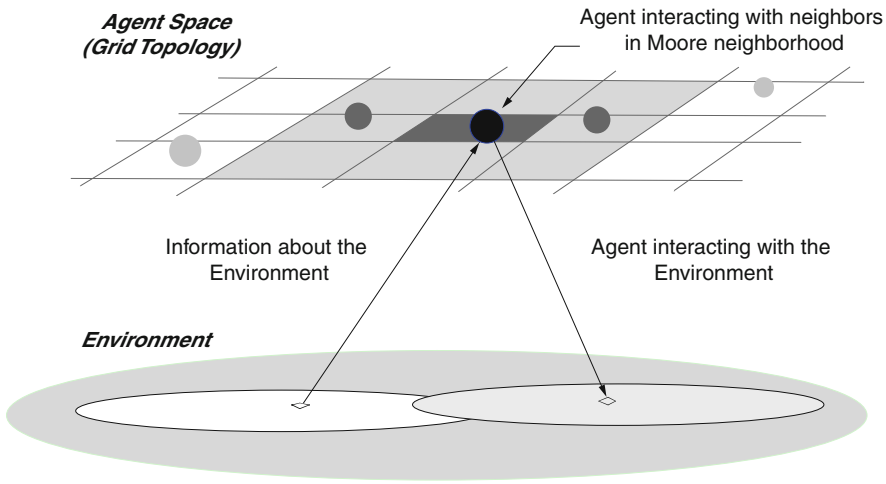
Elements of Agent-Based Simulation

Agent-based simulation does not have an accepted formalism, such as the Discrete Event System Specification (DEVS) formalism for discrete-event simulation (Ziegler et al. 2000). An informal notation for agent-based simulation covers the essential elements of an ABS. Figure 1 shows the elements of a typical ABS.

An agent-based simulation is represented by four elements:

$$\text{ABS} = \{A,\ I,\ E,\ T\} \tag{1}$$

where A = a set of agents, I = agent interaction space, E = an environment independent of the agents, and



Agent-Based Simulation, Fig. 1 Elements of an Agent-based Simulation

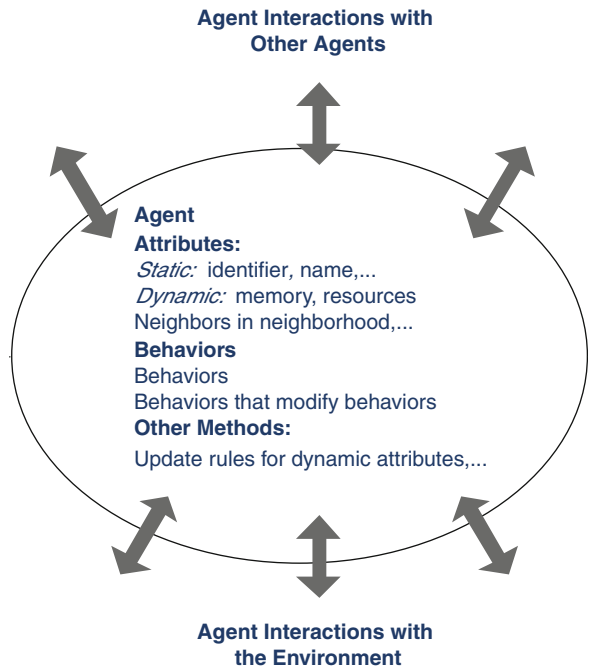
T = a time advance mechanism. An agent interaction space, I, is the social space where agents interact with other agents. The environment, E, is a backdrop against which the agents receive location-specific information, for example, a landscape that agents move over. Agents communicate with the environment, and mobile agents move through it. E encompasses the methods for updating the state of the environment as well as the environment's attributes. The time advance mechanism, T, is the procedure for advancing time in the simulation.

A distinction can be made between ABM and ABS based on (1). The term simulation, as used in the OR/MS community, refers to modeling a dynamic process that unfolds over time (Pritsker 1979). Some optimization algorithms, however, most notably particle swarm optimization (PSO), satisfy all of the requirements of an ABS noted above. Since PSO is a static algorithm without time, the time advance mechanism in (1) is replaced with a more general agent interaction mechanism.

An agent is represented by five elements. Figure 2 shows the elements of a typical agent:

$$a = \{B,\ S,\ D,\ N,\ M\} \qquad a \in A \qquad (2)$$

where B = a set of agent behaviors, S = a set of static attributes for the agent, D = a set of dynamic attributes for the agent, N = the agent's neighborhood, and M = a set of mechanisms for updating the agent's state.



Agent-Based Simulation, Fig. 2 Elements of an Agent

Agent behaviors include rules by which agents transform sensory input information coming from an agent's neighbors and the environment into decisions and actions. Behaviors include deciding, communicating, moving, learning, etc. Each agent has a set of static attributes, S, consisting of, for example, demographic variables such a name, birth date, gender, etc. Dynamic attributes, D, are attributes that are updated during the simulation.

Dynamic attributes include an agent's memory that records relevant agent experiences and interactions. Agents interact with a small subset of agents in a local area termed the agent's neighborhood, N. An agent's neighborhood may be static and fixed in space throughout the entire simulation, or may be dynamic and constantly updated as the agent moves or develops new relationships. The mechanisms for updating the agent's state, M, operate on the agent.

Agents

Based on a study of ABS applications, there does not appear to be general agreement on what constitutes an agent (Macal and North 2010). Some authors consider any model in which a population is represented as a set of individuals to be an ABS. Others require agents to have behaviors that dynamically respond to conditions in the model and to interact extensively with other agents. Still others require an agent to exhibit adaptive behavior to be properly classified as an agent. There are, however, several characteristics that an agent should have to be distinguished from a software object or conventional entity in a simulation model. These are described below.

First, an agent is autonomous and self-directed, able to function independently in its environment and in its interactions with other agents over the limited range of situations it encounters in the model. Thus, agents do not necessarily have any degree of intelligence, but can respond independently to events.

Second, an agent has a well-defined state that varies over time. Just as a system has a state consisting of the collection of its state variables, an agent also has a state that represents the essential attributes associated with its current situation. The state of an agent-based model is the collective states of all the agents in the model, along with the state of the environment. It follows that to implement an agent's behavior requires that the information upon which the agent's behavior is based be included in the set of agent attributes.

Third, an agent is a social entity, having dynamic interactions with other agents that influence its behavior. Agents have protocols for interaction with other agents, such as for communication, movement and contention for space, the capability to respond to the environment, etc.

Agents often have other important characteristics, such as adaptability and purposefulness. An agent may adapt its behaviors, by selecting the most successful behaviors over repeated interactions, and learning new behaviors. An agent may simply execute its behaviors with no apparent purpose or an agent may have goals to achieve, but not necessarily objectives to maximize. Agents in an ABS are often heterogeneous, having diverse characteristics and behaviors across the entire population of agents.

Agents and Behavior

Modeling agent behavior is an important part of developing an ABS. Common approaches are to apply behavioral theories from social and cognitive sciences, develop behavioral representations based on surveys and empirical observations, and hypothesize agent behaviors based on reasonable assumptions amenable to testing in the model.

Rational choice theory provides the most common model of decision-making used in the management and social sciences. Rational choice models are based on the assumption that agents have the ability to compute optimal solutions to arbitrarily complex decision problems based on utility maximization.

Bounded rationality is an alternative to rational choice theory. Bounded rationality assumes that actors are not able to optimize their behaviors because they possess limited computational resources and information on which to base decisions (Simon 1997). Actors instead satisfice: they make suboptimal, yet adequate decisions using simple heuristics or rules of thumb. Learning from experience is important to exploring satisficing behaviors and creating heuristics in which agents have limited information. Agent-based modeling can represent bounded rational agents in which agents learn from repeated experiences and make decisions based on affective factors, i.e., emotion, in addition to rational factors.

Although it is common to describe agent behavior by if-then rules, agent behavior can be modeled in many ways, from simple rules to abstract models. Abstract representations, such as neural networks and genetic programs, relate agent inputs to outputs through adaptive mechanisms and filters. For example, Manson (2006) uses evolutionary programming in agent-based modeling to implement the theory of bounded rationality in a land use ABS.

ABS and Space

In addition to time, most ABS models have a notion of physical space. Some agent-based models are

non-spatial, meaning there is no need for agents to have a locational attribute. In a typical non-spatial ABS, pairs of agents are randomly selected from the pool of agents, the pairs interact, possibly changing the agents' states, and the agents then return to the agent pool for the next iteration. This is equivalent to the assumption of perfect mixing, for example, in an epidemiological model in which agents have random contact and possibly transmit infection. Other examples include molecules randomly reacting as part of a chemical transduction network and economic agents randomly selected to engage in bilateral trades in a market.

Cellular automata grids are an efficient way of representing and updating agent interactions in a dynamic simulation model. By convention, each grid cell in a CA is either unoccupied or occupied by no more than one agent. Many early agent-based models used a CA grid as their environment. But, generally, CA models do not represent how agents interact in the real world. Networks and other structures are now commonly used to represent agent interaction in agent-based simulations.

Networks are general and flexible representations of agent interaction; an agent interacts with the set or a subset of the agents that it is linked to its network. For example, a contact network in an infectious disease ABS indicates the agents contacted by an agent in its daily activities. Networks are static, predefined by the modeler, or dynamic, changing according to the results of agent interactions that occur endogenously in the simulation. For example, an ABS of a social networking site has models that specify why people join and leave the social network. An agent may participate in multiple networks in the same model. For example, an agent in an infectious disease ABS might be connected to an information network of trusted sources of public health information in addition to its contact network.

An agent interaction network can be implemented either implicitly by including a list of an agent's current neighbors as a dynamic agent attribute, or explicitly by creating a network entity which links all agents to their neighbors. This is a model implementation consideration, as the computational performance of the model is sensitive to the approach taken, especially for larger networks.

Other agent interaction topologies are also used in ABS models. These include continuous space in 1-, 2- or 3-dimensions, and geographical information systems (GIS). Keeping track of agent neighbors in continuous space can be computationally challenging, and, often, a spatial bucketing approach is used that leads to some degree of approximation. An example of the use of a GIS topology consists of agents that move over geographically defined patches; for example, neighborhoods or zip-code areas, and is often accomplished by directly linking an ABS with a GIS (Brown et al. 2005).

ABS and Time

Agent-based simulation is sometimes described as a form of discrete-event simulation because of the similarity of the event scheduling mechanisms commonly employed in DES and ABS. An event scheduling mechanism is the logic of how the simulation advances time and how events are generated internally within a model. A scheduling mechanism in a DES moves the simulation forward through time to the next point at which an event is scheduled to occur. In an ABS, events consist of the times at which agent interactions occur. A scheduling mechanism in a continuous simulation moves the simulation forward continuously through time; a continuous simulation is described by a set of differential equations that indicate how the system state changes over time as a function of the current and past states. Virtually all ABS are discrete-event simulations, although the ABS framework does not preclude the development of continuous or combined continuous/discrete simulations.

A special case of discrete-event simulation is time-stepped simulation in which time advances at fixed time increments. Virtually all of the early agent-based simulations were time-stepped simulations in which each agent interacted with its neighbors at each time step. This time advance mechanism is the one used by cellular automata.

A general representation for a time-stepped, discrete-event, agent-based simulation is an iterated map from dynamical systems theory in which a transform function is applied to the system state at each time point to update the state for the next time:

$$x_{t+1} = F(x_{\mathrm{t}})$$

where x_t is the system state at time t and the transform F is a general mapping applied to the system state each time period. The system state, x_t, is composed of the

states of all the agents in the simulation, as well as the state of the environment at t. The transform F is not a function per se, but represents the entirety of the complex logic encoded in the simulation model and can be a deterministic or a stochastic mapping. For example, some ABS models are in the form of a Markov chain model. Agent state transitions are represented by a matrix of probabilities. Slightly more complex ABS models condition agent state transitions on events that have happened to an agent in its past, information stored in an agent's memory, akin to a semi-Markov model.

Conflicts may arise in time-stepped ABS when all agents interact simultaneously at a time step due to agents contending for the same space or resources. A common approach to resolving such conflicts is to randomly reorder the sequence of agent interactions at each time step. This removes any bias in the simulation results due to an arbitrary ordering of agent interactions. The disadvantage is that many replications of the simulation have to be run to generate statistically significant results and understand the full range of model outcomes due to the order of agent updates. If agent interactions are scheduled at discrete real-valued times, as in DES, the problem is avoided because ties at specific times are unlikely.

As the need for realistic, large-scale agent-based simulations has advanced, event-scheduling mechanisms have been generalized in ABS development software tools. The most advanced ABS toolkits provide users the tools to implement time-stepped or discrete-event scheduling. Events can be exogenously specified to occur at particular times, or generated and scheduled endogenously based on the outcomes of agent interactions and agent monitors that detect changes in agent states.

Event calendars offer a major improvement in computational efficiency in DES. At the completion of the computation arising from an event, the system can simply skip to the next time at which an event is scheduled to occur. While calendaring is possible for ABS as well, it is infrequently used. The much higher number of interactions among events in an ABS requires correspondingly more re-computation as the new events affect schedules of other events. In turn, these interactions and re-computations greatly complicate debugging. Finding a computational improvement for ABS along the lines of DES calendaring is a promising area of research.

ABS and Emergence

One of the most oft cited reasons for developing an ABS is the ability to capture emergent processes, i.e., processes whose outcome cannot be understood or anticipated solely from examination of the individual parts of the system (Bedau and Humphreys 2007). Emergence, in the most general sense, refers to the emergence of order, which is a well-defined notion and can be expressed using various entropy measures. It can easily be shown that simple ABS models, which are completely described by deterministic rules, can produce self-organizing, emergent and sustainable patterns that have not been explicitly programmed into the models.

Swarm intelligence is a concept related to emergence (Bonabeau, et al. 1999). Natural systems seemingly exhibit collective intelligence without the existence of, or the direction provided by, a central authority. Typical examples are the workings of ant colonies and beehives, the schooling behavior of fish, and search behavior of collaborating predators. Swarm intelligence has inspired practical optimization techniques such as ant colony optimization and particle swarm optimization that have been used to solve practical scheduling and routing problems. Swarm intelligence algorithms can be implemented in an agent-based simulation framework, as described by (1)-(2) above.

Agent-Based Simulation Applications

Agent-based simulation has been applied in many scientific disciplines—physical, life, medical, social, and management. ABS applications in economics (Tesfatsion and Judd 2006), sociology (Macy and Willer 2002), anthropology (Kohler et al. 2005), cognitive science (Sun 2006), business, marketing (Rand and Rust 2011) and many other areas suggest that ABS is a general technique with wide application. Macal and North (2011) discuss many published applications as diverse as agriculture, air traffic control, anthropology, biomedical research, crime analysis, ecology, energy analysis, epidemiology, evacuation, market analysis, organizational decision making, and social networks.

Agent-based simulation applications range from elegant conceptual models having minimalist detail, to large-scale models having much detail that correspond closely to the real-world system modeled.

Minimalist models are based on a set of idealized assumptions, designed to capture only the most salient features of a real-world system. These models are often used as electronic laboratories to implement and test the implications of qualitative theories for a range of many scenarios.

Large-scale ABS models have been developed that answer real-world policy questions, include real data, and have been shown to pass appropriate validation tests to establish their credibility. Such models have been developed for the operation of physical infrastructures and associated economic markets, including electric power, epidemics, economies, traffic and transportation, pedestrian movement, and evacuation modeling. Agent-based models are useful for modeling very high volumes of entities for cases in which simple sets of agent behaviors are adequate. For example, prior to 2006, discrete-event models of crowd movement could handle no more than a few thousand people in real time. Samuelson et al. (2007, 2010) used custom agent-based software to depict real-time evacuation from a stadium for up to 70,000 people.

Large-scale ABS models have been used to inform policy-making for epidemiological studies (Germann et al. 2006). Here, individual agents are modeled as they go through their daily activities, make contact with other agents, and possibly pass on infection (Carley et al. 2006). The models can incorporate realistic agent behaviors, such as how agents respond to their disease state, public health information, and health care interventions (Epstein 2009). ABS models are used as in silico (computer-based) experimental laboratories to understand how these behaviors might affect the outcome of an epidemic and to understand their possible severity under various assumptions.

Developers of ABS models contend that agent-based modeling offers unique benefits for the problems studied in their disciplines that are beyond conventional modeling approaches. For example, many economic models assume that the real-world system will be driven toward a long-run stable, equilibrium state. The conditions for the equilibrium state are represented by a set of non-linear equations that are to be solved in order to solve the model. An agent-based simulation model, in contrast, makes no assumptions about a long-run equilibrium state, but rather computes the outcome of the process of repeated agent behaviors and interactions (Axtell 2000). Agent-based models are particularly useful for assessing when equilibria are likely to cease to exist, what transient behavior can then be expected, what trigger events are likely to promote stability or instability, and how robust the system is likely to be.

Agent-Based Simulation Development

ABS Design

Methods for developing ABS largely follow established methods for developing any kind of simulation model. Additional tasks include defining agents, modeling agent behaviors and interactions, and validating agent models and results (North and Macal 2007). There is also a close connection between agent-based simulation design and object-oriented modeling through the use of object-oriented design methods, such as design patterns (Grimm et al. 2006).

The complexity of agent interactions can make both model debugging and validation difficult. Because of this, validation and verification of agent-based models is a vital and interesting area of research. A comprehensive review by the National Research Council (2008) proposed a number of principles, primarily a clear definition of the model's intended purpose and a strict adherence to assessing whether the model was useful for that purpose. The effect of stringent adherence to this discipline on the discovery and evaluation of unexpected phenomena appears to be a subject that needs further exploration. An additional, open question is: how complex can a model become without overwhelming the modeler's ability to interpret the output, in particular to distinguish genuine rare emergent phenomena from deficiencies in the programming? One promising approach is to use agents to assist in validation, (Niazi et al. 2009). The rapid development of computational capabilities and new logical approaches in software promise to keep the discussion on ABS validation ongoing and lively.

ABS Toolkits

A number of computing alternatives are available for developing agent-based simulations. These range from laptop computers to computer clusters, and, potentially, to cloud computing.

Desktop computing environments for ABS development include spreadsheets, such as Excel augmented by a macro programming language for

programming agent behaviors and time-advance mechanisms, and general computational mathematics systems such as MATLAB and Mathematica. Desktop computing environments can be used to develop agent models, although the agent-specific functionality must to be written by the developer from scratch, because no agent-based packages exist for these systems. Desktop agent-based models may not scale well for larger applications. Whatever software is chosen, agent-based simulations are often developed in phases using multiple approaches. Projects often begin small, using one of the desktop tools to prototype agent behaviors and to perform limited analyses. They are then scaled up using agent-based toolkits in a later phase of development to capture larger numbers of agents and more complex agent behaviors.

Agent-based models are also developed directly in programming languages such as C++, Java, and Python, especially for large-scale applications. The object-oriented modeling paradigm these languages embody is the basis for most agent-based modeling; an agent can be considered a self-directed object with the capability to autonomously choose actions in response to an agent's situation. The use of object classes as agent templates and object methods to represent agent behaviors is a natural extension to ABS. Most large-scale agent-based modeling toolkits are also object-oriented.

Since the original Swarm toolkit, there has been a steady progression of ABS software toolkits, development environments, and modeling approaches. Open source and/or freely available ABS development environments include NetLogo, Repast Simphony, and MASON (Macal and North 2010). These environments provide special facilities for modeling agents and are designed for new users to get started as quickly as possible in developing ABS. Commercial agent-based modeling software includes AnyLogic, among others.

Concluding Remarks

Agent-based simulation can offer distinct advantages over conventional simulation approaches, depending on the type of problem being modeled. Agent-based simulation is often used when:

- The system being modeled has a natural representation as being composed of interacting agents,
- The past is no predictor of the future, and the behavior of the system must be built from the ground up based on the behaviors and incentives of the individual agents,
- Process structural change needs to be an endogenous result of the simulation, rather than an input to the simulation, and
- Scaling-up to arbitrary levels is important, in terms of the number of agents, agent interactions, and agent states.

Agent-based simulation offers benefits compared to other modeling approaches when agents in the model:

- Have behaviors and make decisions central to the questions to be addressed by the simulation,
- Adapt and change their behaviors in response to actions and events in the model,
- Engage in repeated strategic interactions and can learn from their experiences,
- Have dynamic, changing, or evolving relationships with other agents,
- Self-organize into cohesive groups or organizations and the mechanisms by which this happens are known or hypothesized, and
- Have a spatial component to their behaviors, movements, and interactions.

Agent-based simulation has grown rapidly since the mid-1990s and many new applications continue to be published. Disciplines where simulation has not previously been the modeling technique of choice are experimenting with ABS, and research on the theoretical and methodological foundations of ABS is very active.

See

- ▶ Agent
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Swarm Intelligence
- ▶ System Dynamics

References

Axelrod, R., & Hamilton, W. D. (1981). The evolution of cooperation. *Science, 211*(4489), 1390–1396.

Axtell, R. (2000). *Why agents? On the varied motivations for agent computing in the social sciences* (Working Paper 17). Washington, DC: Center on Social and Economic Dynamics, Brookings Institution.

Bedau, M. A., & Humphreys, P. (Eds.). (2007). *Emergence: Contemporary readings in philosophy and science*. London: MIT Press.

Bonabeau, E. (2001). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences of the United States of America, 99*(3), 7280–7287.

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm intelligence: From natural to artificial systems*. New York: Oxford University Press.

Brown, D., et al. (2005). Spatial process and data models: Toward integration of agent-based models and GIS. *Journal of Geographical Systems, 7*(1), 25–47.

Carley, K., et al. (2006). Biowar: Scalable agent-based model of bioattacks. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 36*(2), 252–265.

Epstein, J. (2009). Modelling to contain pandemics. *Nature, 460*(6), 687.

Epstein, J., & Axtell, R. (1996). *Growing artificial societies: Social science from the bottom up*. Cambridge, MA: MIT Press.

Germann, T., Kadau, K., Longini, I., & Macken, C. (2006). Mitigation strategies for pandemic influenza in the United States. *Proceedings of the National Academy of Sciences, 103*(15), 5935–5940.

Gilbert, N., & Troitzsch, K. G. (1999). *Simulation for the social scientist*. Buckingham, UK: Open University Press.

Grimm, V., et al. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling, 198*(1–2), 115–126.

Holland, J. H. (1975). *Adaptation in natural and artificial Systems*. Ann Arbor, MI: University of Michigan Press.

Holland, J., & Miller, J. H. (1991). Artificial adaptive agents in economic theory. *The American Economic Review, 81*(2), 365–371.

Kohler, T. A., Gumerman, G. J., & Reynolds, R. G. (2005). Simulating ancient societies. *Scientific American*.

Langton, C. G. (1989). Artificial life. In C. G. Langton (Ed.), *Artificial life: The proceedings of an interdisciplinary workshop on the synthesis and simulation of living systems* (pp. 1–47). Reading, MA: Addison-Wesley.

Macal, C. M., & North, M. J. (2011). Introductory tutorial: Agent-based modeling and simulation. In S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, & M. Fu (Eds.), *Proceedings of the 2011 Winter Simulation Conference* (pp. 1456–1469).

Macal, C. M., & North, M. J. (2010). Tutorial on agent-based modelling and simulation. *Journal of Simulation, 4*(3), 151–162.

Macy, M. W., & Willer, R. (2002). From factors to actors: Computational sociology and agent-based modeling. *Annual Review of Sociology, 28*, 143–166.

Manson, S. M. (2006). Bounded rationality in agent-based models: Experiments with evolutionary programs. *International Journal of Geographical Information Science, 20*(9), 991–1012.

National Research Council. (2008). *Behavioral modeling and simulation: From individuals to societies*. Washington, DC: National Academies Press.

Niazi, M., Hussain, A. & Kolberg, M. (2009). Verification and validation of agent-based simulations using the VOMAS approach. *Proceedings of the Third Workshop on Multi-Agent Systems and Simulation '09 (MASS '09)*. Sep 7–11, 2009. Torino, Italy.

North, M. J., & Macal, C. M. (2007). *Managing business complexity: Discovering strategic solutions with agent-based modeling and simulation*. Oxford, UK: Oxford University Press.

Pritsker, A. A. B. (1979). Compilation of definitions of simulation. *Simulation, 33*(2), 61–63.

Rand, W., & Rust, R. T. (2011). Agent-based modeling in marketing: Guidelines for rigor. *International Journal of Research in Marketing, 28*(3), 181–193.

Sakoda, J. M. (1971). The checkerboard model of social interaction. *Journal of Mathematical Sociology, 1*, 119–132.

Samuelson, D. (2000). Designing organizations. *OR/MS Today, 27(6)*.

Samuelson, D., & Macal, C. (2006). Agent-based modeling comes of age. *OR/MS Today, 33*(4), 34–38.

Samuelson, D., et al. (2007). *Agent-based simulation of mass egress after an improvised explosive device attack*. Homeland Security Institute Final Report to the Department of Homeland Security, Science and Technology Directorate. HSI Document Number RP06-IOA-31-03.

Samuelson, D., et al. (2010). Agent-based simulations of mass egress after an IED attack. In W. Klingsch, C. Rogsch, A. Schadschneider, & M. Schreckenberg (Eds.), *Pedestrian and evacuation dynamics 2008 (PED2008)*. London/ New York: Springer.

Schelling, T. C. (1971). Dynamic models of segregation. *Journal of Mathematical Sociology, 1*, 143–186.

Simon, H. A. (1997). Behavioral economics and bounded rationality. In H. A. Simon (Ed.), *Models of bounded rationality* (pp. 267–298). Cambridge, MA: MIT Press.

Sun, R. (2006). *Cognition and multi-agent interaction: From cognitive modeling to social simulation*. Cambridge, UK: Cambridge University Press.

Tesfatsion, L., & Judd, K. L. (Eds.). (2006). *Handbook of computational economics, volume II: Agent-based computational economics*. Amsterdam: Elsevier/North-Holland.

Ziegler, B. P., Praehofer, H., & Kim, T. G. (2000). *Theory of modeling and simulation* (2nd ed.). San Diego, CA: Academic Press.

Agriculture and the Food Industry

Filmore Bender[1] and Gerald Kahan[2]
[1]University of Maryland, College Park, MD, USA
[2]McCormick and Company, Sparks, MA, USA

It is often difficult to determine where the agricultural sector of an economy ends and the nonagricultural sector begins. For the purpose of this article, the agricultural sector of the economy is defined as production and supply of agricultural inputs, the

production of agricultural goods on farms and ranches, the processing and transportation of those goods, as well as the wholesaling and retailing of finished products. Defined in this way, the agricultural sector of the economy in the United States represents approximately 24% of the gross national product.

As with the nonagricultural sector of the economy, operations research was first used to solve agricultural problems in the 1940s and 1950s. *A Survey of Agricultural Economics Literature, Vol. 2: Quantitative Methods in Agricultural Economics, 1940s to 1970s* traces the development of operations research in addressing problems of importance to agriculture (Judge et al. 1977). This work ranged from quantifying production functions to the development of models, simulation structures and the use of linear programming and nonlinear optimization models to quantify or predict economic consequences, or alternatively, the use of these tools to solve specific problems for specific firms.

Different components of the agricultural economy have embraced the tools of operations research with different levels of enthusiasm. Those sectors of agriculture that can exercise considerable control of inputs and environmental factors (e.g., feeder cattle, broilers, eggs, pork and dairy production) began adopting the tools of operations research during the late 1950s. By 1965, essentially all of the feed formulated for poultry in the United Sates was done using least-cost linear-programming feed formulations. Simultaneously, during the 1960s, the beef industry began to adopt linear programming on a limited basis for least-cost feed formulation and for the development of optimal production and marketing strategies. The use of linear programming for least-cost dairy rations became standard practice during the late 1970s. Forestry, which is a branch of agriculture, uses multi-period linear programming models to determine optimal planting and harvesting schedules.

A number of interesting examples have been reported in the literature which describe how linear programming was used to solve a variety of agriculture problems. Upcraft et al. (1989) reported that the soil water deficit is the main decision variable that British farmers monitor to decide when to irrigate a particular field and how much water to use. The decision is generally based on the soil water deficit in the first strip to be irrigated within each field. A mixed-integer linear program was constructed to model the short-term irrigation scheduling problem for a hose-reel/rain-gun irrigation system. Optimal schedules were produced by quantifying the costs and benefits of irrigation, subject to the constraints of equipment, labor, and availability of water. The model is unique in producing whole farm-irrigation schedules, rather than individual field schedules for hose-reel/rain-gun irrigation systems.

The efficient operation of a beef cattle feedlot is controlled by the price of the animals, purchase and selling weights, and the feeding system. The optimal feeding system involves feeding least-cost rations to animals at each stage in the production process. Glen (1980) reported the development of an optimal method for determining optimal feeding systems that meet the nutrient standards recommended by the US National Research Council. The approach involved using linear programming to determine the least-cost rations to produce specified live weight gains in animals of known live weight. Dynamic programming was used to determine the optimal sequence of rations to feed to produce animals of specified live weight from known live weight at minimum cost, using least-cost rations from the linear programming model. Results from the dynamic programming model can be used to determine the optimal combination of purchase weight, selling weight, and feeding system. The linear programming model must be solved a large number of times to use the dynamic programming model.

In assessing feeding policy in livestock production, it is generally assumed that an optimal feeding policy will involve using least-cost rations throughout the production process. Glen (1980) showed that this assumption may not always be valid, particularly when the supply of some of the feedstuffs used for feeding the livestock is limited. A technique for testing the validity of this assumption was presented using a linear programming model of an integrated crop and intensive beef production enterprise in which some of the crops are used for livestock feeding. An interactive solution procedure was proposed for cases where this assumption was not valid. While the computational burden associated with the procedure for finding an improved solution is large, experience with realistic data suggests that the results from the linear programming model are likely to be optimal.

Intra-year milk supply patterns depend largely on the distribution of cow calving dates which are in turn

influenced by climatic conditions. The most important and least-costly input to milk production is the fresh growth and high digestibility of grass in spring and early summer that often gives rise to a highly seasonal distribution of calving resulting in a seasonal milk supply pattern. However, milk for liquid consumption and production of perishable milk products must be geared to meet a constant consumer demand throughout the year, which necessitates a considerable amount of production outside the least-cost period. Killen and Keane (1978) re-ported on the development of a linear programming model that gives the distribution of calving dates, minimizes production costs, and meets consumer demand for milk and related products. In addition, the dual solution gives a set of seasonal prices which should be paid to producers, equitably compensating them for the costs they incur.

The agricultural sector deals with a biological system. By its very nature, agriculture has elements that foster the use of operations research techniques and other elements that greatly impede the application of these tools. Because many agricultural production units are relatively small in size, they are unable to adopt operations research techniques in a cost effective manner. On the other hand, because agricultural firms are dispersed, those firms that either supply inputs to farms or harvest and process agricultural products can make effective use of truck routing and other spatial optimization techniques.

Because of the savings in costs that can be achieved, as well as the increasing availability of computers and software, it is reasonable to expect increasing use of the tools of operations research in agriculture. In fact, as early as 1973, Beneke and Winterboer published *Linear Programming Applications to Agriculture*, a book devoted exclusively to the use of linear programming in agriculture.

In the food industry, linear programming is becoming increasingly common. Publications have re-ported the use of linear programming in formulating preblended meats (Rust 1976); luncheon or sandwich meat (IBM 1966; Wieske 1981); a protein-enriched luncheon sausage (Nicklin 1979); bologna (IBM 1966); frankfurters (IBM 1966); and a variety of sausage products (MacKenzie 1964; IBM 1966; Skinner et al. 1969). Ice cream is another food product which has been successfully formulated using linear programming (IBM 1964; Dano 1974; Singh et al. 1979).

Cereal-based food blends have been formulated using linear programming to insure adequate levels of good-quality protein. Since these blends are sometimes shipped to developing countries, linear programming has helped to ensure that the prominent grain of the country is present in the blend as a major ingredient. It is desirable to blend cereal grains, since plant proteins are usually deficient in one or more of the essential amino acids. Inglett et al. (1969) used linear programming to bring the essential-amino-acid pattern of a cereal-based food as close as possible to the pattern found in a hen's egg. Cavins et al. (1972) used linear programming to formulate a least-cost cereal-based food. The protein quality was controlled by setting both lower and upper limits on each essential amino acid in terms of its percent of total essential amino acid content. Hsu et al. (1977a, b) studied the blending of a wide range of plant and animal protein sources in formulations for bread, pasta, cookies, and extruded cornmeal snack and sausage. Constraints were used to restrict both the nutritional and functional properties.

Roush et al. (1994) reported on using chance constrained programming to formulate commercial feeds for animals. Nutritional consistency of finished feeds increased by 40 % while costs dropped compared to feeds formulated by linear programming with a margin of safety. With the exception of the probabilistic constraints, the objective function and most other constraints were linear in this model.

A detailed description of the formulation of a low-cholesterol, low-fat beef stew using linear programming was given by Bender et al. (1976). The objective was to minimize cost, while enforcing nutritional and other constraints based on the recommendations for fat-modified and low-cholesterol diets. These constraints were for a 100-g portion of stew and set an upper limit on cholesterol content; a lower limit on protein, vitamin A, thiamin, riboflavin, niacin, vitamin C, and iron; and both an upper and a lower limit on carbohydrate, fat, and calories.

Dano (1974) provided an full description of the application of linear programming to a beer-blending problem, and Wieske (1981) described the formulation of an optimal margarine product. Another application has been the formulation of mayonnaise (Bender et al. 1982).

An interesting problem in the production of champagne was reported by Hruby and Panton (1993). In one of two methods used to produce

champagne in Australia, a base wine called tirage is allowed to ferment and mature in a bottle for as long as 6 months. The tirage is then transferred to a tank for 2 weeks where it is further processed. Finally, fresh bottles are filled with finished product and stored for 6 weeks. Uncertainties in consumer demand and constraints on production stretched a 9 month process into 12 months. In addition, inventories of maturing and finished product were far too high. The problem was solved when a time-staged linear-programming model was used to smooth production and reduce stock levels.

The feasibility of planning menus by computer was generally established in the early 1960s (Balintfy and Blackburn 1964), as was the feasibility of computerized menu analysis (Brisbane 1964). In these models, nutritional requirements were provided at lowest cost. Developing models which meets sensory objectives as well as nutritional requirements has proved to be a much more difficult problem.

Renaud and Yacout (1996) reported on a company that processes lobster primarily for foreign markets. With increasing international competition, stricter standards and decreasing annual volume of lobster catches, the company wanted to know the optimal product mix to maximize profit. Linear programming was used to generate five scenarios that encompassed different possibilities available to management.

See

- ▶ Linear Programming
- ▶ Natural Resources
- ▶ Stigler's Diet Problem
- ▶ Vehicle Routing

References

Balintfy, J. L., & Blackburn, C. R. (1964). From New Orleans: A significant advance in hospital menu planning by computer. *Institutions Magazine, 55*(1), 54.

Bender, F. E., Kramer, A., & Kahan, G. (1976). *Systems analysis for the food industry*. Westport, Connecticut: AVI Publication.

Bender, F. E., Kramer, A., & Kahan, G. (1982). Linear programming and its application in the food industry. *Food Technology, 36*(7), 94.

Beneke, R. R., & Winterboer, R. D. (1973). *Linear programming applications to agriculture*. Ames: Iowa State Press.

Brisbane, H. M. (1964). Computing menu nutrients by data processing. *Journal of the American Dietetic Association, 44*, 453.

Cavins, J. F., Inglett, G. E., & Wall, J. S. (1972). Linear programming controls amino acid balance in food formulation. *Food Technology, 26*(6), 46.

Dano, S. (1974). *Linear programming in industry* (4th ed.). New York: Springer.

Glen, J. J. (1980). A Mathematical programming approach to beef feedlot optimization. *Management Science, 26*, 524–535.

Hruby, H. F., & Panton, D. M. (1993). Scheduling transfer champagne production. *International Journal of Management Science, 21*, 691–697.

Hsu, H. W., Satterlee, L. D., & Kendrick, J. G. (1977a). Experimental design: Computer blending predetermines properties of protein foods, part I. *Food Product Development, 11*(7), 52.

Hsu, H. W., Satterlee, L. D., & Kendrick, J. G. (1977b). Results and discussion: Computer blending predetermines properties of protein foods, part II. *Food Product Development, 11*(8), 70.

IBM. (1964). *Linear programming — ice cream blending*. White Plains, NY: IBM Technical Publications Department.

IBM. (1966). *Linear programming — meat blending*. White Plains, NY: IBM Technical Publications Department.

Inglett, G. E., Cavins, J. F., Kwokek, W. F., & Wall, J. S. (1969). Using a computer to optimize cereal based food composition. *Cereal Science Today, 14*(3), 69.

Judge, G. G., Day, R., JohnsonSR, R. G., & Martin, L. R. (1977). *A survey of agricultural economics literature* (Quantitative methods in agricultural economics, 1940s to 1970s, Vol. 2). Minneapolis: University of Minnesota Press.

Killen, L., & Keane, M. (1978). A linear programming model of seasonality in milk production. *Journal of Operational Research Society, 29*, 625–631.

MacKenzie, D. S. (1964). *Prepared meat product manufacturing*. Chicago: AMI Center for Continuing Education, American Meat Institute.

Nicklin, S. H. (1979). The use of linear programming in food product formulations. *Food Technology in New Zealand, 14*(6), 2.

Renaud, I., & Yacout, S. (1996). Resource allocation and optimal product mix at a fish and seafood processing company. *Computers and Industrial Engineering, 31*, 355–358.

Roush, W. B., Stock, R. H., Cravener, T. L., & D'Alfonso, T. H. (1994). Using chance-constrained programming for animal feed formulation at Agway. *Interfaces, 24*(2), 53–58.

Rust, R. E. (1976). *Sausage and processed meats manufacturing*. Washington, DC: AMI Center for Continuing Education. American Meat Institute.

Singh, R. V., et al. (1979). Least cost ice-cream mix formulation: A linear programming approach. *Agricultural Situation in India, 33*(1), 7.

Skinner, R. H., et al. (1969). Food Industry applications of linear programming. *Food Manufacturing, 44*(10), 35.

Upcraft, M. J., et al. (1989). A mixed linear programme for short-term irrigation scheduling. *Journal of Operational Research Society, 40*, 923–931.

Weske, R. (1981). *Criteria of food acceptance* (6th ed.). Zurich: Forster-Verlag.

AHP

▶ Analytic Hierarchy Process

AI

▶ Artificial Intelligence

Air Force Operations Analysis

▶ Air Force Operations Research

Air Force Operations Research

Clayton J. Thomas[1], Robert S. Sheldon[2] and Mark A. Gallagher[3]
[1]Air Force Studies and Analyses Agency (AFSAA), Washington, DC, USA
[2]Military Operations Research Society (MORS), Alexandria, VA, USA
[3]Air Force Studies and Analyses, Assessments, and Lessons Learned, Washington, DC, USA

Introduction

Air Force Operations Analysis (OA), as military operations research (OR) was often termed in the Air Force, began in the Army Air Forces in World War II (WWII). After the war, the Air Force became a separate service in 1947, and the service leaders decided to continue operations analysis sections in the headquarters and major commands. The OA office in Air Force Headquarters led procedures for steady state systems of analyst recruitment, training, rotation, etc., through its Air Force Regulation AFR 20-7. This office regulated the OA program until 1971, when it was merged into the Air Force Studies and Analyses office (which has had several titles and organizational settings since its creation in the mid-1960s). In 1993, the Air Force created a Directorate of Modeling, Simulation, and Analysis with the Air Force Studies and Analyses Agency (AFSAA) serving as its field operating agency. The directorate was expanded in 1997 to the Directorate of Command and Control that included an Associate Director for Modeling, Simulation, and Analysis with AFSAA continuing as a Field Operating Agency. In 2001, AFSAA became a direct reporting unit to the Vice Chief of Staff of the Air Force (VCSAF). This agency was converted into an Air Staff function of AF/A9, Studies and Analyses, Assessments, and Lessons Learned, in 2006, reporting as a directorate organization to the Chief of Staff of the Air Force (CSAF). Each of the major commands also established an A9 office overseeing their studies and analyses, assessments, and lessons learned and as a focal point for OR. The Air Force has effectively returned to organizing analysis similar to the way it was during and after WWII.

World War II in the 1940s

Brothers (1951) states, in WWII, 245 analysts (professional personnel, not including clerical and administrative staffs) had been in the OA program at one time or another with the peak strength having been 175. These analysts were distributed over 26 OA sections, one with every combat Numbered Air Force plus several with other overseas Air Force headquarters and several with Air Force training establishments in the continental U.S. Brothers (1951) reports there were many types of studies:

> . . . offensive ones dealing with bombing accuracy, weapons effectiveness, and target damage. . .defensive ones dealing with defensive formations of bombers, battle damage and losses of our aircraft, and air defense of our bases . . . studies of cruise control procedures, maintenance facilities and procedures, accidents, in-flight feeding and comfort of crews, possibility of growing vegetables on South Pacific islands, and a host of others. The first and largest of the OA sections was that at the Eighth Air Force.

McArthur (1990) gives a detailed account of its work and much information about the analysts, with emphasis on the mathematicians. In its foreword, Hugh Miser notes:

> During the two and a half years of existence of the Eighth Air Force section, forty-eight persons with scientific and technical training were involved, representing more than

> a dozen specialties; mathematicians were the largest subgroup, with fifteen persons, thirteen of whom stayed with the section for six months or more. . . . It should be noted that the mathematicians were functioning, not just in a mathematical role, but as scientists, developing theories about actual phenomena and applying them to problems of operations, policy, and plans.

Brothers (1954) gives an account of the well-known improvement in bombing accuracy to which these analysts contributed. The commanding general had asked, "How can I put twice as many bombs on my targets?" In 1942, less than 15% of the bombs dropped fell within 1,000 ft of the aiming point. The rate improved gradually and within two years had reached 60%. Some of the analytical recommendations that played a part in this were the nearly simultaneous release of their bombs by all the bombardiers (instead of the practice of each bombardier aiming and releasing his own bombs), the salvoing of bombs instead of presetting them to release in a string, and the decrease in the number of aircraft per formation from a range of 18–36 to a range of 12–14. The successful work of this first section made other Army Air Forces commands aware of the OA concept and led to the establishment of the other OA sections. Those sections also had their successes, all of which led to the postwar continuation of OA in the Air Force. In the forward of *Operations Analysis in WWII* (United States Army Air Forces 1948), General Carl Spaatz, the Commanding General and later first Chief of Staff of the Air Force, takes credit for requesting and establishing the first OR section while he commanded the Eighth Air Force. With the war's end, most of the analysts returned to universities, laboratories, or other civilian pursuits. Brothers (1951) reports that by January 1946 there were only a dozen left, about half of whom were finishing final reports.

Post-World War II and the Korean War in the 1950s

Brothers (1951) recalls that the United States Air Force (USAF), having decided to establish a peacetime OA program, also decided on the basis of wartime experience that it needed an analysis unit in the headquarters. This unit had two functions: to furnish scientific assistance to the Air Staff, and to serve as a focal point in the Air Force-wide OA organization. AFR 20-7 established the OA Division in Headquarters, USAF, and authorized Air Force commanders to establish OA offices in their commands, getting needed help from the Headquarters OA office.

From the OA low point of January 1946, it had grown by mid-1951 to ten offices in field commands plus the headquarters office. As a stable postwar program was established, the number of analysts grew. By 1951 there were 70 assigned, with 95 authorized. The 95 authorized professional positions were mostly civilian (under Civil Service), as at that time there were few uniformed analysts available. In addition, the RAND Corporation's work emphasized problems of the far future, freeing the OA offices to work primarily on current and near-future problems. However, when analysts were needed in the Korean War, some came from RAND (and a smaller think-tank also), as well as from OA.

Any history of OR, particularly an Air Force one, must highlight Dr. George B. Dantzig's role as "the father of linear programming (LP) and the inventor of the simplex method" and "arguably one of the most influential mathematicians of the twentieth century" (Cottle et al. 2007). WWII interrupted Dantzig's doctoral program studies at the University of California (Berkeley) when he went to support the Army Air Forces' Combat Analysis Branch of Statistical Control, Pentagon. He developed a reporting system with which combat units were able to record the number of sorties flown, aircraft lost and damaged, bombs dropped, and targets attacked; he also became familiar with the Air Force concepts of program planning of interrelated activities, ideas that would later help him structure the basic form of the LP model. The War Department awarded him the Exceptional Civilian Service Medal for his accomplishments. In 1946, Dantzig returned to Berkeley to complete his Ph.D. in mathematics. He then accepted the position as mathematical advisor to the comptroller of the newly established Department of the Air Force. Here he worked on formulating mathematical models of the Air Force's program planning process that led to the first LP models (and the use of the word *programming* in LP). And, most important, while working for the Air Force, Dantzig developed the simplex algorithm for solving

LP problems. Dongarra and Sullivan (2000) include the simplex algorithm as one of the top ten algorithms developed in the twentieth century.

In June of 1947, the Air Force established a major task force to work on the high-speed computation of its planning process, later named Project SCOOP (Scientific Computation of Optimal Programs), under the direction of the economist Marshall K. Wood, with Dantzig as chief mathematician. Project SCOOP was the setting in which the LP structure and the simplex method were proved and introduced to the world. The SCOOP civil service staff of mathematicians, statisticians, and computational experts was responsible for formulating and solving a wide range of Air Force planning and programming problems, as well as installing, in June 1952, the first computer in the Pentagon, a UNIVAC-I for solving Air Force problems. SCOOP also funded the construction and use of the 1950 National Bureau of Standards Eastern Automatic Computer (SEAC), as well as supporting academic researchers who helped to bring the application of LP to industry and business (Gass 2002).

In 1952, Dantzig transitioned from the Air Force to RAND. He joined the Berkeley faculty in 1960 and then Stanford in 1966. President Ford awarded him the 1975 President's National Medal of Science. Dantzig was a founder of OR and made further significant contributions to the field during his career.

By the mid-1950s, the headquarters OA office had 25 professional positions divided among five teams. Two of the teams were primarily concerned with implications of new types of weapons: one with atomic and nuclear weapons, and one with ballistic and cruise missiles. A third team dealt primarily with deriving information about combat operations from tests, exercises, etc. A fourth team integrated inputs from the previous three teams to use in assisting Air Staff planners. The fifth team maintained liaison with the existing field OA offices and helped commanders who wished to establish new field offices where they did not yet exist.

The field OA offices were organized according to the same general principles. There should be analysts available to study combat operations and related problems, as well as others with understanding of new technology and its implications for new weapons. Most of the growth in the OA program at that time came through the establishment of new offices, rather than the enlargement of existing offices.

Force Structure Analysis and Vietnam in the 1960s

In the 1960s, the situation began to change markedly, through the combination of two developments. One came fairly abruptly when the Secretary of Defense McNamara of the Kennedy administration institutionalized systems analysis (used to denote OR on broad systems problems) in the Office of the Secretary of Defense. Many of the RAND analysts became McNamara's whiz kids. Their efforts greatly increased the demand for cost-effectiveness studies from the military services. The other development came throughout the decade as the increase in computer hardware and software capabilities led to great increases in the development, size, and use of computer simulation models.

The headquarters OA office was caught up in both of the above trends, which made it more difficult to devote as much effort as desired to the analysis of operations in Vietnam. A small group of OR professionals worked at the Seventh Air Force Headquarters in Vietnam. This group was involved in both the day-to-day operations of the command and in longer term analyses. Analysts presented daily briefings containing trend analysis, principally truck kill projections, to the Director of Operations and the Commander. They supplemented these with weekly trend analyses to assist decision making for the next week. These analyses investigated truck kill claims and battle damage assessment. In 1970, the air sortie debrief reports were incorporated into Southeast Asia Database (SEADAB) to support better analysis. The principal OR tool used was regression analysis to project future results. This analysis cell also conducted special-purpose studies and explored subjects like subsystem effectiveness, such as the Black Crow, which was a highly sensitive passive sensor deployed on AC-130s that could detect North Vietnamese trucks hidden under the dense jungle canopy along the Ho Chi Minh trail. Finally, the office compiled a comprehensive history of the Southeast Asia war in an annual report termed Commando Hunt.

In the mid-1960s, a new and larger organization of Studies and Analyses was formed in the Pentagon incorporating an office that had been set up in the late 1950s to operate a large (for that time period) computer simulation model. It had been difficult to acquire the data and manpower to make effective use

of that model, and the resources of that office became available to staff the new office created to meet the growing need for cost-effectiveness studies. The newer office of Studies and Analyses and the smaller headquarters OA office (about 35 professionals at that time) both reported at high levels, required the same kind of competent analysts, and used OR techniques. These similarities suggested the merger of the smaller OA headquarters office into the larger office. It was finally accomplished in the first 6 months of 1971. The Studies and Analyses office chose not to continue implementation of AFR 20-7. The immediate consequences were not striking. The field OA offices continued, though a few made slight changes in name. Most of the other trends noted above continued, or even accelerated. There was a proliferation of computer simulation models and of their use in large studies.

The Air Force Studies and Analyses Agency (AFSAA) served informally as an OR research focal point. Technical exchanges across the Air Force continued in the course of business and at meetings of professional societies. Initially, the Air Force analysts held a semi-annual OA technical symposia, however they discontinued these as they made increasing use of the multiservice classified symposia of the Military Operations Research Society (MORS). The Air Force was one of the founding organizational sponsors when MORS was incoporated in 1966. While continuing to participate in MORS, in 1994, the Air Force reinstituted conducting internal technical exchanges with their annual Air Force Operations Research Sympoisium (AFORS); in 2009, this name changed to the Air Force Analyses, Assessment, and Lessons Learned (A2L2) Symposium.

The Cold War in the 1970s and 1980s

Through the 1970s and 1980s, concern related to the Cold War dominated in spite of the hot war in Vietnam. The Air Force Analysis community responded. Three main organizations focused extensive resources on nuclear warfare analyses; Headquarters Strategic Air Command (SAC), AFSAA, and the Joint Staff J8. After WWII, General LeMay had recruited to SAC preeminent operations researchers. The Headquarters at Offutt Air Force Base maintained a centralized civilian analysis organization along with military analysis shops in each of the functional areas. AFSAA dedicated a third of its analysts to evaluating nuclear war. These three analysis offices annually conducted and compared detailed plans of potential Soviet massive nuclear attacks on the U.S. and planned response options. The predominant approaches were LP and discrete-event simulations. These studies provided the foundation for force structure decisions including requirements and acquisitions.

At the Air Staff, the bulk of the studies dealt with future weapon systems and future force posture. The occurrance of many very highly classified studies of black systems began and continues today. Many studies evaluate weapon systems exploiting the latest technology. The difference in emphasis between RAND and the in-house Air Force analytical offices that had prevailed in the 1950s diminished, to a large extent because of the impact of the institutionalization of systems analysis in the Department of Defense (DoD). Lt Gen Glenn Kent (2008), who led AFSAA and later worked at RAND, summarizes several of the analytical approaches in his analytical memoir. The primary war in this period remained the Cold War, until, suddenly, it was won.

During this period, the Air Force began educating personnel in military OR. The United States Air Force Academy offered OR as an undergraduate major beginning in 1978 and has conferred 1,031 degrees to OR majors through 2010. The Air Force Institute of Technology started conferring this specialty for master's degrees in 1973 and doctorates in 1992. Through 2010, they have conferred 1,107 master's in OR and closely related programs and 43 doctorates in OR.

The total number of Air Force analysts generally continued to increase, at a somewhat slower rate, through the mid-1980s. In 1988, an Air Force personnel database showed 476 civilian analysts in the OR analyst career series. With the end of the Cold War in the late 1980s, there began a general decrease in the size of the DoD, including military OR. By the end of 1993 Air Force civilian levels in career series relevant to analysis were about 20% lower than 1988 levels through the 1990s. For military analysts, in 1986, the Air Force had 1,626 military scientists with approximately 60% OR analysts. After the turn of the century, the number of analysts began to increase. The Air Force had

increased to 566 civilian OR analysts in 2010. The Air Force completed 2010 with 499 military OR analysts.

The Middle East Wars in the 1990s through 2010

The end of the Cold War resulted in a significant drawdown in the Air Force, as the federal government reallocated resources to other concerns. This reduction in defense spending was commonly referred to as the peace dividend. The end of the Cold War resulted in an Air Force shift from a primary emphasis on strategic bombing to fighter operations. In 1992, SAC was disestablished and a joint US Strategic Command was established at Offutt Air Force Base. This Combatant Command continues to have a stronger presence and reliance on OR analysts than the other combatant commands. With the Air Force as Executive Agent, these analysts are Air Force civilians and predominately Air Force among the military members. While the major commands were realigned, the Air Staff was also reorganized. In 1991, the OR organization known as the Air Force Center for Studies and Analyses was renamed the Air Force Studies and Analyses Agency (AFSAA), which reported to the Air Staff. This alignment remained through the 1990s. The First Gulf War in 1991 extensively employed members from Air Force Studies and Analyses in build up deliberations and support for the Black Hole team.

The terror attacks on September 11, 2001 accelerated the DoD expansion that had began two years prior. OR experienced more than a decade of growing influence in the Air Force. In 2001, AFSAA became a direct reporting unit to the Air Force Vice Chief of Staff (VCSAF). In 2006, the OR organization became an Air Staff directorate, designated AF/A9, reporting to the Chief of Staff of the Air Force (CSAF). The other services and joint organizations aligned their analysis center under their planning, programming, and budgeting organizations. The equal status within the Air Staff enabled supporting a wide range of decisions across manpower, operations, planning, and resources. The major commands also each established A9 organizations to lead their analysts. Furthermore, combat analysts were deployed into the staffs fighting the wars in Iraq and Afghanistan. These developments have resulted in widespread recognition among the leadership of the role and contributions of OR in supporting their decisions.

In 1993, under Defense Secretary Les Aspin, the DoD completed the Bottom Up Review to adjust the National Defense Strategy following the end of the Cold War. Congress decided to mandate these episodic reports, which became the Quadrennial Defense Reviews (QDRs). The Air Force contributed significant force structure analyses in support of the QDRs of 1993, 1996, 2000, 2006, and 2010. The differences between the Services, Joint Staff, and OSD led to the formulation of standard scenario and campaign model inputs in the Analytical Agenda, which began in 2006. In 2010, the name was changed to Support for Strategic Analysis. The Air Force developed Synthetic Theater Operations Research Model (STORM), a discrete-event simulation of about 2 million lines of C code, has become the standard campaign model for the Air Force, Navy, and Marines by 2010.

Concluding Remarks

The end of the Cold War also precipitated a significant shift in force planning and, consequently, the analysis needed to support force development. Force planning guidance based on a single, peer threat gave way to a new paradigm comprised of multiple, often concurrent commitments. These engagements might occur anywhere on the globe at any time and range in intensity from limited humanitarian operations to full-scale theater warfare. Inherent uncertainty concerning time, place, and adversary led to production and regular revision of many planning scenarios. This evolution of force planning to reflect the new geo-political environment placed new demands on force development analysis. Years of analyzing a single, peer threat produced a large, complex hierarchy of analysis tools. While these tools provided comprehensive, high-fidelity analysis of individual campaigns, they are extremely data- and labor-intensive. The development, or even modification, of a scenario in these tools is a costly and time-consuming process extending far beyond department planning, programming, and budgetary cycle timelines. The analysis community is thus

faced with a difficult analytic challenge. They need to examine many, disparate planning scenarios to hedge against future uncertainty; their primary tools are inflexible on the timescale required by decision processes.

A resolution to this dilemma may be found in the development of new analytic tools guided by a conceptual model for addressing future uncertainty. With a less concrete vision of future conflict, optimal force design for a single scenario is less meaningful. Forces must instead be designed for robustness against unexpected contingencies and with enough inherent flexibility to allow for adaptation to impending shifts in the geo-political environment. Potential force constructs must be tested in a wide variety of scenarios informed by a multi-perspective approach to force development, recognizing no single approach provides a complete solution. Multi-scale tools must be developed to provide variable scope and fidelity necessary to address different problems. The tools must be agile enough to enable testing within decision-relevant time spans. This approach harkens back to the roots of OR: teams of scientists and operators cooperating to examine a variety of problems from the perspective, and at the levels of detail and complexity required, by the situation at hand.

See

- ▶ Battle Modeling
- ▶ Cost Analysis
- ▶ Cost-Effectiveness Analysis
- ▶ Military Operations Research
- ▶ RAND Corporation
- ▶ Systems Analysis

References

Brothers, L. A. (1951). *Development of operations analysis*. Working Paper 17.1.4, Operations Analysis Division, Directorate of Operations. Washington, DC: Headquarters, United States Air Force.

Brothers, L. A. (1954). Operations analysis in the United States Air force. *Opns Res, 2*, 1–16.

Cottle, R., Johnson, E., & Wets, R. (2007). George B. Dantzig (1914–2005). *Notices AMS, 54*(3), 343–362.

Dongarra, J., & Sullivan, F. (2000). Top ten algorithms of the century. *Comput Sci Engin, 2*(1), 22–23.

Gass, S. I. (2002). The first linear-programming shoppe. *Opns Res, 50*(1), 61–68.

Kent, G. A., Ochmanek, D., Spirtas, M., & Pirnie, B. R. (2008). *Thinking About America's Defense*. Santa Monica, CA: RAND Corporation.

McArthur, C. W. (1990). *Operations analysis in the U.S. army eighth air force in World war II*: *Vol. 4. History of mathematics*. Providence, Rhode Island: American Mathematical Society.

United States Army Air Forces. (1948). *Operations analysis in World War II*. Alabama: Air University Library, Maxwell Air Force Base.

Air Traffic Management

Michael Ball
University of Maryland, College Park, MD, USA

Introduction

The world-wide air transportation system represents an essential component in the operation of the various national economies. Many businesses would degrade to the point of insolvency without air transportation. Because of this high dependency, any deficiency in the performance of the air transportation system imposes a very large cost on the economies of virtually all countries. A recent study commissioned by the U.S. Federal Aviation Administration (Ball et al. 2010) estimated that the cost of air transportation delays to the US economy in 2007 to be $32.9 billion.

Air Traffic Management (ATM) is the process by which an air transportation system is managed. While individual flight operators and operators of large fleets, e.g., airlines, play a large role in ATM, large developed countries typically establish a national air navigation service provider, ANSP, that has responsibility for ATM. ANSP's traditionally are government agencies; however, recently several have adopted a privatized or semi-privatized business model. In either case, the primary systems and procedures for ATM are under the control of ANSP's. At the same time, increasingly, ATM is being viewed as a joint undertaking between the ANSP's and flight operators. Of course, there still remains a broad set of airline planning and control problems that would not be classified as ATM, e.g., fleet and crew planning problems and airline operational control problems that largely are

independent of ANSP decisions and processes. Ball et al. (2006) and Hoffman, Mukherjee and Vossen (2011) provide detailed reviews of ATM.

While the application of operations research to airline planning problems is perhaps the better known part of aviation operations research, the research community has devoted substantial attention in recent years to ATM problems. ATM problems possess particular characteristics that make them particularly challenging but also very interesting from the research perspective. Specifically,

- They are large, e.g., on there are over 50,000 daily scheduled, controlled flights in the U.S. airspace;
- Models and decision support tools should represent dynamic and stochastic elements as weather and other uncertain events greatly impact the capacity of airspace elements and decisions can be updated almost continuously;
- Distributed decision making and control strategies are used as both the ANSP's and flight operators distribute responsibilities among multiple entities;
- On the flight operator side, the decision makers represent competing economic entities so both economic and technical controls must be considered;
- The system requires an extremely high level of safety so that safety concerns can dominate the design of many solution strategies.
- Perhaps less directly related to modeling, but no less important from an implementation perspective, is the broader environment in which aviation is contained, which includes features such as a large unionized controller workforce, a high level of government control with related political influences and the need to achieve international harmonization among national ATM strategies and systems.

Air Traffic Flow Management Background

It is convenient to break down the domain of ATM into a tactical component and a strategic component. The tactical component, Air Traffic Control (ATC), is concerned with controlling individual aircraft on a time horizon ranging from seconds to 30 min for the purpose of ensuring safe separation from other aircraft and from terrain. The strategic component, Air Traffic Flow Management (ATFM) works at a more aggregate level and on a time horizon of up to about 12 h in the United States and 48 h in Europe. Its objective is to insure the efficient flow of aircraft through the airspace. It seeks to avoid congestion and delays and, when delays are unavoidable, to reduce as much as possible their overall impact. ATC problems are largely driven by safety concerns and the operating characteristics of aircraft. Thus, they largely fall within the disciplines of Control and Aeronautical Engineering. ATFM, on the other hand, is a large systems problem involving many stochastic elements and large decision spaces and thus falls within the domain of Operations Research. This chapter focuses on ATFM problems. Odoni (1987) provided an early description of operations research models for ATFM problems.

Air traffic is diverse consisting of individual flights by general aviation pilots, a variety of commercial on-demand and business jet services and the operations of large scheduled air carriers; military air traffic is another category: some military traffic is restricted to separate airspace and some is mixed with other air traffic. The large air carriers dominate the major airports and airspace. ATFM systems tend to be focussed on their needs. Specifically, air carrier schedules represent airspace demand and also serve as the basis for measuring the performance of ATFM systems. The fundamental premise of ATFM is that, roughly speaking, if all operations occurred at their scheduled times and if all airspace elements were in their normal operating states, then there would be little need for any flow management. Under such ideal conditions demand on all airspace elements would be less than capacity and operations would generally proceed as if there were no constraints. ATFM recognizes that the complexity of the airspace system and its susceptibility to weather conditions imply that it is extremely rare that nominal operating conditions exist over extended periods of time and/or large portions of the airspace.

ATFM performance is measured primarily with reference to deviations from schedules. Consequently, ATFM systems generally seek to minimize deviation between the timing of actual operations and scheduled operations. Thus, the key performance indicators usually involve measures of delay.

Before discussing ATFM decisions and controls, it is important to consider the roles and responsibilities of

the various stakeholders. The primary mission of ANSP's is to insure safety. As such they have a very strong role in the tactical (ATC) domain; further, even within ATFM their main responsibilities involve insuring airspace does not become overloaded causing unsafe situations. Over time, they have taken on a greater role in seeking to insure the overall efficient operation of the system, however, one should keep in mind that the air carriers and other flight operators will have strong (perhaps stronger) role in this area. In fact, in the past 15 years a new philosophical approach to ATFM has emerged. This new approach was originally called Collaborative Decision Making (CDM) and more recently is known as Collaborative Air Traffic Management (CATM). Its philosophical tenets include:

- Generating a better knowledge base by merging information provided by flight operators with the data that are routinely collected by an ANSP;
- Creating common situational awareness by distributing the same information to both the ANSP and the flight operators;
- Creating tools and procedures that allow flight operators to respond directly to capacity/demand imbalances and to collaborate with ANSP specialists in the formulation of flow management actions.

There are three basic types of controls used to manage air traffic:

Ground Holding: the simplest way to insure a portion of airspace does not become overloaded is to prevent certain flights from departing at the origin airports; in both the U.S. and Europe ground holding can be issued by way of multiple types of larger initiatives; the ANSP generally explicitly must give a flight operator permission before a flight can depart; a somewhat extreme version of ground holding is a ground stop, where all flights that have not yet departed and are destined for a particular airport are held on the ground until further notice.

Rerouting: a second approach to avoiding congested airspace is to fly around it; rerouting is a fundamental method used to manage congestion; the decision to change a route can occur before departure or after a flight is airborne; generally the flight operator requests a route or route change and this must be approved by the ANSP; however, it is also the case that the ANSP might indicate that only certain route options are available effectively forcing flight operator routing decisions; an extreme version of rerouting is a diversion, where a flight is redirected to an airport other than its scheduled destination.

Airborne Speed Control: once airborne, a certain amount of speed control can be exercised on flights; small changes can be implemented by simply adjusting the actual velocity of the aircraft; greater adjustments can be implemented by various types of maneuvering, e.g., vectoring, which involves flying in an indirect zig-zag pattern, and airborne holding, which involve flying in circular or oval patterns so as to remain in a designated area until progress toward the destination is allowed. Typically a flight operator chooses the speed of its flights to optimize business objectives; speed controls might subsequently be put in place by the ANSP.

These three classes of controls broadly represent the decision space of the ANSP (although as indicated each type of action requires some type of collaborative decision on the part of the flight operator). Another very important control is the decision to cancel a scheduled flight. Canceling certain flights clearly can greatly help relieve congestion but at the expense of a high degree of inconvenience imposed on certain passengers. While ANSP actions might greatly influence which flights are canceled, the final decision on a flight cancelation rests with the flight operator.

Basic Decision Problems and Optimization Models

It is useful to start with a description of three basic ATM problems and models. These problems represent basic functions that underlie higher level ATM processes. Further, at least in their simplest form the underlying models are classic operations research models.

The (Single) Flight Planning Problem

At the heart of a large, complex air traffic plan are the flight plans that individual flights follow from their origin airports to their destination airports. The key elements of a flight plan include the (1) the route the flight will follow, which indicates the points on a two-dimensional map it will fly over, (2) the flight

profile, which indicates the altitude the flight uses over the course of the route, (3) the speed to use over the course of the route, and (4) the trip fuel required for the flight. A controlled flight is one for which an approved flight plan has been filed with the air traffic management (ATM) system. Airline and general aviation operators prepare and file flight plans usually based on criteria that consider each flight in isolation. Air carriers typically employ sophisticated software, including advanced route optimization programs, for this purpose. By accepting a flight plan, an ANSP agrees to take responsibility for the safe separation of that aircraft from all other controlled aircraft in the airspace and to provide many other types of assistance toward the goal of completing the flight safely and expeditiously. Practically all airline flights and a large number of general aviation flights are controlled.

Considering the flight planning problem in unconstrained, three-dimensional space, control-based methods can be brought to bear. These employ aircraft performance characteristics and the relationship among weight, fuel remaining and aircraft performance. This basic approach can have limited applicability because much of today's airspace is highly organized so that many flight plan options are limited to a relatively small number of discrete choices. Thus, dynamic programming and shortest path methods become appropriate. A variety of factors add to the problem's complexity, including the need to take into account weather conditions, restrictions on various portions of airspace, differences in overflight charges among countries to name a few. A common decomposition approach first determines a route and then optimizes the speed and profile in a second step. Solution robustness can also be of interest to account for possible changes in weather or other conditions. While this basic problem is quite challenging and spans multiple domains there is surprisingly little literature on it. Sorensen and Goka (1985) provided an early reference on the topic and Altus (2007) a more recent survey.

Arrival Sequencing Problem

In order to maintain a high level of arrival throughput into an airport one would seek to space successive aircraft as close as safely possible. There are two physical effects that determine this safe separation: one is the long-standing safety requirement that two arriving aircraft cannot occupy the same runway simultaneously and the second is based on the separation requirements related to the *wake vortex* hazard. A wake vortex is a disturbance of the air caused by one aircraft. This disturbance can cause a second trailing aircraft to become unstable and even crash. Thus, there needs to be a certain separation distance or time to allow the wake to dissipate. The required distance depends on the characteristics of the (ordered) aircraft pair, e.g., a very heavy aircraft followed by a very light aircraft is the least stable situation requiring the larger separation distance. The net result of these characteristics is that the arrival throughput in terms of rate of arriving aircraft depends on the types of aircraft and their sequence. For example, a light aircraft followed by two heavies would require less total time to land than a heavy followed by a light followed by a heavy.

Thus, a fundamental problem in air traffic management is determining and arrival sequence for a group of aircraft (Balakrishnan and Chandran 2010; Beasley et al. 2000). The simplest version could be stated as: given n airborne aircraft, all approaching a single runway, determine a sequence of landings so as to minimize the time when the last aircraft lands. Here in the most general form there would be a required separation or distance, t_{ij}, between every possible ordered aircraft pair, (i, j). This is an instance of the well-known Hamiltonian path problem, which is very closely related to the even more celebrated traveling salesman problem.

However, the classic Hamiltonian path problem is a static problem. On the other hand, the problem of sequencing aircraft on a runway is dynamic: over time, the pool of aircraft available to land changes, as some aircraft reach the runway while new aircraft join the arrivals queue. Moreover, minimizing the "latest landing time" (or maximizing "throughput") should not necessarily be the objective of optimal sequencing. Many alternative objective functions, such as minimizing the average waiting time per passenger, are just as reasonable. A further complication is that the very idea of "sequencing" runs counter to the traditional adherence of ATM systems to a first-come, first-served (FCFS) discipline, which is perceived by most as fair.

These observations have motivated a variety of research on the arrival sequencing problem with the

objective of increasing operating efficiency while ensuring that all airport users are treated equitably. It is common to assume that aircraft start in an initial sequence and so both the dynamic aspect and fairness considerations can be modeled by limiting the deviations from this sequence. Other constraints might include adherence to a landing time window for each flight and also precedence constraints. The simplest versions of this problem can be solved in polynomial time, however, more complex versions become more difficult. Nonetheless, because the amount of sequence shifting that can take place can be quite limited, the problem can be effectively solved in practice.

The Slot Assignment Problem

In the U.S. ground holding is most often implemented via a ground delay program (GDP). Here the arrival capacity at a destination airport is reduced due to poor weather and so the rate of flow into the airport is reduced by delaying flights on the ground at their origin airports. The problem can be modeled by defining a set of slots (time intervals) at the destination airport with a capacity in terms of number of arrivals for each such slot. Each flight is assigned to an arrival slot, which can be no earlier than the flight's earliest arrival time. This effectively determines an arrival delay for each flight, which is in turn converted into a departure delay (ground hold). This basic model of assigning flights to slots arises in other contexts as well, for example in the planning of airspace flow programs, which are used in the U.S. to ration the flow through a portion of the airspace.

Model inputs include: set of slot (time intervals) $t(t = 1,2,\ldots,T)$ and set of flights $f\,(f = 1,2,\ldots,F)$. Let

$$b_t = \text{the (reduced) arrival capacity of slot } t,$$

$$e(f) = \text{the earliest time slot at which flight } f \text{ can arrive},$$

$$c_{ft} = \text{the cost of assigning flight } f \text{ to arrive at time interval } t,$$

and the variables:

$$x_{ft} = 1 \text{ if flight } f \text{ is assigned to time interval } t;\ 0 \text{ otherwise.}$$

Then, the slot assignment problem can be formulated as:

Slot_Assign

$$\begin{aligned} \text{Min}: \quad & \sum_{f,t} c_{ft}x_{ft} \\ \text{s.t.} \quad & \sum_{f} x_{ft} \le b_t \quad \text{for all } t, \\ & \sum_{t \ge e(f)} x_{ft} = 1 \quad \text{for all } f, \\ & x_{ft} \ge 0 \text{ and integer} \quad \text{for all } f \text{ and } t. \end{aligned}$$

As can be seen, this is a simple transportation model that can be solved very efficiently. This model was first described in Terrab and Odoni (1993). It has been noted that the definition $c_{ft} = r_f(t - e(f))^{1+\epsilon}$ is attractive since flight delay costs tend to grow with time at a greater than linear rate. In addition, solutions produced using this objective function are attractive from the standpoint of equity or fairness (Vossen and Ball 2006a).

CDM and Resource Allocation and Exchange

The CDM effort grew out of a desire on the part of both the airlines and the FAA for improvements in the manner in which GDPs were planned and controlled (Wambsganss 1996; Vossen and Ball 2006a). The FAA and, more specifically, the air traffic control system command center (ATCSCC) had realized the need for more up-to-date information on the status of flights currently delayed due to mechanical or other problems or even canceled unbeknownst to the ATCSCC. Equally important, the success of GDPs also depends vitally on timely information regarding airline intentions with respect to flight cancelations and delays over the next few hours. At the same time, the airlines did not feel the allocation procedures used by the ATCSCC were always fair and efficient. In addition, each airline wished to gain more control over how delays were allocated among its own flights. Thus, both the airlines and the FAA had specific (although different) objectives that motivated their participation in CDM. These issues bring another dimension to research in this area involving the relationship between resource allocation methods and stakeholder incentives. These topics are more typically considered in the economics literature, e.g., under the domain of mechanism design.

$$\textit{Ration-by-Schedule} \rightleftarrows \textit{Cancelations \& Substitutions} \rightleftarrows \textit{Compression}$$
$$\Leftrightarrow$$
$$\textit{Fair slot allocation} \rightleftarrows \textit{Intra-airline slot exchange} \rightleftarrows \textit{Inter-airline slot exchange}$$

Air Traffic Management, Fig. 1 CDM resource allocation

CDM Resource Allocation

It is instructive to consider CDM slot assignment within a more general framework. Figure 1 illustrates the overall resource allocation process. The basic (initial) slot assignment performed by the ANSP (FAA in this case) employs the ration-by-schedule, RBS algorithm. Each airline, using the cancelations and substitution process, may then cancel flights and modify slot-to-flight assignments for its own flights (intra-airline exchange). Thus, although RBS, in concept, allocates slots to flights, the cancelation and substitution process effectively converts the slot-to-flight assignment into a slot-to-airline assignment. The final step, compression, which is carried out by the ANSP/FAA, maximizes slot utilization by performing an inter-airline slot exchange in order to ensure that no slot goes unused.

RBS defined below is a simple priority rule for findisng a (feasible) solution to SLOT_ASSIGN. In this description, slots are defined so that each has capacity 1, i.e., $b_t = 1$ for all t. The algorithm employs a scheduled arrival time for each flight f, $\hat{e}(f)$, which may be different from the earliest arrival time $e(f)$:

RBS:

Step 1: Order slots by increasing value of slot time (t).

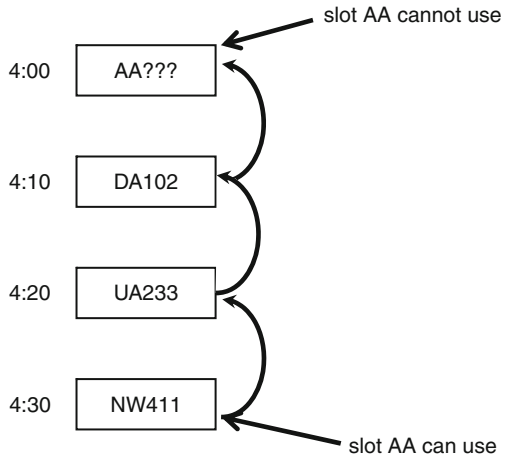
Step 2: For each slot t, choose the unassigned flight f with the earliest scheduled arrival time (value of $\hat{e}(f)$) and assign f to t, i.e., set $x_{ft} = 1$.

The fact that priority is based on scheduled arrival time $\hat{e}(f)$ as opposed to say earliest arrival time $e(f)$ is important. In general $e(f)$ can be larger than $\hat{e}(f)$ as mechanical problems or upstream delays to inbound flights can delay aircraft availability. Earlier (pre-CDM) procedures, which were based on $e(f)$, encouraged airlines to withhold or alter flight status information, as higher values of $e(f)$ led to less desirable slot allocations. By basing the allocation on $\hat{e}(f)$, the resources obtained are independent of the flight status information provided. On the other hand, by using $\hat{e}(f)$, it is quite possible that certain flight-to-slot allocations (instances where $x_{ft} = 1$) are not feasible leading to the need for the intra-airline slot exchange process; it is even possible that airlines will receive slots that are infeasible for all of their flights, which leads to the need for the inter-airline slot exchange.

Airlines can view the intra-airline slot exchange (cancelation and substitution) step as an internal resource allocation process. Specifically, the collection of slots allocated to a given airline's flights can be viewed as a set of slots owned by that airline. The airline could then view the internal problem of assigning its flights to the slots it owns as an optimization problem; again this is a transportation or assignment problem. In this case, each airline could employ its own internal cost function where the cost of the delay assigned to each flight might depend on a variety of internal (possible proprietary) factors unique to that airline. The decision variables in this problem would, in general, include the possibility of canceling certain flights.

Slot Trading Models

As indicated above, the third step of the CDM allocation process, the compression algorithm, can be viewed as a form of inter-airline trading or bartering. Compression is necessary when one or more airlines are assigned slots that they cannot use. Specifically, when a slot time t is earlier than the earliest arrival time $e(f)$ of all of an airlines' unassigned flights, then that airline cannot use the slot t. In such a case, the owner of the slot would be willing to give up the (early) slot for a later one that it can use. As illustrated in Fig. 2, the underlying exchange is not a simple one-for-one trade. Rather, there is a daisy chain of slot reassignments, which results in freeing up the earliest slot usable by the airline that has given up its slot. The compression algorithm is a batch process that identifies all unusable slots and finds for each an appropriate sequence of slot reassignments/exchanges. Research has shown that the problem of finding a set of such exchange sequences is equivalent to finding a set of arc-disjoint cycles in a directed graph (Vossen and Ball 2006a). Viewing the problem in this way leads to the consideration of alternate objective functions and the possibility of



Air Traffic Management, Fig. 2 Compression exchange: AA gives up the 4:00 slot and receives in return the 4:30 slot

including trading mechanism in more complex optimization models.

While the original CDM-based decision support system for GDP's included the batch compression process, later implementations included near-real-time trading mechanisms. The slot credit substitution mechanism is initiated by an airline that wishes to trade an earlier slot for a later one. This mechanism allows an airline to insure it will receive a slot at a particular time before freeing up an earlier slot by canceling a flight. The adaptive compression procedure is initiated by a background process run by the ANSP. The process continuously searches for slots that are in danger of going unused and executes slot exchanges in order to insure that arrival capacity does not go unused.

More recent research (Vossen and Ball 2006b) has focused on implementing more complex slot exchanges, e.g., where an airline gives up two or more slots in exchange for a similar number of other slots. While compression (and its real-time versions) are always motivated by the existence of a slot an airline cannot use, under the more complex trading models, an airline can trade multiple "usable" slots for another set of usable, but higher value, slots. Research has demonstrated that more complex trading schemes can yield an increase in the overall welfare of participating airlines. Additionally, integer programming models have been developed for identifying the best set of trades to execute.

Of course, a very logical further step in the progression of trading mechanisms would be to consider the use of monetary-based exchanges. In fact, it is quite easy to envision the inclusion of side payments with any of the trading mechanisms described above. Research has shown that adding the possibility of side payments can lead to increases in overall welfare over models that do not employ monetary payments. Recent work has developed and analyzed specific mechanisms of this type. However, implementation in practice has yet to take place.

Modeling Equity and Equity Efficiency Tradeoffs

RBS evolved out of a set of human-in-the-loop (war-gaming-like) and consensus-building exercises. It is now accepted as a de facto allocation standard within the U.S. With this background, subsequent research examined its inherent properties as a fair allocation rule. It was also then used as an equity standard in the development of models that considered equity efficiency tradeoffs.

RBS assigns to each flight f a controlled time of arrival $CTA(f)$. ($CTA(f)$ is the time of the slot to which f is assigned, i.e., the t with $x_{ft} = 1$.) The assignment of $CTA(f)$ is equivalent to assigning a delay $d(f)$ to flight f given by $d(f) = CTA(f) - \hat{e}(f)$. Assuming that flight delays, i.e., the $d(f)$ values, are all integer and defining D to be the maximum delay assigned to any flight and $a_i = |\{f : d(f) = i\}|$ for $i = 0,1, 2,\ldots, D$, then the important properties of unconstrained RBS can be defined by,

Property 1: RBS minimizes total delay $= \sum_f fd(f)$.

Property 2: RBS lexicographically minimizes $(a_D,\ldots, a_1, a_0)$. That is, a_D is minimized; subject to a_D being fixed at its minimum value, a_{D-1} is minimized; subject to (a_D, a_{D-1}) being fixed at its lexicographic minimum value, a_{D-2} is minimized; and so on.

Property 3: For any flight f, the only way to decrease a delay value, $d(f)$, set by RBS is to increase the delay value of another flight g to a value greater than $d(f)$.

Property 3, which follows directly from Property 2, expresses a very fundamental notion of equity that has been applied in a number of contexts. It is interesting to note that RBS, which was developed as a practical alternative by means consensus-building exercises, has such elegant and desirable properties. On the

other hand, this may not be surprising in that these properties may represent a large part of the basis for reaching consensus.

These properties show that unconstrained RBS produces a fair allocation. However, RBS is not used in its pure form. Rather, certain flights are exempted from RBS (and given no delay); the flights exempted are "long haul" flights greater than a certain distance away from the GDP (destination) airport. The motivation for these exemptions will be described later when stochastic models are discussed. One then naturally considers whether the exemption policies, in fact, introduce bias (in the sense that certain airlines consistently receive more delay on the average than others). Vossen et al. (2003) showed that exemptions do introduce a bias. In additions, optimization procedures have been developed for mitigating these biases. The approach taken initially computes the unconstrained RBS solution and defines it as the ideal allocation. Optimization procedures are then described that minimize the deviation of the actual allocation from the ideal. The resulting approaches maintain the exemption policies, but take into account the advantages provided to an airline by its exempted flights when allocating delays to its other flights. The principle underlying this work is that equity or fairness can be measured as the deviation from RBS. This basic concept can be applied more generally by considering equity as one objective and another performance criterion, e.g., throughput, as a second objective. One can then consider designing decision support systems that take a multi-objective approach trading off these two objectives.

Resource Allocation Based on User Supplied Priorities

The preceding set of processes can be viewed under a general paradigm in which (1) the ANSP allocates resources to flight operators, (2) each flight operator optimizes the use of the resources it owns, (3) a limited amount of inter-operator trading is supported. An alternative approach to allowing flight operator costs and priorities to be taken into account involves processes in which each flight operator provides priority or tradeoff information to the ANSP and the ANSP takes this information into account in determining resource allocation. This approach underlies the new Collaborative Trajectory Options Program (CTOP), which is now being tested by the FAA for deployment in assigning reroute options to flights competing for common resources. CTOP uses flight operator priority information in determining exactly which resource to give to each flight on a list, but determines priorities among flights based on a variant of RBS. A fundamentally different system could use flight operator inputs to determine the priority order of flights.

One such approach assigns a certain number of credits to each flight operator (Sheth and Gutierrez-Nolasco 2010). These credits constitute a type of artificial currency. Specifically, a flight operator would assign a certain number of credits to each of its flights. These credits would be used in a bidding/negotiation process when competing for access to airspace resources. Simulations and human-in-the-loop experiments have been conducted that show promise for this concept. In concept, it could address some of the tradeoffs that various types of slot trading systems would address, e.g., allowing a flight operator to gain access to an early slot for a highly valued flight by giving up priority for a lesser valued flight. In the end such systems are types of marketplaces, which implies that, in a highly competitive environment, mechanisms must be in place to determine the equivalent of a market clearing price. Research is needed to address such challenges. Another important issue is the design of a fair method for allocating credits to the various flight operators.

Large-Scale Deterministic Optimization Models

There are two broad classes of global air traffic flow models (Bertsimas and Stock Paterson 1998; Helme 1992; Lindsay et al. 1994; Sherali et al. 2002). The first assumes that the trajectory (route) of each flight is fixed and optimizes the timing of the flight's operations. The second allows the route of each flight to vary, as well. While conceptually models of the second type have a much larger decision space, recent research has produced models of the second type with computation times close to those of the first type.

The modeling approach for both cases chooses a time horizon of interest and decomposes it into a discrete set of time intervals. A geographic scope is

also selected. This determines the set of capacitated elements under consideration. Two capacitated elements might include the arrival and departure processes for an airport (or a combined arrival and departure process). Another could be a sector: a volume of airspace with a limit on the number of aircraft that can simultaneously have access. Another could be a waypoint: a point in 3-dimensional space that multiple flights seek to pass through over time. The combination of the model's temporal and geographic scope determines the set of flights to be considered.

For models of the first type, the basic decision variables specify the airspace element occupied by a flight at each time interval, i.e.,

$x_{fte} = 1$ if flight f occupies airspace element e during time interval t; 0 otherwise.

The capacity constraint associated with an element e and time interval t is of the form:

$$\sum_f x_{fte} \leq cap(t, e) \quad \text{for all } t \text{ and } e,$$

where *cap(t, e)* is the capacity of element e during time interval t. For airport arrival and departure capacities and for waypoints, $cap(t, e)$ is equal to the maximum number of flights that could flow through that element during time interval t. For a sector, it is equal to the maximum number of flights that can occupy the sector simultaneously.

The remaining constraints define temporal restrictions on the manner in which each flight can progress through the airspace. For example, they might specify that, once a flight enters a sector, it must remain in the sector for 3, 4 or 5 time intervals. In this case, 3 time intervals would correspond to traversing the sector on a direct path at maximum speed and 5 time intervals might correspond to a longer traversal time based on application of some type of speed control. Since the flight's route is an input, the progression from departure airport through a specific sequence of sectors to a destination airport is a fixed model input, as well.

Models of this type can be solved very efficiently. Of particular note is their use of an alternative set of decision variables. Specifically, the x_{fte} variables are replaced with a set, w_{fte}, defined by:

$w_{fte} = 1$ if flight f arrives at airspace element e by time interval t; 0 otherwise.

While the w variables can be obtained from the x variables through a simple linear transformation ($w_{fte} = \sum_{s=1}^{t} x_{se}$), the w variables are much easier to work with because they produce very simple and natural temporal progression constraints. Further, the associated linear programming relaxations are very strong in the sense that they lead to the fast solution of the associated integer programs (Bertsimas and Stock Paterson 1998). A variety of additional features can be included in models of this type, including the propagation of delays that occurs when a delay in the arrival of a flight causes a delay of an outbound flight that uses the same aircraft. Models can also capture the dependence between an airport's arrival and departure capacities and choose the appropriate combination of the two for each time interval.

A second type of model also allows for flight routes to vary. The most direct way to develop such models would employ variables of the form:

$x_{frte} = 1$ if flight f uses route r and occupies airspace element e during time interval t; 0 otherwise.

Since these variables involve the incorporation of an additional dimension, running times can increase dramatically. While the large-scale models of the previous type, e.g., involving thousands of flights, could be solved to optimality very quickly using commercial integer programming solvers, much smaller instances of problems of the second type could only be addressed uses approximate techniques. Bertsimas, Lulli and Odoni (2011) describe new models with route choice using variables of the first type. This was accomplished by embedding constraints within the formulations that implicitly represent the route choice options.

Stochastic Models

Uncertainty on multiple levels has led to the failure of many attempts at practical implementation of various

air traffic flow management models. To be effective, models must include stochastic components explicitly or they must address problems restricted to limited geographic and time domains for which available information is less subject to uncertainty. The term demand uncertainty describes the possibility that, due to random or unforeseen events, flights may deviate from their planned departure or arrival times or from their planned trajectories. Similarly, capacity uncertainty refers to the possibility that random or unforeseen events will cause changes to the maximum achievable flow rates into and out of airports or through airspace waypoints or to the maximum number of flights that can occupy simultaneously a portion of the airspace. Examples of factors contributing to demand uncertainty include problems in loading passengers onto an aircraft, mechanical problems, queues on the departure airport's surface or in the air and en route weather problems. Examples of factors contributing to capacity uncertainty include weather conditions at an airport or in the en route airspace, failures or degradation of air traffic control equipment, unavailability of air traffic control personnel, and random changes in flight sequences and aircraft mix that require alterations of anticipated flight departure or arrival spacing.

The largest body of work in this area has focused on ground holding models that explicitly take into account uncertainty in airport arrival capacity (Ball et al. 2003; Mukherjee and Hansen 2007; Richetta and Odoni 1993). As discussed in the previous section, optimization models for the ground holding problem subdivide time into an arbitrary number of discrete intervals. Typical time intervals might be 10 or 15 min or even as much as 1 h for the most aggregate models. The rate of arrivals into an airport is called the airport acceptance rate (*AAR*). A GDP is motivated by a reduction in the AAR over a period of time, e.g., four hours, usually due to poor weather. A weather forecast might be characterized by an AAR vector, which specifies the predicted AAR value for a sequence of time intervals into the future. As discussed earlier GDP's must be planned well in advance based on a weather forecast. In a typical GDP caused by a weather disturbance that moves through a region, the AAR would start at its normal level, e.g., 60 arrivals per hour, decrease to one or more degraded levels, e.g., 30 arrivals per hours, for several time intervals, e.g., 4 h, and then return to its original level. If it were known that such a scenario would occur with certainty, then a deterministic ground holding model, such as those discussed earlier, could obviously be used with this scenario providing the capacity constraints input. Of course, in general, there may be significant uncertainty associated with any single AAR vector so that ideally the AAR should be treated as a random variable. The typical stochastic programming modeling approach used takes as input several such scenarios together with associated probabilities. For example, an optimistic scenario indicating no capacity degradation would consist of a vector of hourly AARs of 60 throughout the period of interest, whereas, a more pessimistic scenario might assume that the AAR will be 30 during every hour in the period. Such an input can be characterized as:

$$D_{tq} = \text{AAR for time interval } t \text{ under scenario } q, \quad \text{for } t = 1, \ldots, T \text{ and } q = 1, \ldots, Q.$$

$$p_q = \text{probability of the occurrence of scenario } q, \quad \text{for } q = 1, \ldots, Q.$$

The x_{ft} variables are then defined as in the deterministic ground holding model, but the capacity constraints are replaced with a new set of scenario-based constraints and associated variables. The new variable set is defined by:

$$y_{tq} = \text{number of flights held air from period t to } t+1, \text{ under scenario } q, \text{ for } q = 1, \ldots, Q.$$

The new set of capacity constraints then is:

$$\sum_{f=1}^{F} x_{ft} + y_{t-1q} - y_{tq} \leq D_{tq} \text{ for } t = 1, \ldots, T \text{ and } q = 1, \ldots, Q.$$

Thus, under these constraints, there is a separate capacity for each scenario. However, the y variables allow for flow between time intervals, so the number of flights assigned to land in an interval under a particular scenario can exceed the AAR by allowing excess

flights to flow to a future time interval. Note that this set of capacity constraints defines a small network flow problem for each q, with flights flowing from earlier time intervals to later ones. To be feasible, for each given q, the total arrival capacity for the entire period of interest, $\sum_t D_{tq}$, must be at least as large as the total number of flights (F).

The objective function for the model minimizes the sum of the cost of ground delay plus the expected cost of airborne delay. It requires a parameter, c^a, defined as the cost of holding one flight in the air for one time period:

$$\text{Min}: \quad \sum_{f=1}^{F}\sum_{t=1}^{T} c_{ft}x_{ft} + \sum_{q=1}^{Q} p_q \sum_{t=1}^{T} c^a y_{t-1q}.$$

This class of models has been generalized to address both airport and airspace problems. That is, weather may also cause portions of the airspace to experience capacity reductions or to be blocked entirely. A similar approach to modeling capacity uncertainty can be used.

The models just discussed can be classified as static stochastic models. Specifically, they are two-stage models in which a stage one plan (the x variables – ground delay assignment) is developed but the cost of that plan is measured based on a random stage two process (the realization of the arrival capacity and the ensuing airborne delays). In reality, the stage one decisions can be adjusted in reaction to the random events. For example, if the weather clears early, then most, or even all, flights serving ground delay can be released early. More recent research has modeled the possible adjustment of decisions on ground delays. In general such models can lead to very large scenario trees. However, some research has taken advantage of the fairly simple scenario/capacity structure associated with many GDP plans. Specifically, in many cases there is an onset of bad weather which reduces the AAR from its nominal value, say D^N, to a reduced value, say D^R. Further, it is assumed that the decision to release flights from their assigned ground delay cannot be made until the time at which the AAR switches from D^R to D^N, e.g., this decision will not be made based on a change in the weather forecast, only a change in the actual weather. While this latter assumption might seem like a strong one, it is, in fact, how the system in the U.S. operates today. With this model of the stochastics of the AAR and this dynamic decision regime, a relative simple scenario tree and compact integer program result. Effectively, this dynamic, stochastic model of GDP planning can be cast as a two stage stochastic integer program where the second stage decision variable is the time the weather clears.

Models of this type have been able to formalize the rationale behind giving priority to long-haul flights. As discussed earlier, during GDP's, certain flights are exempt and given no ground delay. The principal set of exempt flights are flights greater than a certain distance from the GDP airport. The heuristic rationale is that such flights would have to be given ground delay well in advance of their arrival at the airport and, therefore, this decision would have to be based on a forecast of weather conditions several hours into the future. Since there is uncertainty there is a (possibly large) likelihood that the ground delay would be applied unnecessarily. Instead these flights are exempt and flights closer to the airport are allowed to absorb the required ground delay if necessary. The formalization of priority based on distance leads to the ration-by-distance (*RBD*) algorithm. RBD is conceptually similar to RBS; the only difference is that Step 2 is replaced by:

Step 2: For each slot t, choose the unassigned flight f the furthest from the GDP airport (with the largest value of L_f) and assign f to t, i.e., set $x_{ft} = 1$.

Here L_f is length of flight f, i.e., the difference between its scheduled arrival time and scheduled departure time. Using a stochastic model of the type just described, i.e., where there is a distribution of possible weather clearance times and a decision to release a flight (or reduce its ground delay) is made as soon as the weather clears, it can be shown that RBD minimizes the total expected delay assigned to all flights in the GDP (Ball, Hoffman and Mukherjee 2010). RBD is not viewed as a practical algorithm since it can assign extreme delays to some short flights; a practical alternative, Equity-Based RBD (*ERBD*), uses the RBD priority principle but ensures the delay assigned to each flight does not deviate beyond a specified limit from the delay RBS would assign.

Stochastic integer programs based on models of this type have also been developed to analyze enroute problems. Since these models have the complexity of 2-stage stochastic programs they are reasonably tractable, while representing what is, from a practical perspective, a multi-stage, dynamic problem.

Concluding Remarks

Air traffic management is of vital importance to the smooth operation of the world-wide economy. The underlying essential problems have a strong systems nature, are large and subject to uncertainty. Moreover, decision-making must dynamically adjust to changing conditions and be distributed throughout multiple diverse organizations. Operations research has much to offer this area.

See

► Airline Industry Operations Research

References

Altus, S. (2007). Flight planning – the forgotten field and airline operations. In *Proceedings of AGIFORS Airline Operations 2007*, available at: http://www.agifors.org/studygrp/opsctl/2007.

Balakrishnan, H., & Chandran, B. (2010). Algorithms for scheduling runway operations under constrained position shifting. *Operations Research, 58*, 1650–1655.

Ball, M.O., Barnhart, C., Dresner, M., Hansen, M., Neels, K., Odoni, A., Peterson, E., Sherry, L., Trani, A., Zou, B (2010). *Total delay impact study: a comprehensive assessment of the costs and impacts of flight delay in the United States*, NEXTOR Report, October, 2010, available at: http://www.nextor.org/rep2010.html.

Ball, M., Barnhart, C., Nemhauser, G., & Odoni, A (2006). Air transportation: Irregular operations and control. In C. Barnhart & G. Laporte (Eds.), *Handbook of operations research and management science: Transportation*, Amsterdam: Elsevier.

Ball, M. O., Hoffman, R., & Mukherjee, A. (2010). Ground delay program planning under uncertainty based on the ration-by-distance principle. *Transportation Science, 44*, 1–14.

Ball, M. O., Hoffman, R., Odoni, A., & Rifkin, R. (2003). A stochastic integer program with dual network structure and its application to the ground-holding problem. *Operations Research, 51*, 167–171.

Beasley, J. E., Krishnamoorthy, M., Sharaiha, Y. M., & Abramson, D. (2000). Scheduling aircraft landings – the static case. *Transportation Science, 34*, 180197.

Bertsimas, D., Lulli, G., Odoni, A. (2011). An integer optimization approach to large-scale air traffic flow management. *Operations Research*, *59*, 211–227.

Bertsimas, D., & Stock Paterson, S. (1998). The air traffic flow management problem with en route capacities. *Operations Research, 46*, 406–422.

Helme, M. P. (1992). Reducing air traffic delay in a space-time network. *IEEE International Conference on Systems, Man, and Cybernetics, 1*, 236–242.

Hoffman, R., Mukherjee, A., & Vossen, T. W. M. (2011). Air traffic flow management. In C. Barnhart & B. Smith (Eds.), Quantitative problem solving methods in the airline industry: A modeling methodology handbook. International Series on Operations Research and Management Sciences. Norwell: Springer.

Lindsay, K., Boyd, E., & Burlingame, R. (1994). Traffic flow management modeling with the time assignment model. *Air Traffic Control Quarterly, 1*, 255–276.

Mukherjee, A., & Hansen, M. (2007). Dynamic stochastic model for a single airport ground-holding problem. *Transportation Science, 41*, 444–456.

Odoni, A. (1987). The flow management problem in air traffic control. In A. R. Odoni, L. Bianco, & G. Szego (Eds.), *Flow control of congested networks* (pp. 269–288). Berlin: Springer.

Richetta, O., & Odoni, A. (1993). Solving optimally the static ground holding policy problem in air traffic control. *Transportation Science, 24*, 228–238.

Sherali, H. D., Smith, J. C., & Trani, A. A. (2002). An airspace planning model for selecting flight-plans under workload, safety, and equity considerations. *Transportation Science, 36*, 378–397.

Sheth, K., & Gutierrez-Nolasco, S. (2010). Analysis of factors for incorporating user preferences in air traffic management: a system perspective. In *Proceedings of ICAS 2010, 27th International Conference on the Aeronautical Sciences*, availabe at: http://www.icas.org/ICAS-ARCHIVE/ICAS2010/index.html.

Sorensen, J.A., & Goka, T. (1985). Design of an advanced flight planning system. *American Control Conference*, 663–668, available at: http://ieeexplore.ieee.org/xpl/mostRecentIssuejsp?punumber=4788560.

Terrab, M., & Odoni, A. R. (1993). Strategic flow management for air traffic control. *Operations Research, 41*, 138–152.

Vossen, T., & Ball, M. (2006a). Optimization and mediated bartering models for ground delay programs. *Naval Research Logistics, 53*, 75–90.

Vossen, T., & Ball, M. (2006b). Slot trading opportunities in collaborative ground delay programs. *Transportation Science, 40*, 29–43.

Vossen, T., Ball, M. O., Hoffman, R., & Wambsganss, M. (2003). A general approach to equity in traffic flow management and its application to mitigating exemption bias in ground delay programs. *Air Traffic Control Quarterly, 11*, 277–292.

Wambsganss, M. (1996). Collaborative decision making through dynamic information transfer. *Air Traffic Control Quarterly, 4*, 107–123.

Airline Industry Operations Research

Michael D. D. Clarke[1] and David M. Ryan[2]
[1]Sabre Research, Southlake, TX, USA
[2]The University of Auckland, Auckland, New Zealand

Introduction

The dramatic growth of the airline industry over the past thirty years into a highly competitive world-wide transport network has been accompanied by the extensive use of operations research and management science methodology in all areas of airline operations. All airlines make major investments in sophisticated aircraft and employ highly trained and skilled staff. Efficient utilization of such valuable resources is clearly an important objective in the management of a profitable airline.

In 1960, the airline industry recognized the potential benefits of OR/MS by setting up the Airline Group of the International Federation of Operations Research Societies (AGIFORS) as a special interest group. Since that time, annual AGIFORS symposia have been held and the proceedings of these meetings provide excellent documentation of the many applications and problems which have been addressed by the use of OR/MS techniques (Richter 1989). A comprehensive discussion of the direction of OR/MS applications in the airline industry by the late 1980s was given by Teodorovic (1988), and a special issue of *Interfaces* edited by Cook (1989) presented six specific OR/MS airline-industry case studies.

Over the years since then, an extensive range of practical problems involving long-term planning, short-term planning and "day of operation" decision making have been considered and the full range of methods and techniques including forecasting, simulation, heuristics and optimization have been used to provide practical solutions and decision support systems. In particular, such methods as set partitioning and set covering optimization have been widely applied in many airline scheduling problems. In recent years, linear optimization models generated from airline applications have stimulated much research into the development of interior point and improved simplex methods for solving such problems. The following broad application areas of OR/MS in the airline industry can be clearly identified and will be discussed in further detail:

- Flight Scheduling Planning
- Fleet Assignment
- Yield Management
- Crew Scheduling
- Aircraft Maintenance Routing
- Schedule Recovery

Flight Scheduling Planning

The design of a flight schedule is probably the most important and fundamental task for any airline. The schedule which determines the frequency and departure times of flights between airports served by the airline is usually prepared and published many months before it is due to be operated. The preparation of the schedule must take into account forecast passenger demand, the operational limitations of both aircraft and crews, and the access limitations imposed by airports either due to meteorological conditions, airport congestion, operational restricted hours or differential landing tariffs. Besides many constraints on the form of feasible flight schedules, there is also considerable variation in the choice of objective ranging from maximizing profit, maximizing passenger-kilometers, maximizing load factors, minimizing the number of aircraft and minimizing direct and indirect operating costs. A discussion of this problem was given by Soumis and Nagurney (1993).

Two particular forms of schedule also reflect the airlines' mode of operation as either a network or a hub-and-spoke operation. Airlines operating hub-and-spoke systems design schedules that bring many aircraft into a hub airport within a short space of time, thus enabling passengers to transfer to another aircraft before all of the aircraft then depart (on the spokes) from the hub over a short period of time. In both forms of operation, the airline schedule can be represented as a network problem in which one must determine conserved flows of aircraft between ports at times chosen to satisfy operational constraints and optimize a specified objective. Because of the enormous combinatorial complexity of the network model, many heuristic methods have been developed to assist airline schedule planners. The problem continues to motivate the development of improved optimization methods.

Fleet Assignment

Given a predetermined flight schedule, the fleet assignment problem is to determine which aircraft type is assigned to a given flight segment in the carrier's network. The goal of the fleet assignment problem is to assign as many candidate flight-segments as possible in a schedule pattern to specific aircraft types, based on such factors as operating costs, revenues, and operational constraints and capabilities. The problem is formulated and solved as an integer-programming model that permits the assignment of multiple fleet types to a flight schedule simultaneously. The fleet assignment model can be classified as a large multicommodity flow problem with side constraints defined on a time-space network. One of the earliest published articles on the topic of fleet assignment was presented by Abara (1989), who discussed the application of integer linear programming to the fleet assignment problem, and explained how this technique was already being used extensively throughout American Airlines.

Subramanian et al. (1994) presented a solution procedure referred to as Coldstart, which is a fleet assignment methodology developed by Delta Airlines. Coldstart minimizes a combination of operating and passenger spill costs, subject to operational constraints. Hane et al. (1995) outlined a model for the fleet assignment problem and discuss solution problems that often exist with such large problems including degeneracy, that leads to poor performance of standard LP solution techniques. The solution methodology presented incorporates an interior point algorithm, cost perturbation, model aggregation, branching on set-partitioning constraints, and prioritizing the order of branching, in an effort to develop more efficient solution procedures for the problem. Clarke et al. (1996) discussed maintenance and crew considerations in the basic daily fleet assignment problem of Hane et al. (1995), and implementation issues related to its solvability. The solution methodology presented involves the use of the dual steepest edge simplex method and solving the mixed integer problem by branch and bound.

The most recent advances in the fleet assignment problem have included the development of origin-destination based models that incorporate passenger flow issues more accurately into the decision model. These are able to control passenger mix and reflect the upstream and downstream effects of the fleet assignment decision. Jacobs (1999) and Kniker (1998) independently discussed the origin-destination fleet assignment problem.

Yield Management

The yield management process maximizes revenue by selectively accepting and rejecting reservation requests based on its relative value. Excellent reviews with good bibliographies are provided in Weatherford and Bodily (1992) and McGill and van Ryzin (1999). The procedure is designed to manage perishable seat inventory effectively, since an empty seat at flight departure cannot be resold. The control of reservation inventory to maximize revenues is normally accomplished through a series of sequential processes including overbooking, fare mix or discount allocation control, group control and traffic flow control. Overbooking is the process by which additional reservations beyond physical capacity (seats) are sold to compensate for the effects of cancellations, noshows, duplicate bookings and misconnects. The primary objective of discount allocation control is to limit the number of seats sold to lower valued passengers by protecting seats for late booking higher valued passengers. The optimal discount allocation controls result in the optimal onboard mix of full fare, discount, leisure and deep discount passengers to maximize the total onboard revenue. The process of yield management effectively manages the risk associate with this uncertainty and maximizes expected revenues.

Group control is done using a group evaluator that assists in deciding whether to accept or reject the group booking. The group evaluator determines the minimum acceptable fare based on the expected displacement cost of individual passengers, projected group attrition forecast, the size of the group, the peripheral profit, and the number of complementary seats requested by the group.

The process of traffic flow control is very important in an airline network with high levels of connecting traffic. The control of reservation inventory by origin-destination (itinerary) is accomplished using the value of the individual passenger to determine reservation availability. The passenger value is based on several factors including itinerary, departure

date, fare class, actual paid fare, and point of sale. The concept of virtual nesting was developed to approximately control reservation requests by origin-destination. It relies on the aggregation of various origin-destination fare classes that flow over a flight leg into amanageable number of virtual buckets based on reservation value. The value of an origin-destination class is the fare net of up-line and down-line displacement costs. The buckets are serially nested to ensure that as sales build up for a flight, the lower valued classes are automatically closed.

First generation yield management techniques were developed to maximize revenues on a leg-based inventory control scheme. Second generation systems clustered similar origin-destination/fare classes into "buckets" (see Smith et al. 1992, for a comprehensive description of this type of system, with attendant forecasting and performance measurement, at American Airlines), and the current state-of-the-art controls directly at the origin-destination level of detail. In a full origin-destination inventory control environment, reservation inventory is controlled by the actual origin and destination based on reservation value. This is accomplished using a network optimization model that takes the flight schedule, network capacity and the origin-destination demand forecasts and variance by class, to determine the probabilistic bid prices by leg and base compartment (Smith 1998). The bid price can be interpreted as the minimum acceptable fare for a reservation request on a flight leg to be accepted. The bid price for a multiple leg itinerary is the summation of the bid prices on each flight leg. Fares greater than the minimum acceptable fare or the total bid price are open for sale, subject to satisfying the associated fare rules. The fundamental difference between nested controls and origin-destination control is that availability is not prestored, but is dynamically calculated for each reservation request.

Belobaba (1989) discussed the development and application of a probabilistic decision model to airline seat inventory control. The concept referred to as Expected Marginal Seat Revenue (EMSR) takes into account the uncertainty associated with estimates of future passenger demand, as well as the nested structure of booking limits in airline reservation systems. Curry (1990) discussed an optimal airline seat allocation method that handles fare classes nested by origin and destinations. The solution procedure combines concepts from marginal seat revenue methods and mathematical programming to develop equations that find optimal allocation of seats when fare classes are nested on an origin-destination itinerary and the inventory is not shared among origin-destinations.

Williamson (1992) provided a comprehensive review of the application of mathematical programming and network flow models to the origin-destination seat inventory control problem. Belobaba (1998) reviewed the evolution of airline yield management from fare class to origin-destination seat inventory control. The author highlights the major milestones in the airline yield management arena, discusses the origin-destination seat inventory control problem, and outlines a new solution approach that uses the minimum acceptable bid prices derived from leg-based optimization models to control seat availability. Talluri and van Ryzin (1999a) discussed theoretical issues relating to bid prices, including an example of how they can be non-optimal, while also showing that bid prices are asymptotically optimal. A randomized version of the deterministic linear-programming method for computing network bid prices was given in Talluri and van Ryzin (1999b).

Crew Scheduling

The topic of crew scheduling can be subdivided into several categories including crew pairing generation, crew rostering, preferential bidding, and crew recovery. The crew pairing problem consists of constructing a set of pairings that cover at minimum cost a given set of flight segments. Typically all flight legs pertain to the same aircraft fleet and individual crew members are not considered in the problem. Each pairing has to be constructed in accordance with the prevailing collective agreement and airline regulations. The crew rostering problem entails the construction of monthly work schedules that assign pairings and rest periods to each employee, while considering pre-assigned activities such as recurrent training and personal holidays. The preferential bidding problem is a slight variation of the crew rostering problem wherein employee preferences are incorporated into the crew scheduling process. Although a large amount of research has been done

on crew scheduling, the problem of crew recovery in the aftermath of irregular airline operations has only recent surfaced in the research community.

Gershkoff (1989) outlined an optimization model that uses a set-partitioning framework, wherein the rows represent flights to be covered and the columns represent candidate crew trips. The primary objective of the model is to minimize the cost of flying the published airline schedule, subject to operational crew constraints. The crew scheduling problem is formulated as an integer-programming problem and can be solved using a commercial optimization software package. In these early efforts, it was found that a global optimization to the problem was difficult to achieve, and much research focused on the development of efficient heuristic procedures to address the problem. Concepts such as dynamic column generation and LP relaxation play a major role in the ability of researchers to tackle and successfully solve such largescale mathematical programs of crew pairing optimization. Crew pairing optimization has evolved substantially in the last ten years, as advances in computer technology, CPU run time, and the availability of efficient optimization software packages, have given practitioners the ability to solve large problems (Barutt and Hull 1990; Anbil et al. 1991). Innovative methods such as constraint branching (Ryan and Foster 1981), branch and cut (Hoffman and Padberg 1993), and column generation (Lavoie et al. 1988; Desrosiers et al. 1993) have successfully solved problems with very large numbers of feasible pairings.

The rostering problem involves the allocation of pairings to crew members to build a legal and feasible roster for each crew member in a crew rank. Often, such allocations are based on the so-called seniority preferential bidding (SPB) system in which pairings are allocated to crew members in decreasing order of seniority satisfying, whenever possible, each individual crew member's bids for certain types of work or rest periods. Heuristic algorithms of a greedy sequential type are most commonly used to solve this allocation problem, but they usually result in an inequitable distribution of work and often some pairings (referred to as "open flying") remain unallocated. An alternative form of the rostering problem involves the equitable allocation of pairings to all crew members of a crew rank. Measures of equitability are usually based on the notion that all members of a crew rank should do approximately the same amount of work. Equitability rostering problems can again be formulated as specially structured and generalized set partitioning models (Ryan 1992). Many alternative legal and feasible roster lines are generated for each crew member and the optimal solution of the model selects one line for each crew member to cover all pairings with the required number of crews and at a minimal total cost. The cost in this context can reflect an individual's preference for certain types of work. Column generation methods can again be used to reduce the need to generate many roster lines a priori for each crew member.

Aircraft Maintenance Routing

The aircraft routing problem is traditionally solved after the successful completion of the fleet assignment problem. It involves the allocation of candidate flight segments to a specific aircraft tail number within a given sub-fleet of the airline. The process of aircraft routing has traditionally been a manual activity at many carriers, but in recent years, researchers have developed efficient solution procedures that can be applied to the problem. During the normal operations of a carrier, situations often develop wherein modifications have to be made to the existing schedule plan. In addition, due to the inherent variation in passenger demand over the course of the week, airlines find it necessary to adjust their daily flight schedules to adequately meet demand. This will result in the need to make minor modification to aircraft routings and possibly fleet assignments.

Bard and Cunningham (1987) explored aircraft routing, while taking into consideration the benefits of through-flight schedules and the potential for increased revenues. Talluri (1996) describes an algorithm for making aircraft swaps that will not affect the equipment type composition overnighting at various stations throughout the airline's network. He also outlines the application of the swapping procedure in the airline schedule development process. Soumis et al. (1980) presented a model for largescale aircraft routing and scheduling problems which incorporates passenger flow issues. The authors discuss the technique used to transfer

information from the passenger flow problem to the aircraft routing problem. Daskin et al. (1989) discussed a Lagrangian relaxation approach to an integer-linear program model which is used to assign aircraft to routes in a hub and spoke network. Zhu et al. (1995) presented a mathematical formulation for the aircraft rotation problem and discuss its similarity with the asymmetric traveling salesman problem. Kabbani and Patty (1992) discussed aircraft maintenance routing at American Airlines, and the application of mathematical programming techniques to solve the problem. Desaulniers et al. (1994) outlined the daily aircraft routing and scheduling problem and presented two different formulations of the problem. The first is a set partitioning type formulation and the second a time constrained multicommodity network flow formulation.

Schedule Recovery

Schedule recovery in the aftermath of irregularities address how airlines reassign operational aircraft to scheduled flights. The main aspect of recovery for an airline is centered around flight rescheduling. Today, flight rescheduling is typically done manually at airlines, since research on the topic is relatively a recent phenomenon. Teodorovic and Stojkovic (1990) discussed a greedy heuristic algorithm for solving a lexicographic optimization problem which considers aircraft scheduling and routing in a daily schedule while minimizing the total number of canceled flights in the network. Jarrah et al. (1993) presented an overview of a decision support framework for airline flight cancellations and delays. Two separate network flow models provide solutions in the form of a set of flights delays (the delay model) or a set of flight cancellations (the cancellation model), while allowing for aircraft swapping among flights and the utilization of spare aircraft.

Yan and Yang (1996) developed a decision support framework for handling schedule perturbations. The framework is based on a basic schedule perturbation model constructed as a time-space network from which several perturbed network models are established for scheduling following irregularities. Cao and Kanafani (1997a, b) discussed a real-time decision support tool for the integration of airline flight cancellations and delays. They presented a quadratic 0-1 programming model for the integrated decision problem that maximizes operating profit, while taking into consideration both delay costs and penalties for flight cancellations. Arguello et al. (1997) presented a time-band optimization model for reconstructing aircraft routings in response to groundings and delays experienced in daily operations. Clarke (1997) discussed the airline schedule recovery problem that provides a comprehensive framework for reassigning operational aircraft to scheduled flights in the aftermath of irregularities. It is formulated as a mixed-integer linear-programming problem and solved using an optimization-based solution procedure. Lettovský (1997) outlined the airline integrated recovery problem that addresses crew assignment, aircraft routing, and passenger flow. The problem is formulated as a mixed-integer linear-programming problem and solved using Benders' decomposition. The master problem generates a cancellation and delay plan that satisfies imposed landing restrictions and assigns equipment types. The decoupled aircraft and crew subproblems are first solved and then the passenger flow subproblem is solved to determine new itineraries for disrupted passengers.

Concluding Remarks

Over the last forty years, operations research has played an integral role in many of the technological advancements credited to the airline industry. For instance, revenue management was developed using concepts from statistics, forecasting, and linear optimization. The various stages of the airline scheduling process have been modeled and implemented using ideas from network flow theory and mathematical programming. As researchers continue to make advances in these underlying fields of operation research, practitioners will be given added tools to tackle unanswered problems that exist in the industry. The continued improvement in commercial optimization software packages has encouraged and fostered the development of many of the state-of-the-art decision support tools now in use or currently under development. Such improvements, coupled with advances in computer hardware, will continue to push the horizon for OR practitioners in the airline industry.

See

- Air Traffic Management
- Integer and Combinatorial Optimization
- Linear Programming
- Network Optimization
- Yield Management
- Revenue Management

References

Abara, J. (1989). Applying integer linear programming to the fleet assignment problem. *Interfaces, 19*(4), 20–28.

Anbil, R., Gelman, E., Patty, B., & Tanga, R. (1991). Recent advances in crew-pairing optimization at American airlines. *Interfaces, 21*(1), 62–74.

Arguello, M., Bard, J., & Yu, G. (1997). *An optimization model for aircraft routing in response to groundings and delays*. Working paper, University of Texas– Austin.

Bard, J., & Cunningham, I. G. (1987). Improving through-flight schedules. *IIE Transactions, 19*, 242–251.

Barnhart, C., et al. (1996). A column generation technique for the long-haul crew assignment problem. In T. A. Cirani & R. C. Leachman (Eds.), *Optimization in industry, II*. New York: John Wiley.

Barnhart, C., et al. (1997). *Flight string models for aircraft fleeting and routing*. Working paper, MIT Center for Transportation Studies, Cambridge, MA.

Barutt, J., & Hull, T. (1990). Airline crew scheduling: Supercomputers and algorithms. *SIAM News*, 23(6), 1 and 20–22.

Belobaba, P. (1987). Airline yield management – An overview of seat inventory control. *Transportation Science, 21*, 63–73.

Belobaba, P. (1989). Application of a probabilistic decision model to airline seat inventory control. *Operations Research, 37*, 183–197.

Belobaba, P. (1998). The evolution of airline yield management: Fare class to origin-destination seat inventory control. In *Handbook of airline marketing*, Chapter 23. New York: McGraw-Hill.

Berge, M., & Hopperstand, C. (1993). Demand driven dispatch: A method for dynamic aircraft capacity assignment, models and algorithms. *Operations Research, 41*, 153–168.

Bertsimas, D., & Patterson, S. S. (1998). The air traffic flow management problem with enroute capacities. *Operations Research, 46*, 406–422.

Cao, J.-M., & Kanafani, A. (1997a). Real-time decision support for integration of airline flight cancellations and delays, part I: Mathematical formulation. *Transportation Planning and Technology, 20*, 183–199.

Cao, J.-M., & Kanafani, A. (1997b). Real-time decision support for integration of airline flight cancellations and delays, part II: Algorithm and computational experiments. *Transportation Planning and Technology, 20*, 201–217.

Clarke, M. (1997). *Development of heuristic procedures for flight rescheduling in the aftermath of irregular airline operations*. Sc.D. Dissertation, MIT International Center for Air Transportation, Cambridge, MA.

Clarke, L. W., Hane, C. A., Johnson, E. L., & Nemhauser, G. L. (1996). Maintenance and crew considerations in fleet assignment. *Transportation Science, 30*, 249–260.

Cook, T. M. (1989). Airline operations research. *Special Issue of Interfaces*, *19*(4), 1–74.

Curry, R. (1990). Optimal airline seat allocation with fare classes nested by origin and destinations. *Transportation Science, 24*(3).

Daskin, M., et al. (1989). A lagrangian relaxation approach to assigning aircraft to routes in hub and spoke networks. *Transportation Science, 23*(2).

Desaulniers, G., et al. (1994). *Daily aircraft routing and scheduling*. Les Cahiers du GERAD, June.

Desrosiers, J., et al. (1993). *The airline crew pairing construction problem*. Working paper, GERAD, Montreal.

Farkas, A. (1996). The influence of network effects and yield management on airline fleet assignment decisions. MIT Flight Transportation Report R96-1, Cambridge, MA.

Gershkoff, I. (1989). Optimizing flight crew schedules. *Interfaces, 19*(4), 29–43.

Gershkoff, I. (1998). A hybrid scheduled/charter framework for long-haul air service. In *Handbook of airline marketing*, Chapter 48. New York: McGraw-Hill.

Hane, C. A., Barnhart, C., Johnson, E. L., Marsten, R. E., Nemhauser, G. L., & Sigismondi, G. (1995). The fleet assignment problem: Solving a large scale integer program. *Mathematical Programming, 70*, 211–232.

Hoffman, K., & Padberg, M. (1993). Solving airline crew scheduling problems by branch-and-cut. *Management Science, 39*, 657–682.

Jacobs, T., et al. (1999). *Origin-destination fleet assignment: Incorporating passenger flow into the fleeting process*. Presentation at the AGIFORS schedule and strategic planning study group meeting, New Orleans.

Jarrah, A., et al. (1993). A decision support framework for airline flight cancellations and delays. *Transportation Science, 27*, 266–280.

Kabbani, N., & Patty, B. (1992). Aircraft routing at american airlines. AGIFORS annual fall symposium.

Kniker, T. (1998). Itinerary based airline fleet assignment. Ph.D. dissertation, MIT Operations Research Center, Cambridge, MA.

Lavoie, S., et al. (1988). A new approach of crew pairing problems by column generation and application to air transport. *European Journal of Operational Research, 35*, 45–58.

Lee, A. O. (1990). *Probabilistic and statistical models of the airline booking process for yield management*. Sc.D. Dissertation, Department of Civil Engineering, MIT, Cambridge, MA.

Lettovsky, L. (1997). *Airline operations recovery: An optimization approach*. Ph.D. dissertation, Georgia Institute of Technology, Atlanta.

McGill, J. I., & van Ryzin, G. J. (1999). Revenue management: Research overview and prospects. *Transportation Science, 33*, 233–256.

Richter, H. (1989). Thirty years of airline operations research. *Interfaces, 19*(4), 3–9.
Ryan, D. M. (1992). The solution of massive generalised set partitioning problems in aircrew scheduling. *Journal of the Operational Research Society, 43*, 459–467.
Ryan, D. M., & Foster, B. A. (1981). An integer programming approach to scheduling. In A. Wren (Ed.), *Computer scheduling of public transport* (pp. 269–280). Amsterdam: North-Holland.
Shenoi, R. G. (1996). *Integrated airline schedule optimization: Models and solution methods*. MIT dissertation, Center for Transportation Studies, Cambridge, MA.
Smith, B. C. (1990). *A break-even approach to group control*. AGIFORS symposium proceedings, Macau, September.
Smith, B. (1998). Airline planning and marketing decision support: A review of current practices and future trends. In *Handbook of airline marketing*, Chapter 10. New York: McGraw Hill.
Smith, B. (1999). *Frequency generation model: A better starting point*. Rotations: Flight Scheduling Jl., The Sabre Group, Inc., Dallas/Fort Worth, Texas.
Smith, B. C., Leimkuhler, J. F., & Darrow, R. M. (1992). Yield management at American airlines. *Interfaces, 22*(1), 8–31.
Smith, B., et al. (1997). *Optimal load factors: Coordinated scheduling, pricing and yield management decisions*. AGIFORS symposium proceedings, Denpesar Bali, Indonesia.
Soumis, F., & Nagurney, A. (1993). A stochastic multiclass airline network equilibrium model. *Operations Research, 41*, 710–720.
Soumis, F., et al. (1980). A model for large scale aircraft routing and scheduling problems. *Transportation Research: Part B, 14*, 191–201.
Subramanian, R., Scheff, R. P., Jr., Quillinan, J. D., Wiper, D. S., & Marsten, R. E. (1994). Coldstart: Fleet assignment at delta airlines. *Interfaces, 24*(1), 104–120.
Talluri, K. (1996). Swapping applications in a daily airline fleet assignment. *Transportation Science, 30*, 237–248.
Talluri, K., & van Ryzin, G. (1999a). An analysis of bid-price controls for network revenue management. *Management Science, 44*, 1577–1593.
Talluri, K., & van Ryzin, G. (1999b). A randomized linear programming method for computing network bid prices. *Transportation Science, 33*(2).
Teodorovic, D. (1988). *Airline operations research*. New York: Gordon and Breech.
Teodorovic, D., & Stojkovic, G. (1990). Model for operational daily airline scheduling. *Transportation Planning and Technology, 14*, 273–285.
Vasquez-Marquez, A. (1991). American airlines arrival slot allocation system. *Interfaces, 21*(1), 42–61.
Weatherford, L. R., & Bodily, S. E. (1992). A taxonomy and research overview of perishable-asset revenue management: yield management, overbooking, and pricing. *Operations Research, 40*, 831–844.
Williamson, E. L. (1992). *Airline network seat inventory control: Methodologies and revenue impacts*. Ph.D. dissertation, MIT Flight Transportation Laboratory Report R92-3, Cambridge, MA.
Yan, S., & Yang, D.-H. (1996). A decision support framework for handling schedule perturbation. *Transportation Research: Part B, 30*, 405–419.
Zhu, Z., et al. (1995). *The aircraft rotation problem*. Working paper, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta.

Algebraic Modeling Languages for Optimization

Robert Fourer
Northwestern University, Evanston, IL, USA

Introduction

Algebraic modeling languages are sophisticated software packages that provide a key link between an analyst's mathematical conception of an optimization model and the complex algorithmic routines that seek out optimal solutions. By allowing models to be described in the high-level, symbolic way that people think of them, while automating the translation to and from the quite different low-level forms required by algorithms, algebraic modeling languages greatly reduce the effort and increase the reliability of formulation and analysis. They have thus played an essential role in the spread of optimization to all aspects to OR/MS and to many allied disciplines.

Background and Motivation

Practical software packages for solving optimization problems emerged in the 1950s, as soon as there were computers to run them. Initially based on linear programming, these solvers were soon generalized to allow for nonlinearities and to accommodate integer variables and other discrete decisions. Despite continuing progress in algorithms and in computing, however, by the beginning of its second decade large-scale optimization had come to be seen as failing to live up to its potential. The key weakness in early optimization systems was not in their algorithms, however, but in their interaction with modelers. The human time and cost of preparing a solver's

input and examining its output often greatly dominated the computer costs of solving. The cause of this difficulty, and its ultimate cure, can best be understood by considering the steps of the optimization modeling process and their interaction with the technical requirements of large-scale optimization.

The process of building practical optimization models involves several interrelated steps. The first and most important is extensive communication with the owner of a decision problem to identify the problem ingredients and to ascertain the extent to which optimization is feasible within the managerial structure of the client organization and the cognitive limitations of the model user. Next is the formulation of a mathematical abstraction of the problem — a model — that offers a sufficiently accurate characterization of the real situation in terms of reasonably available data. Further steps build datasets, generate the corresponding optimization problem instances, feed the problem instances to solvers, run the solvers to produce results that are optimal or near — optimal by the model's criteria, and process the results into descriptions of decisions in forms that clients can understand and analyze. These tasks are carried out repeatedly in a kind of feedback loop, as further communication results in model modifications and data refinements due to invalid assumptions, bad data, programming errors, and (most interestingly) the identification of previously unelucidated policies, constraints and preferences. The success of an optimization application depends critically on how fast one can implement the central feedback loop — formulation, solution, analysis, revision. The faster these steps, the greater the likelihood that the modeling effort will receive sufficient attention from the client in the communication phase to ensure that the model will eventually be adopted and supported. Thus, as the number — crunching solution phase became progressively more efficient with advances in algorithms and computers, the steps involving human analysts became the bottlenecks in this process.

In fact, the optimization development cycle was found to take much more analyst time than expected. The culprit was the awkward and error-prone work of converting an optimization problem between the modeler's conception and the algorithm's representation. Indeed, the natural way for a modeler to think about and express models is in direct conflict with the input requirements of solution algorithms. As detailed in Fourer (1983), whereas the modeler's form is symbolic, general, concise, and understandable to other modelers, the solver's form is contrary in every respect: explicit, specific, extensive, and convenient for computation. For all but the smallest and simplest instances, the only practical way to make the conversion from the modeler's to the algorithm's form is by writing a computer program for the purpose, and it was the continued maintenance and debugging of this program in successive cycles of the development process that unexpectedly soaked up so much analyst time. Whether a program of this kind is working correctly is particularly hard to confirm, as the only detailed evidence of its performance consists of voluminous coefficient lists and other details that are specifically intended for algorithmic efficiency rather than human comprehension.

Optimization modeling languages were conceived as a way of alleviating this bottleneck of conversion. They allow people to convey their formulations to computer systems in much the same way that they would write them out or describe them to colleagues. Computer systems that implement modeling languages also facilitate analysis and reporting using the terminology of the model, thus further speeding the development cycle.

Any convenient form of representation for some class of optimization applications can in principle give rise to a modeling language. Many general-purpose modeling languages, however, are based on the familiar mathematical representation of an optimization problem as the minimization or maximization of a function of decision variables, subject to equations and inequalities in functions of the variables. The most popular languages are founded in particular on familiar expressions — like $\sum_{j=1}^{n} a_{ij} x_j$, $\sqrt{\sum_{s \in S} (g_{rs} - h_s)^2}$, or $G_{km} \cos(\delta_k - \delta_m)$ — that use the operators and functions of elementary algebra, though written in a form that requires only a computer character set. Most such languages have been generalized through the use of notations from logic, computer programming, and other disciplines, but in recognition of their origins they are widely known as algebraic modeling languages (Bisschop and Meeraus 1982).

The initial popularity of algebraic modeling languages derived in part from their users' familiarity with mathematical optimization theory. They quickly became recognized as offering a valuable tradeoff between the convenience of highly application-specific representations and the power of informal natural-language problem descriptions. Their combination of precision and generality enabled them to support optimization as a paradigm for modeling and decision making in diverse applications of operations research and throughout engineering, science, economics, and management. At the same time, their flexibility enabled them to accommodate the unique features that distinguish individual applications in realistic situations.

Example. To give a further view of the issues involved in designing, selecting, and using an algebraic modeling language, a modest example of a model of optimal multiperiod transportation of a single commodity is presented. The presentation describes the model first in words and mathematical formulas, and then equivalently in one of the widely used modeling languages, concluding by describing three major aspects of working with the model: the specification of data, the invocation of solvers, and the examination of results.

Mathematical formulation. To begin describing an algebraic model for transportation, it may be said that there is a set I of cities where supply of a product originates, and a set J of cities where demand must be met. A set of links $L \subseteq I \times J$ specifies those origin-destination pairs (i,j) for which shipments from i to j are possible. The goal to plan for the next T weekly time periods.

The objective of this model is to decide how much to ship from each origin to each destination in each week, so as to minimize the total cost of all shipments. Decision variables $x_{ijt} \geq 0$ and parameters $c_{ijt} \geq 0$ for each $(i,j) \in L$ and $t = 1, \ldots, T$ are introduced, representing respectively the amounts to be shipped (which will be determined by optimization) and the costs per unit of shipment (which are supplied as data). In terms of these quantities, the objective may be written algebraically as

$$\text{Minimize} \sum_{(i,j)\in L} \sum_{t=1}^{T} c_{ijt}\, x_{ijt}$$

The essential constraints on the decision variables are next described in terms of parameters a_{it} for each $i \in I$ and $t = 1, \ldots, T$, representing the amount that becomes available for shipment at origin i in week t, and b_{jt} for each $j \in J$ and $t = 1, \ldots, T$, representing the amount required to meet expected demands at destination j in week t. The possibility of week-to-week fluctuations in shipping costs suggests that not all supply should be shipped out in the week that it becomes available. Decision variables y_{it} for each $i \in I$ and $t = 1, \ldots, T$ are also introduced, to represent the amount of product in inventory at origin i at the end of week t. The following algebraic constraints then serve to express the limitations on shipping out of each origin and the requirements of meeting demand at each destination:

$$\sum_{j\in J:(i,j)\in L} x_{ijt} + y_{it} \leq a_{it} + y_{i,t-1}, \text{ for each } i \in I, t = 1, \ldots, T$$

$$\sum_{i\in I:(i,j\in L)} x_{ijt} = b_{jt}, \text{ for each } j \in J, t = 1, \ldots, T$$

For the sake of this simple model the possibility of initial inventories is disregarded, thus implicitly setting to zero all terms y_{i0} in the origin constraints for $t = 1$.

The shipment plan is also commonly subject to certain operational considerations. As just one example, the amount shipped over link (i,j) in all weeks may be required to sum to at least a certain amount, given by a parameter $d_{i,j,}$ if that link is used in any period at all. A quite general way of implementing this restriction through algebraic constraints is by defining a corresponding collection of decision variables z_{ij} that can only take the values 0 or 1. Then it may be written:

$$d_{ij}Z_{ij} \leq \sum_{t=1}^{T} x_{ijt} \leq \min\left(\sum_{t=1}^{T} a_{it}, \sum_{t=1}^{T} b_{jt}\right) z_{ij}, \text{ for each } (i,j) \in L$$

which forces shipments to be zero when z_{ij} is zero, or to be at least $d_{i,j}$ (and at most some implied upper bound) when z_{ij} is one.

Modeling language formulation. The representation of this model in an algebraic modeling language is fundamentally the same as this mathematical formulation, with the differences deriving mainly from the need to communicate the model unambiguously and to use a standard character

set. Thus for instance in the AMPL modeling language (Fourer et al. 1990) the sets and parameters that describe the data might be specified as follows:

```
set ORIG; # origins
set DEST; # destinations
set LINKS within {ORIG,DEST};
param T integer > 0;
param supply {ORIG,1..T} >= 0;
param demand {DEST,1..T} >= 0;
param cost {LINKS,1..T} > 0;
param minShip {LINKS} >= 0;
```

AMPL defines a standard indexing expression such as {ORIG,1..T} to correspond to a statement like "for each $i \in I, t = 1, \ldots, T$" in the mathematical formulation (though the i and t need be included only where actually used). The use of more meaningful names like ORIG for I and supply for a, while not required, often proves useful for keeping models understandable as they grow in complexity. Models can also be more thoroughly documented through a variety of comments, which are seen here for the first two sets but will be otherwise omitted in this description for the sake of brevity.

Decision variables are next defined in much the same way as parameters:

```
var Ship {LINKS,1..T} >= 0;
var Inv {ORIG,1..T} >= 0;
var Use {LINKS} integer >= 0, <= 1;
```

Indeed the only difference between parameters and variables is that the former are specified as part of the data while the latter are given their values by the solver so as to optimize the objective. Given the definitions in this example, AMPL's statement for the objective of the model is as follows:

```
minimize TotalCost:
  sum {(i,j) in LINKS, t in 1..T} cost
  [i,j,t] * Ship[i,j,t];
```

This is the same algebraic expression as in the mathematical formulation, adapted for input to a computer system; sum {...} corresponds to the sigma expressions, while cost[i,j,t] and Ship[i,j,t] are the AMPL representations for c_{ijt} and x_{ijt}. An explicit operator * is introduced to represent the multiplication that is customarily implicit in mathematical expressions.

Constraints are similarly converted to algebraic expressions in the modeling language. They are somewhat more complex than the objective because they come in indexed collections and use relational operators for equalities and inequalities:

```
subject to Supply {i in ORIG, t in
1..T}:
  sum {(i,j) in LINKS} Ship[i,j,t] +
  Inv[i,t]
    <= supply[i,t] + (if t > 1 then Inv
    [i,t-1] else 0);
subject to Demand {j in DEST, t in
1..T}:
  sum {(i,j) in LINKS} Ship
  [i,j,t] = demand[j,t];
subject to ZeroMin1 {(i,j) in LINKS}:
  minShip[i,j] * Use[i,j] <= sum
  {t in 1..T} Ship[i,j,t];
subject to ZeroMin2 {(i,j) in LINKS}:
  sum {t in 1..T} Ship[i,j,t] <=
    min (sum {t in 1..T} supply[i,t],
      sum {t in 1..T} demand[j,t]) *
      Use[i,j];
```

The emphasis is on keeping the original forms of the constraints as much as possible, while letting the AMPL translator automate the work of evaluating coefficient expressions, collecting terms on the left, and other regularizations that may be required by solvers. Each modeling language does introduce some changes; here AMPL requires the double-inequality constraint to be split in two, but streamlines the specifications of the supply and demand constraints by interpreting {(i,j) in LINKS} so that it specifies indexing over only whichever index has not already been fixed. Also the assumption of zero initial inventories must be made explicit, in this example by using an if-then-else construct to handle inventories at the end of "week 0" specially.

Specification of data. Each modeling language offers its own format for associating actual data values with the sets and parameters in the symbolic model. A small collection of data for our example could be specified by an AMPL text file that begins as follows:

```
set ORIG := YYZ LAF CVG PIT CLE;
set DEST := ABE ORF;
set LINKS := (YYZ,ABE) (YYZ,ORF)
(LAF,ABE) (CVG,ORF)
  (PIT,ABE) (PIT,ORF) (CLE,ABE)
  (CLE,ORF);
param T := 5;
```

```
param demand: 1 2 3 4 5 :=
  ABE 1000 1200 1900 2500 2000
  ORF 2100 3000 4900 7700 5000;
param supply: 1 2 3 4 5 :=
  YYZ 2100 2250 3190 3120 3500
  LAF 1400 1250 1320 1220 1100
  CVG 1650 1250 2290 2120 2300
  .......
```

Model and data together specify a particular instance of an optimization problem for which a solution can be sought.

Modeling language systems typically also offer facilities for exchange of data with popular databases, spreadsheets, and other repositories of data for decision support. Indeed there is a close correspondence between the way that data values are described in algebraic models and the way they are organized in relational databases (Fourer 1997). Close interaction with data management software is often important to the integration of optimization into business operations.

Invocation of solvers. Modeling language software automatically reads and interprets a model and data, generates an instance, and conveys the instance to a solver in the required form. In AMPL it suffices to give only a few commands for these purposes:

```
ampl: model multiEORMS.mod;
ampl: data multiEORMS.dat;
ampl: option solver cplexamp;
ampl: solve;
  73 variables:
     8 binary variables
     65 linear variables
  51 constraints, all linear;
  221 nonzeros
  1 linear objective; 40 nonzeros.
  CPLEX 12.2.0.2: optimal integer
  solution; objective 288503.5
  65 MIP simplex iterations
  2 branch-and-bound nodes
```

The solver software is a separate product for which there may be many alternatives. For this model, a different mixed-integer programming solver might have been used instead:

```
ampl: model multiEORMS.mod;
ampl: data multiEORMS.dat;
ampl: option solver gurobi;
ampl: solve;
  Gurobi 4.0.1: optimal solution;
  objective 288503.5
  71 simplex iterations
  plus 52 simplex iterations for
  intbasis
```

Also a full variety of options, specific to each solver, are accessible as needed to set algorithmic options and report progress of long runs.

Examination of results. Once the solver has returned a solution, the same expression forms that are so convenient in describing the model to the computer system can also be used to describe the results to be viewed. For example to show for each link the ratio of total shipments to minimum shipment over all periods, one can simply ask AMPL to display the appropriate sum, adapting the same summation syntax that was used in the model:

```
ampl: display {(i,j) in LINKS}
ampl? sum {t in 1..T} Ship[i,j,t] /
minShip[i,j];
: ABE    ORF :=
CLE 0    1.69032
CVG .    1.48992
LAF  1.38636 .
PIT 0    0
YYZ 1    2.99143
```

Simple displays of this kind do much to support the cycle of development, by speeding the modeler's aggregation, transformation, and analysis of solutions. For later deployment of the model, facilities are also available for writing more precisely formatted text and for sending results off to other software for analysis.

Advantages

The fundamental concept of algebraic modeling languages — that the entire optimization modeling cycle is best carried out at the level of the model formulation — makes possible the creation of modeling systems that have a number of desirable characteristics. This article has already described how such systems promote optimization modeling by making the entire process more efficient and reliable. The modeling language concept has proven to have other benefits as well. Principally these relate to independent treatment of distinct aspects of

optimization, and to extensions well beyond linear optimization.

Independence. In contrast to the highly integrated design common of software for mathematical and statistical modeling and for simulation, modeling language systems for optimization have promoted an independence of model, data, and solvers. This property has proved to be of benefit of users in several ways.

Because the sets and parameters of a model are described symbolically along with the variables, objectives, and constraints, the same model readily describes any number of problem instances of different sizes and purposes. This model-data independence allows prototypes to be scaled up quickly to larger and more realistic scenarios through changes to the data files alone. Equally it provides flexibility to experiment with different formulations on the same data, as is often essential for finding tractable approaches to difficult mixed-integer modeling applications. Following the initial development stages, model-data independence is also beneficial, allowing the model to be frozen while deployment focuses on periodically generating data for new runs. The full symbolic model description remains accessible, however, whenever modifications are necessary to accommodate new circumstances or analyses.

Because modeling languages are designed to describe models and their data in an abstract way, they are not tied to particular software for optimization or even to particular methods. This model-solver independence allows instances to be benchmarked over a range of solvers. The choice of a solver for deployment can then be based on a tradeoff between price and performance, and can be revisited as optimization technology evolves. The very substantial changes in linear optimization packages that have occurred over recent decades have thus not required corresponding changes in modeling language design.

Another virtue of model-solver independence is to relieve the analyst of much tedious work of converting between the modeler's form and the various algorithms' forms. Originally this work consisted mainly of generating coefficient lists and bound vectors. But as languages have become more sophisticated it has come to include conversions to linear representations from other forms that may be closer to the original model conception, such as piecewise-linear formulations and network node-arc descriptions.

Extensions. Algebraic languages can express nonlinear optimization problems as easily as linear ones, simply by permitting expressions that are nonlinear in the variables. Thus for instance in our transportation example if it is desired to encourage shipments of moderate size, neither too small nor too large, the objective could be changed to

$$\text{Minimize} \sum_{(ij)\in L} \sum_{t=1}^{T} c_{ijt} \frac{x_{ijt}^{\alpha}}{1 - x_{ijt}/l_{ij}}$$

where $0 < \alpha < 1$ and l_{ij} is some positive link capacity. To specify this in a modeling language it suffices to write the corresponding nonlinear expression:

```
minimize TotalCost:
  sum {(i,j) in LINKS, t in 1..T}
    cost[i,j,t] * Ship[i,j,t]^alpha / (1 - Ship[i,j,t]/
    lim[i,j]);
```

After the model and data have been processed, a representation of each nonlinear objective and constraint expression is included as part of the instance representation passed to the solver interface. The interface then uses this representation to compute function values at successive points generated by the solver as it iterates toward the optimum; the interface also provides analytical (not approximate) first and second derivatives automatically by methods that avoid the overhead of symbolically differentiating each nonlinear expression (Rall and Corliss 1996; Griewank and Walther 2008). This approach is considerably more efficient and less error-prone than working directly with the nonlinear solver, which would require writing, debugging, and maintaining a program for each nonlinear expression and its derivatives.

The technology for recognizing and processing conventional nonlinear expressions extends moreover to virtually any kind of expression that can be written in terms of functions and operators. Thus, it is possible to substantially extend the range of models that can be expressed naturally through algebraic modeling languages. Current implementations allow for example the specification of complementarity conditions, and the description of logical restrictions

using operators like "or" and "not" rather than through the introduction of zero-one variables. Also special cases like quadratic objectives and constraints can be detected and transformed automatically.

The algebraic expressions that are useful in describing individual objectives and constraints are also useful in describing manipulations of models and transformations of data. Thus almost as soon as modeling languages became available, users started finding ways to adapt model notations to implement sophisticated solution strategies and iterative schemes. These efforts stimulated the evolution within algebraic modeling languages of scripting features, which include statements for looping, testing, and assignment. Thus, for instance, to test the sensitivity of our multiperiod transportation model to the minimal-shipment thresholds, the modeler could write a simple loop:

```
for {k in 1..10} {
  let {(i,j) in LINKS} minShip[i,j] := minShip[i,j]
  + 250;
  solve;
  if solve_result = "infeasible" then break;
}
```

Industrial and research applications now commonly employ scripts involving many hundreds of lines. The efficiency and convenience of algebraic modeling is thus extended to situations much more complex than the solving of individual optimization problems.

Alternatives

The ideas and benefits of algebraic modeling languages are available to various extents in several kinds of software.

General-purpose algebraic modeling languages embody model-data-solver independence to the greatest degree, supporting links to numerous independently-developed solvers and data-management systems. The most widely used commercial systems in this category are AIMMS (Paragon Decision Technology 2011), AMPL (AMPL Optimization 2011), GAMS (GAMS Development 2011), and MPL (Maximal Software 2011); for noncommercial uses, GNU MathProg (Free Software Foundation 2011) and Zimpl (Zuse Institute 2011) are open-source alternatives licensed under the GNU GPL. All base their language designs on the same fundamental ideas, though with varying specifics in some key respects. They differ more substantially in aspects of their user, solver, and data management interfaces.

Solver-specific algebraic modeling languages offer similar designs but have been implemented to be used mainly or exclusively with one solver developer's products. By forgoing solver independence, they can offer more complete and integrated support for one suite of solvers, often including ones that go beyond the traditional algorithmic approaches for linear and smooth nonlinear problems. Among the best-known are LINGO (LINDO Systems 2011a), Mosel (Fair Isaac 2011), and OPL (IBM Corporation 2011b).

An algebraic modeling framework for optimization can also be implemented within a general object-oriented modeling language. Specialized object types are defined to represent common model entities such as parameters, variables, and constraints; then all of the standard operators and functions are overloaded to act specially when applied to these types. Thus for example using the CPLEX Optimization Studio's Concert C++ library (IBM Corporation 2011a) one can make definitions such as

```
IloEnv env;
IloNumArray supply(env);
IloNumVarArray Use (env, nOrig, 0, 1, ILOINT);
IloExpr totalShipFrom(env);
```

and then express, for example, some supply-limit constraints by writing

```
for (i = 0; i < nOrig; i++) {
  for (j = 0; j < nDest; j++) {
    totalShipFrom += Ship[i][j];
  }
  mod.add(totalShipFrom <= supply[i] * Use[i]);
}
```

What appear to be arithmetic and comparison operations are in fact interpreted as instructions to build up a constraint data structure for an affiliated solver. A similar Concert interface is available for Java and .NET, and the same idea with more general-purpose solver support has been carried through by, among others, FLOPC++ (COIN-OR 2011), OptimJ (Ateji 2011) for Java, and Pyomo (Sandia 2011) for Python. Compared to languages specially designed for algebraic modeling, these object-oriented tools have less natural representations — particularly in the use of indexing

sets — and require more user involvement in the lower-level aspects of programming. They can, however, offer the advantages of a much richer programming environment than is afforded by the scripting facilities of specialized modeling languages; also they hold out the possibility of simplifying the integration of optimization models into broader applications.

Several kinds of modeling language integration with general-purpose analytical tools have also proved popular. Some general-purpose modeling languages have connections to solvers running under MATLAB (Mathworks 2011), and there is a MATLAB-based connection from AMPL to many independent solvers through the TOMLAB environment (Tomlab Optimization 2011). The AMPL-like OPTMODEL language (SAS Institute 2011) supports SAS/OR solvers as an integrated part of the SAS business analytics system. By far the most popular are modeling languages implemented as Microsoft Excel add-ins, notably the Frontline Premium Solver (Frontline Systems 2011) and What'sBest (LINDO Systems 2011b), with a variety of solver options. Because these languages are closely tied to Excel spreadsheet data forms, they are very limited in power and expressiveness. They offer, however, the very substantial benefit of being able to integrate optimization into the most widely used environment for all kinds of business analyses.

Extensions

Enhancements to algebraic modeling languages are basically of two kinds: extensions to the languages themselves, and improvements to the ways in which the languages interact with other systems.

Modeling language extensions tend to be driven by solver developments. Whenever algorithms are developed to effectively solve new forms of optimization problems, modeling language developers are challenged to provide more convenient support. Operators and syntaxes may be added to let modelers describe new forms in the most natural ways, as happened, for example, with the advent of more effective solution strategies to handle complementarity conditions (Ferris et al. 1999). Alternatively, additional logic may be introduced to detect special cases that are significant to new algorithms, as occurred with the discovery of efficient methods for second-order cone problems that were equivalent to several common kinds of nonlinear constraints (Lobo et al. 1998); here the recognition technology is still in the process of being developed. Ferris et al. (2009) survey a variety of such problems including also bi-level and generalized nonlinear optimization. Constraint programming solvers have motivated a variety of extended forms for logical and discrete optimization (Lustig and Puget 2001) which have also proved to be valuable for describing discrete optimization more naturally to other kinds of solvers.

Enhancements to the interfaces and interoperability of modeling language systems tend to be driven by more general developments in computing. This has been seen in the creation of more sophisticated user interfaces for model building, more powerful object-oriented programming interfaces for embedding models within larger applications, and closer links to data management systems. Popularity of Python-based optimization modeling tools is an example of such a trend. Another is the growing attractiveness of optimization "software as a service" — as pioneered by the NEOS Server (Czyzyk et al. 1998; Dolan et al. 2002) — which seems likely to motivate widespread access to algebraic modeling languages "in the cloud" in ways that will foster even more efficient and convenient development cycles for optimization.

See

- ▶ Model Management
- ▶ Spreadsheets
- ▶ Structured Modeling

References

AMPL Optimization LLC. (2011). *AMPL modeling language* (Web searchable).

Ateji SAS (2011). *OptimJ modeling language* (Web searchable).

Bisschop, J., & Meeraus, A. (1982). On the development of a general algebraic modeling system in a strategic planning environment. *Mathematical Programming Study, 20*, 1–29.

COIN-OR Foundation. (2011). *FLOPC++ modeling language* (Web searchable).

Czyzyk, J., Mesnier, M. P., & Moré, J. J. (1998). The NEOS server. *IEEE Computational Science and Engineering, 5*, 68–75.

Dolan, E. D., Fourer, R., Moré, J. J., & Munson, T. S. (2002). Optimization on the NEOS server. *SIAM News, 35*, 6, 4, 8–9.

Fair Isaac Corporation. (2011). *Xpress-Mosel modeling language* (Web searchable).
Ferris, M. C., Dirkse, S. P., Jagla, J.-H., & Meeraus, A. (2009). An extended mathematical programming framework. *Computers and Chemical Engineering, 33*, 1973–1982.
Ferris, M. C., Fourer, R., & Gay, D. M. (1999). Expressing complementarity problems in an algebraic modeling language and communicating them to solvers. *SIAM Journal on Optimization, 9*, 991–1009.
Fourer, R. (1983). Modeling languages versus matrix generators for linear programming. *ACM Transactions on Mathematical Software, 9*, 143–183.
Fourer, R. (1997). Database structures for mathematical programming models. *Decision Support Systems, 20*, 317–344.
Fourer, R., Gay, D. M., & Kernighan, B. W. (1990). A modeling language for mathematical programming. *Management Science, 36*, 519–554.
Fourer, R., Gay, D. M., & Kernighan, B. W. (2003). *AMPL: A modeling language for mathematical programming* (2nd ed.). Belmont, CA: Cengage Learning.
Free Software Foundation. (2011). *GNU MathProg modeling language* (Web searchable).
Frontline Systems, Inc. (2011). *Premium solver for excel* (Web searchable).
GAMS Development Corporation. (2011). *GAMS modeling language* (Web searchable).
Griewank, A., & Walther, A. (2008). *Evaluating derivatives: Principles and techniques of algorithmic differentiation* (2nd ed.). Philadelphia, PA: SIAM.
IBM Corporation. (2011a). *Concert technology* (Web searchable).
IBM Corporation. (2011b). *OPL modeling language* (Web searchable).
Kallrath, J. (Ed.). (2004). *Modeling languages in mathematical optimization*. Dordrecht, The Netherlands: Kluwer Academic Publishers.
Kuip, C. A. C. (1993). Algebraic languages for mathematical programming. *European Journal of Operational Research, 67*, 25–51.
LINDO Systems. (2011a). *LINGO modeling language* (Web searchable).
LINDO Systems. (2011b). *What's best excel add-in* (Web searchable).
Lobo, M. S., Vandenberghe, L., Boyd, S., & Lebret, H. (1998). Applications of second-order cone programming. *Linear Algebra and its Applications, 284*, 193–228.
Lustig, I. J., & Puget, J.-F. (2001). Program does not equal program: Constraint programming and its relationship to mathematical programming. *Interfaces, 31*(6), 29–53.
Maximal Software Inc. (2011). *MPL modeling language* (Web searchable).
Paragon Decision Technology. (2011). *AIMMS modeling language* (Web searchable).
Rall, L. B., & Corliss, G. F. (1996). An introduction to automatic differentiation. In M. Berz et al. (Eds.), *Computational differentiation: Techniques, applications, and tools* (pp. 1–17). Philadelphia, PA: SIAM.
Sandia National Laboratories. (2011). *Pyomo modeling language* (Web searchable).
SAS Institute Inc. (2011). *SAS/OR PROC OPTMODEL modeling language* (Web searchable).
The Mathworks, Inc. (2011). *MATLAB technical computing environment* (Web searchable).
Tomlab Optimization. (2011). *TOMLAB optimization environment* (Web searchable).
Zuse Institute Berlin. (2011). *Zimpl modeling language* (Web searchable).

Algorithm

A computational procedure whose application yields a solution to an associated class of problems.

See

► Computational Complexity

Algorithmic Complexity

► Computational Complexity

Alternate Optima

Distinct solutions to the same optimization problem.

See

► Unique Solution

Alternate Paths

In queueing networks, more than one arc connecting the same two nodes.

See

► Queueing Theory

AMPL

A Mathematical Programming Language. An algebraic modeling language for mathematical programming that supports numerous commercial and open source software solvers, including CBC, CPLEX, FortMP, Gurobi, MINOS, IPOPT, SNOPT and KNITRO.

References

Fourer, R., Gay, D. M., & Kernighan, B. W. (2002). *AMPL: A modeling language for mathematical programming* (2nd ed.). Duxbury Press.

Analytic Combat Model

A self-contained military model that directly computes its results from initial conditions, with no intermediate human interaction.

See

► Battle Modeling

Analytic Hierarchy Process

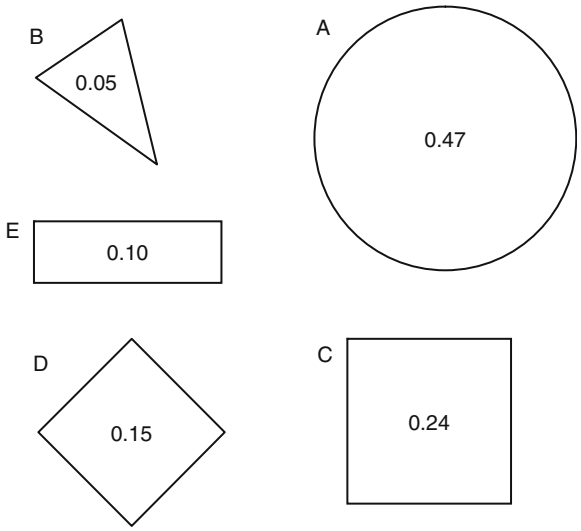
Thomas L. Saaty
University of Pittsburgh, Pittsburgh, PA, USA

Introduction

It is safe to say that people in general think that to measure something is to apply a scale with an arbitrary unit to it designed to measure things with respect to some property. One would then read the numbers on the scale to get the measurement. Such scales are hard physical scales or soft ones like the ones that are used to measure IQ and creativity. What is clear is that whatever is measured must sustain the property being measured long enough to get a reading. Human behavior varies with the circumstance and thus eludes measurement. That is not all. Some things can happen only if other things happen and how strongly they happen depends not only on the strength of those other things but also on their number and variety. How can one capture and measure something whose measurement constantly changes because it depends on a myriad other things that may also need to be measured? What kind of special scale is used to take such measurements? Money, weight, or speed? Hardly.

It is possible to measure in another fundamentally different way. Before showing how, we need to answer the one question: What use do we intend to make of the measurements? Clearly no number speaks to us directly. We must interpret what it means out of our experience or that of an expert who can interpret it for us according to the need or purpose for which it is taken. In other words, to use measurement we need experience and judgment. It appears that our own and other people's subjective perceptions are the bottom line for the meaning and the use of measurements even in spite of how people speak of the objectivity of science. Presumably, objectivity means we will arrive at the same conclusions when we start with the same assumptions if we reason logically, or we will obtain the same measurement if we all follow the same rules. But logic itself is a linear process that goes from assumption to conclusion and cannot be used to synthesize the outcome of many interdependent causes that produce effects that may themselves be part of the causes, so a new way is needed to think about and draw conclusions in such complexity.

In the end, we must depend on our collective understanding to deal with the world. Since it is expert and subjective judgment that interprets what measurements mean and how important they are, can we turn this around and instead associate numbers directly with judgments in such a way that we can derive scales whose readings tell us about the priority or relative importance of what we think we want to measure in some cases without the need for making measurements? If we can do that then we can also ask the expert to use judgment to prioritize intangibles: things or ideas for which we have no measurements. Perhaps we can deal with the meaning of all properties tangible or intangible using priorities so long as we use experienced and informed judgment. The physical sciences are based on formulas that deal with



Analytic Hierarchy Process, Fig. 1 Area example

variables and their measurements from observations that are made and on the interpretation by experts of what the numbers after they are combined in a formula mean. Can we extend the use of expert judgment in a mathematically credible way to thinking in the nonphysical sciences? Let us begin by illustrating how a process that involves expert judgments works with a simple example.

Consider a person who would like to estimate the relative area of the five geometric shapes given in Fig. 1. For the purpose of this illustration, the relative area inside each shape obtained from actual measurement by using a ruler and dividing each measurement by the sum of all five measurements is also provided. Of course, in real-life situations the relative areas would not be known to the person. He must estimate the relative sizes of the figures by comparing them in pairs. A pairwise comparison consists of identifying the figure with the smaller area of the two, and estimating numerically how many times larger the area of the larger one is than the area of the smaller one using the scale in Table 1. The smaller figure is then assigned the reciprocal value when compared with the larger one. These comparisons are arranged in a 5×5 matrix as given in Table 2. By convention, the item on the left side of the matrix is compared with that on top. If it is larger, the whole number corresponding to the judgment is put in that cell. If it is smaller, the reciprocal value is put in the cell. Finally, one derives priorities of the relative sizes of the areas from all the judgments. Table 2 also gives the estimated and actual relative areas resulting from this exercise in the last two columns. They are very close.

The Fundamental Scale

Paired comparison judgments are applied to pairs of homogeneous elements. The fundamental scale of values to represent the intensities of judgments is shown in Table 1. This scale has been validated for effectiveness, not only in many applications by a number of people, but also through theoretical comparisons with a large number of other scales.

There are many situations where elements are close or tied in measurement and the comparison must be made not to determine how many times one is larger than the other but by what fraction it is larger than the other. In other words, there are comparisons to be made between 1 and 2, and what is desired is to estimate verbally the values such as 1.1, 1.2,. . ., 1.9. There is no problem in making the comparisons by directly estimating the numbers. What is proposed here is to continue the verbal scale to make these distinctions so that 1.1 is a "tad," 1.3 indicates moderately more, 1.5 strongly more, 1.7 very strongly more, and 1.9 extremely more. This type of refinement can be used in any of the intervals from 1 to 9 and for further refinements if one needs them, for example, between 1.1 and 1.2 and so on.

An area represented on the left side of the matrix is compared with each area at the top as to how many times it is larger or more dominant. If it is not, the reciprocal value is entered in that position. The judgments are made based on feeling and converted by using the Fundamental Scale. Because the areas are visible to the eyes it is sometimes possible to express the judgments with fractional values. It is known that a small change in the numbers leads to a small change in the final priorities. The next to last priority column can be obtained as an approximation to the priorities given here by adding the numbers in each column (there are five columns) of the judgment matrix, dividing each number in the column by the total obtained, then averaging the first entries in the five columns (adding them and dividing their sum by 5), and then doing the same for the second entries in the

Analytic Hierarchy Process, Table 1 The fundamental scale

Intensity of importance	Definition	Explanation
1	Equal Importance	Two activities contribute equally to the objective
2	Weak or slight	
3	Moderate importance	Experience and judgment slightly favor one activity over another
4	Moderate plus	
5	Strong importance	Experience and judgment strongly favor one activity over another
6	Strong plus	
7	Very strong or demonstrated importance	An activity is favored very strongly over another; its dominance demonstrated in practice
8	Very, very strong	
9	Extreme importance	The evidence favoring one activity over another is of the highest possible order of affirmation
1.1–1.9	When activities are very close a decimal is added to 1 to show their difference as appropriate	A better alternative way to assigning the small decimals is to compare two close activities with other widely contrasting ones, favoring the larger one a little over the smaller one when using the 1–9 values
Reciprocals of above	If activity *i* has one of the above nonzero numbers assigned to it when compared with activity *j*, then *j* has the reciprocal value when compared with *i*	A logical assumption
Real numbers between the above integers		When appropriate according to the person making the comparisons because of special knowledge that person has
Ratios of measurements on a ratio scale		When measurements are available and one interprets their ratios to be equivalent to judgments (not usually recommended)

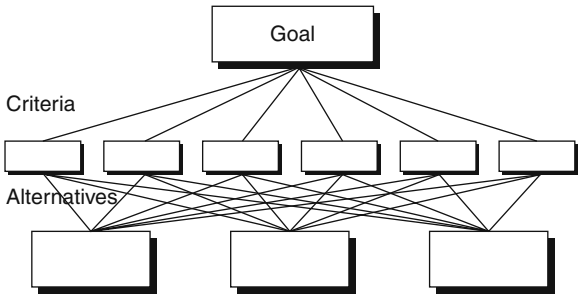
Analytic Hierarchy Process, Table 2 Pairwise comparison judgments of the different areas

Figure	Circle	Triangle	Square	Diamond	Rectangle	Priorities (eigenvectors)	Actual relative size
Circle	1	9	2	3	5	0.462	0.471
Triangle	1/9	1	1/5	1/3	1/2	0.049	0.050
Square	1/2	5	1	3/2	3	0.245	0.234
Diamond	1/3	3	2/3	1	3/2	0.151	0.149
Rectangle	1/5	2	1/3	2/3	1	0.093	0.096

five columns and so on to the fifth entries in the five columns. This only illustrates the use of judgments in a problem where the answer is known to give a little credibility to the idea that it may also work with things that are not visible, but one can think about them and so on.

It has been observed that in general comparisons with respect to the dominance of one object over another with respect to a certain attribute or criterion take three forms: importance or significance that includes all kinds of influence, physical measurements in science, engineering, and economics fall under this category; preference as in making decisions; and likelihood as in probabilities. If there is adequate knowledge, one can compare anything with anything else that shares a common attribute or criterion. Thus, comparisons go beyond ordinary measurement to include intangibles for which there are no scales of measurement.

The Analytic Hierarchy Process (AHP) is a general theory of measurement. It is used to derive relative



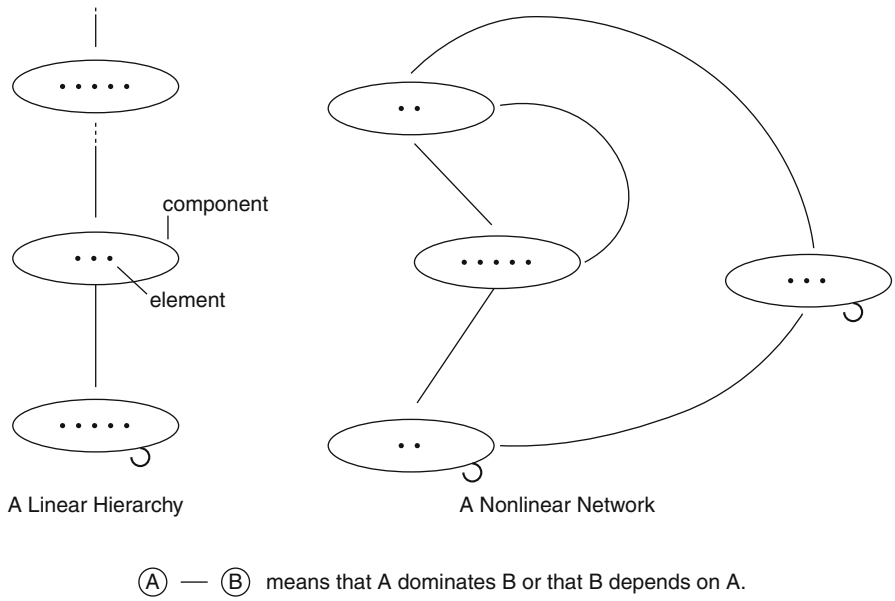
Analytic Hierarchy Process, Fig. 2 A three-level hierarchy

scales of absolute numbers from both discrete and continuous paired comparisons in multilevel hierarchic structures. These comparisons may be taken from actual measurements or from a fundamental scale that reflects the relative strength of preferences and feelings. The AHP has a special concern with departure from consistency and the measurement of this departure, and with dependence within and between the groups of elements of its structure. It has found its widest applications in multicriteria decision making, in planning and resource allocation, and in conflict resolution (Saaty and Alexander 1989). In its general form, the AHP is a nonlinear framework for carrying out both deductive and inductive thinking without use of the syllogism by taking several factors into consideration simultaneously and allowing for dependence and for feedback, and making numerical trade-offs to arrive at a synthesis or conclusion (Figs. 2 and 3).

For a long time, people have been concerned with the measurement of both physical and psychological events. By physical it is meant the realm of what is fashionably known as the tangibles insofar as they constitute some kind of objective reality outside the individual conducting the measurement. By contrast, psychological means the realm of the intangibles, comprising the subjective ideas, feelings, and beliefs of the individual and of society as a whole. The question is whether there is a coherent theory that can deal with both these worlds of reality without compromising either. The AHP is a method that can be used to establish measures in both the physical and social domains.

From the neurological sciences, it is known that sense data are mixed with temperature and other information by the thalamus, before they are recorded in memory. In the end, what we sense is what we are and not fully what is out there. Performance tests indicate that an individual not experienced in ranking objects according to size may well say that one apple which is three times larger than another apple is the same as the smaller one. Only by being exposed to many apples and asked to make careful distinctions in size will the individual begin to show an improved ability to sort and rank apples according to size. What the person does is to adjust sensation and impression with what he or she observes. It is not the real apples that one compares, but the impressions one forms about them. One needs such real experiences to institute early in one's mind the possibility of comparing things in pairs. This applies equally to more abstract ideas and their relative importance to a higher-order property or goal. He or she would then be able to say that one idea is more important than another in terms of the satisfaction of the goal and whether, according to his or her understanding and experience, it is much more important or slightly more important. The lesser of the two is always used as the unit in terms of which the more important one is compared as to how much more important it is, and also how many times more, because the feeling of importance is converted to magnitudes on numerous sense experiences. Thus, there is transfer from the concrete to the abstract, so that the two can be combined to make trade-offs when needed, which happens frequently in daily experience. It is not possible to compare the lesser element with the greater one, because it must first be used as a unit to determine the magnitude of the greater one. Therefore, there is bias in human thinking in using the smaller of two elements as the unit. It is impossible a priori to ask how much less the smaller element is than the larger without first involving it as the unit of measurement. Thus, priorities of many objects can only be derived on the basis of dominance, and their reciprocal is automatically calculated to determine, in a meaningful way, the relative priorities of being "dominated."

In using the AHP to model a problem, one needs a hierarchic or network structure to represent that problem, as well as pairwise comparisons to establish relations within the structure. In the discrete case, these comparisons lead to dominance matrices and in the continuous case to kernels of Fredholm operators (Saaty and Vargas 1993), from which ratio scales are derived in the form of principal eigenvectors or



Analytic Hierarchy Process, Fig. 3 Structural difference between a linear and a nonlinear network

eigenfunctions, as the case may be. These matrices, or kernels, are positive and reciprocal, for example, $a_{ij} = 1/a_{ji}$. In particular, special effort has been made to characterize these matrices (1993). Because of the need for a variety of judgments, there has also been considerable work done to deal with the process of synthesizing group judgments (Saaty 1994). The axiomatic foundations of the AHP may be found in Saaty (1986).

Absolute and Relative Measurement and Structural Information

Cognitive psychologists have recognized for some time that there are two kinds of comparisons: absolute and relative. In absolute comparisons, alternatives are compared with a standard in one's memory that has been developed through experience; in relative comparisons, alternatives are compared in pairs according to a common attribute. The AHP has been used with both types of comparisons to derive ratio scales of measurement. Such scales are called absolute and relative measurement scales. Relative measurement w_i, $i = 1,\ldots, n$, of each of n elements is a ratio scale of values assigned to that element and derived by comparing it in pairs with the others. In paired comparisons, two elements i and j are compared with respect to a property they have in common. The smaller i is used as the unit and the larger j is estimated as a multiple of that unit in the form $(w_i/w_j)/1$ where the ratio w_i/w_j is taken from a fundamental scale of absolute values.

Absolute measurement (sometimes called scoring) is applied to rank the alternatives in terms of the criteria or else in terms of ratings (or intensities) of the criteria, for example, excellent, very good, good, average, below average, poor, and very poor, or A, B, C, D, E, F, and G. After setting priorities for the criteria (or subcriteria, if there are any), pairwise comparisons are also made between the ratings themselves to set priorities for them under each criterion and dividing their priorities each by the largest rated intensity (the ideal intensity). Finally, alternatives are scored by checking off their respective ratings under each criterion and summing these ratings for all the criteria. This produces a ratio scale score for the alternative. The scores thus obtained of the alternatives can in the end be normalized by dividing each one by their sum.

Absolute measurement has been used to rank cities in the United States according to nine criteria as judged by six different people (Saaty 1986). Another appropriate use for absolute measurement is that of schools admitting students (Saaty et al. 1991). Most schools set their criteria for admission independently of the performance of the current crop of students seeking admission. Their priorities are then used to determine whether a given student meets the standard set for qualification. In that case absolute measurement should be used to determine which students qualify for admission.

Comments on Cost-Benefit Analysis

Often, the alternatives from which a choice must be made in a choice-making situation have both costs and benefits associated with them. In this case, it is useful to construct separate costs and benefits hierarchies, with the same alternatives on the bottom level of each. Thus one obtains both a cost-priority vector and a benefit-priority vector. The cost-benefit vector is obtained by taking the ratio of the benefit priority to the costs priority for each alternative, with the highest such ratio indicating the preferred alternative. In the case where resources are allocated to several projects, such cost-to-benefit ratios or the corresponding marginal ratios prove to be very valuable.

Wrong decisions may be made in some cases where only one structure is used for the purpose of generating priorities for the alternatives. In general, one needs two or more of four separate structures: one for benefits, one for costs, one for opportunities, and a fourth for risks. Because one must ask what dominates what in the paired comparisons and by how much (homogeneous elements with clusters and pivots are used for widely spread alternatives), in the end one multiplies the benefits of each alternative by the opportunities it creates and divides by the costs times the risks.

For example, in evaluating three types of copying machines, one represents in the benefits hierarchy the good attributes one is looking for and one represents in the costs hierarchy the pain and economic costs that one would incur in buying or maintaining the three types of machines. Note that the criteria for benefits and the criteria for costs need not be simply opposites of each other but may be totally different. Also note that each criterion may be regarded at a different threshold of intensity and that such thresholds may themselves be prioritized according to desirability, with each alternative evaluated only in terms of its highest priority threshold level. Similarly, three hierarchies can be used to assess a benefit/(cost × risk) outcome.

The Eigenvector Solution for Weights and Consistency

There is an infinite number of ways to derive the vector of priorities from the matrix $\boldsymbol{A} = (a_{ij})$, but emphasis on consistency leads to an eigenvalue formulation.

If a_{ij} represents the importance of alternative i over alternative j and a_{jk} represents the importance of alternative i over alternative j and a_{ik}, the importance of alternative i over alternative k must equal $a_{ij}a_{jk}$ for the judgments to be consistent. Without a scale at all, or not available conveniently, as in the case of some measuring devices, one cannot give the precise values of $a_{ij} = w_i/w_j$ but only an estimate. The problem becomes $\mathrm{A}'\mathrm{w}' = \lambda_{\max}w'$ where $\lambda_{\max}$ is the largest or principal eigenvalue of $A' = \left(a'_{ji}\right)$, the perturbed value of $\boldsymbol{A} = (a_{ij})$, with $a'_{ji} = 1/a_{ij}$ forced. To simplify the notation, write $\boldsymbol{A}\boldsymbol{w} = \lambda_{\max}w$ where $\boldsymbol{A}$ is the matrix of pairwise comparisons.

The solution is obtained by raising the matrix to a sufficiently large power, then summing over the rows and normalizing to obtain the priority vector $w = (w_1,\ldots, w_n)$. The process is stopped when the difference between components of the priority vector obtained at the kth power and at the $(k + 1)$st power is less than some predetermined small value.

An easy way to get an approximation to the priorities is to normalize the geometric means of the rows. This result coincides with the eigenvector for $n \leq 3$. A second way to obtain an approximation is by normalizing the elements in each column of the judgment matrix and then averaging over each row.

For important applications one should use only the eigenvector derivation procedure because approximations can lead to rank reversal in spite of the closeness of the result to the eigenvector. It is easy to prove that for an arbitrary estimate x of the priority vector

$$\lim_{k\to\infty} \frac{1}{\lambda_{\max}^k}\boldsymbol{A}^k x = cw$$

where c is a positive constant and w is the principle eigenvector of $\boldsymbol{A}$. This may be interpreted roughly to say that if we begin with an estimate and operate on it successively by $\boldsymbol{A}/\lambda_{\max}$ to get new estimates, the result converges to a constant multiple of the principal eigenvector.

A simple way to obtain the exact value (or an estimate) of $\lambda_{\max}$ when the exact value (or an estimate) of w is available in normalized form is to add the columns of $\boldsymbol{A}$ and multiply the resulting vector by the vector w. The resulting number is $\lambda_{\max}$ (or an estimate). This follows from

Analytic Hierarchy Process, Table 3 Random consistency index

n	1	2	3	4	5	6	7	8	9	10
Random Consistency Index (R.I.)	0	0	0.52	0.89	1.11	1.25	1.35	1.40	1.45	1.49

$$\sum_{j=1}^{n} a_{ij}w_j = \lambda_{\max}w_i \quad \text{and} \quad \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}w_j = \sum_{j=1}^{n}\left(\sum_{i=1}^{n} a_{ij}\right) w_j = \sum_{i=1}^{n} \lambda_{\max}w_i = \lambda_{\max}$$

The problem now is how good is the principal eigenvector estimate w. Note that if one obtains $w = (w_1,\ldots, w_n)^{\mathrm{T}}$, by solving this problem, the matrix whose entries are w_i/w_j is a consistent matrix which is our consistent estimate of the matrix $\boldsymbol{A}$. The original matrix $\boldsymbol{A}$ itself need not be consistent. In fact, the entries of $\boldsymbol{A}$ need not even be transitive; that is, $\boldsymbol{A}_1$ may be preferred to $\boldsymbol{A}_2$ and $\boldsymbol{A}_2$ to $\boldsymbol{A}_3$ but $\boldsymbol{A}_3$ may be preferred to $\boldsymbol{A}_1$. What is desired is a measure of the error due to inconsistency. It turns out that $\boldsymbol{A}$ is consistent if and only if $\lambda_{\max} = n$, and that the quantity $\lambda_{\max} - n$ is always available. This suggests using $\lambda_{\max} - n$ as an index of departure from consistency. But

$$\lambda_{\max} - n = -\sum_{i=2}^{n} \lambda_i;\ \lambda_{\max} = \lambda_1$$

where λ_i, $i = 1,\ldots, n$ are the eigenvalues of $\boldsymbol{A}$. Thus, the index adopted is the average value $(\lambda_{\max} - n)/(n-1)$, which is the (negative) average of λ_i, $i = 2,\ldots, n$ (some of which may be complex conjugates).

It is interesting to note that $(\lambda_{\max} - n)/(n-1)$ is the variance of the error incurred in estimating a_{ij} . This can be shown by writing

$$a_{ij} = (w_i/w_j)\varepsilon_{ij},\ \varepsilon_{ij} > 0 \quad \text{and} \quad \varepsilon_{ij} = 1 + \delta_{ij},\ \delta_{ij} > -1$$

and substituting in the expression for $\lambda_{\max}$. It is δ_{ij} that is of interest, as the error component and its value $|\delta_{ij}| < 1$ for an unbiased estimator. The measure of inconsistency can be used to successively improve the consistency of judgments.

The consistency index of a matrix of comparisons is given by $\mathrm{CI} = (\lambda_{\max} - n)/(n-1)$. The consistency ratio (CR) is obtained by comparing the CI with the appropriate one of the following set of numbers each of which is an average random consistency index (RI) derived from a sample of size 500 of randomly generated reciprocal matrices using the scale 1/9, 1/8, . . .,1, . . ., 8,9 (Table 3). A CR = CI/RI less than or equal to 0.10 is considered acceptable. If CR is larger than 0.10, the problem should be reanalyzed and the judgments revised.

The consistency index for an entire hierarchy is defined by

$$C_H = \sum_{j=1}^{j}\sum_{i=1}^{n_{i_{j+1}}} w_{ij}\mu_{ij+1}$$

where $w_{ij} = 1$ for $j = 1$, and $n_{\ i\ j-1}$ is the number of elements of the $(j + 1)$st level with respect to the ith criterion of the jth level.

Let $|C_k|$ be the number of elements of C_k and let w_{hk} be the priority of the impact of the hth component on the kth component, that is, $w_{hk} - w_k(C_h)$ or w_k: $C_h \to w_{hk}$.

Labeling the components of a system along lines similar to those followed for a hierarchy and denoting by w_{jk} the limiting priority of the jth element in the kth component,

$$C_s = \sum_{k=1}^{s}\sum_{j=1}^{n_k} w_{jk}\sum_{h=1}^{|C_k|} w_{hk}\mu_k(j,h)$$

where $\mu_k\,(j, h)$ is the consistency index of the pairwise comparison matrix of the elements in the kth component with respect to the jth element in the hth component.

How to Structure a Hierarchy

Perhaps the most creative part of decision making that has a significant effect on the outcome is the structuring of the decision as a hierarchy. The basic principle to follow in creating this structure is always to see if one can answer the following question: "Can

I compare the elements on a lower level in terms of some or all of the elements on the next higher level?"

A useful way to proceed is to come down from the goal as far as one can and then go up from the alternatives until the levels of the two processes are linked in such a way as to make comparison possible. Here are some suggestions for an elaborate design.

1. Identify overall goal. What are you trying to accomplish? What is the main question?
2. Identify subgoals of overall goal. If relevant, identify time horizons that affect the decision.
3. Identify criteria that must be satisfied to fulfill subgoals of the overall goal.
4. Identify subcriteria under each criterion. Note that criteria or subcriteria may be specified in terms of ranges of values of parameters or in terms of verbal intensities such as high, medium, low.
5. Identify actors involved.
6. Identify actor goals.
7. Identify actor policies.
8. Identify options or outcomes.
9. For yes-no decisions take the most preferred outcome and compare benefits and costs of making the decision with those of not making it.
10. Do cost-benefit analysis using marginal values. Because we are dealing with dominance hierarchies, ask which alternative yields the greatest benefit, and for costs, which alternative costs the most.

Software programs such as *superdecisions* and *Expert Choice* incorporate the AHP methodology and enable the analyst to structure the hierarchy and resolve the problem using relative or absolute measurements, as appropriate.

Hierarchic Synthesis and Rank

Hierarchic synthesis is obtained by a process of weighting and adding down the hierarchy leading to a multilinear form. The hierarchic composition principle is a theorem in the AHP that is a particular case of network composition which deals with the cycles and loops of a network.

What happens to the synthesized ranks of alternatives when new ones are added or old ones deleted? The ranks cannot change under any single criterion, but they can under several criteria depending on whether one wants the ranks to remain the same or allows them to change. Many examples are given in the literature showing that allowing rank to change is natural. In 1990, Tversky et al. concluded that the primary cause of preference reversal is the "failure of procedure invariance." In the AHP there is no such methodological constraint (Tversky et al. 1990).

In the distributive mode of the AHP, the principal eigenvector is normalized to yield a unique estimate of a ratio scale underlying the judgments. This mode allows rank to change and is useful when there is dependence on the number of alternatives present or on dominant new alternatives which may affect preference among old alternatives thus causing rank reversals (see phantom alternatives—Saaty 1993). In the ideal mode of the AHP, the normalized values of the alternatives for each criterion are divided by the value of the highest rated alternative. In this manner, a newly added alternative that is dominated everywhere cannot cause reversal in the ranks of the existing alternatives (Saaty 1994).

Examples

Relative Measurement: Choosing the Best House

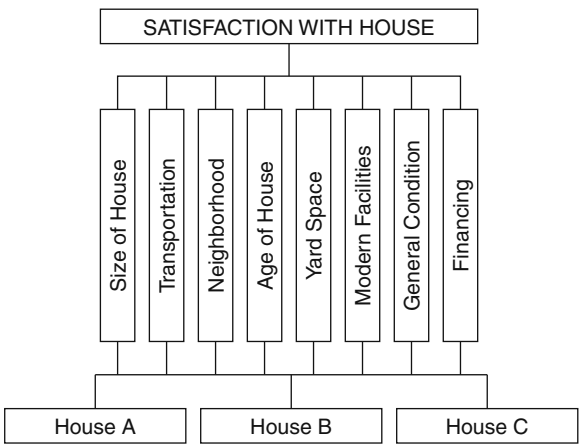
When a family of average income is being advised on buying a house, the family identifies eight factors that they think they have to look for in the house. These factors fall into three categories: economic, geographic, and physical. Although one might begin by examining the relative importance of these clusters, the family feels they want to prioritize the relative importance of all the factors without working with clusters. The problem is to decide which of three candidate houses to choose. In applying the AHP, the first step is decomposition, or the structuring of the problem into a hierarchy (Fig. 4). On the first (or top) level is the overall goal of Satisfaction with House. On the second level are the eight factors or criteria that contribute to the goal, and on the third (or bottom) level are the three candidate houses that are to be evaluated in terms of the criteria on the second level. The definitions of the factor and the pictorial representation of the hierarchy follow.

The factors important to the individual family are:

1. *Size of house*: Storage space, size of rooms, number of rooms, total area of house

2. *Transportation*: Convenience and proximity of bus service
3. *Neighborhood*: Degree of traffic, security view, taxes, physical condition of surrounding buildings
4. *Age of house*: Self-explanatory
5. *Yard space*: Includes front, back, and side space, and space shared with neighbors
6. *Modern facilities*: Dishwashers, garbage disposals, air-conditioning, alarm system, and other such items
7. *General condition*: Extent to which repairs are needed; condition of walls, carpet, drapes, wiring; cleanliness
8. *Financing*: Availability of assumable mortgage, seller financing, or bank financing

The next step is comparative judgment. Arrange the elements on the second level into a matrix and elicit from the people buying the house judgments about the relative importance of the elements with respect to the overall goal, Satisfaction with House.



Analytic Hierarchy Process, Fig. 4 Decomposition of the problem into a hierarchy

The questions to ask when comparing two criteria are of the following kind: Of the two alternatives being compared, which is considered more important by the family buying the house and how much more important is it with respect to family satisfaction with the house, which is the overall goal?

The matrix of pairwise comparisons of the factors given by the home buyers in this case is shown in Table 4, along with the resulting vector of priorities. The judgments are entered using the Fundamental Scale, first verbally as indicated in the scale and then associating the corresponding number. The vector of priorities is the principal eigenvector of the matrix. This vector gives the relative priority of the factors measured on a ratio scale. That is, these priorities are unique to within-a-positive-similarity transformation. However, if one insures that they add up to unity, they are always unique. In this case financing has the highest priority, with 33% of the influence.

In Table 4, instead of naming the criteria, use the number previously associated with each.

Now consider the pairwise comparisons of the houses on the bottom level, comparing them pairwise with respect to how much better one is than the other in satisfying each criterion on the second level. Thus there are eight 3×3 matrices of judgments since there are eight elements on level two, and three houses to be pairwise compared for each element. The matrices (Table 5) contain the judgments of the family involved. In order to facilitate understanding of the judgments, a brief description of the houses is given below.

House A: This house is the largest of them all. It is located in a good neighborhood with little traffic and low taxes. Its yard space is comparably larger than that of houses B and C. However, its general condition is not very good and it needs cleaning and painting.

Analytic Hierarchy Process, Table 4 Pairwise comparison matrix for level 1

	1	2	3	4	5	6	7	8	Priority vector
1	1	5	3	7	6	6	1/3	1/4	0.173
2	1/5	1	1/3	5	3	3	1/5	1/7	0.054
3	1/3	3	1	6	3	4	6	1/5	0.188
4	1/7	1/5	1/6	1	1/3	1/4	1/7	1/8	0.018
5	1/6	1/3	1/3	3	1	1/2	1/5	1/6	0.031
6	1/6	1/3	1/4	4	2	1	1/5	1/6	0.036
7	3	5	1/6	7	5	5	1	1/2	0.167
8	4	7	5	8	6	6	2	1	0.333
	$\lambda_{max} = 9.669$ C.I. $= 0.238$ C.R. $= 0.169$								

Analytic Hierarchy Process, Table 5 Pairwise comparison matrix for level 1

Size of house	A	B	C	Normalized priorities	Idealized priorities	Yard space	A	B	C	Normalized priorities	Idealized priorities	
A	1	6	8	0.754	1.000	A	1	5	4	0.674	1.000	
B	1/6	1	4	0.181	0.240	B	1/5	1	1/3	0.101	0.150	
C	1/8	1/4	1	0.065	0.086	C	1/4	3	1	0.226	0.335	
$\lambda_{max} = 3.136$	C.I. = 0.068			C.R. = 0.117		$\lambda_{max} = 3.086$	C.I. = 0.043					C.R. = 0.074
Transportation	A	B	C	Normalized priorities	Idealized priorities	Modern facilities	A	B	C	Normalized priorities	Idealized priorities	
A	1	7	1/5	0.233	0.327	A	1	8	6	0.747	1.000	
B	1/7	1	1/8	0.005	0.007	B	1/8	1	1/5	0.060	0.080	
C	5	8	1	0.713	1.000	C	1/6	5	1	0.193	0.258	
$\lambda_{max} = 3.247$			C.I. = 0.124		C.R. = 0.213	$\lambda_{max} = 3.197$	C.I. = 0.099					C.R. = 0.170
Neighborhood	A	B	C	Normalized priorities	Idealized priorities	General condition	A	B	C	Normalized priorities	Idealized priorities	
A	1	8	6	0.745	1.000	A	1	1/2	1/2	0.200	0.500	
B	1/8	1	1/4	0.065	0.086	B	2	1	1	0.400	1.000	
C	1/6	4	1	0.181	0.240	C	2	1	1	0.400	1.000	
$\lambda_{max} = 3.130$			C.I. = 0.068		C.R. = 0.117	λ_{max} 53.000	C.I. = 0.000					C.R. = 0.000
Age of house	A	B	C	Normalized priorities	Idealized priorities	Financing	A	B	C	Normalized priorities	Idealized priorities	
A	1	1	1	0.333	1.000	A	1	1/7	1/5	0.072	0.111	
B	1	1	1	0.333	1.000	B	7	1	3	0.650	1.000	
C	1	1	1	0.333	1.000	C	5	1/3	1	0.278	0.428	
$\lambda_{max} = 3.000$			C.I. = 0.000		C.R. = 0.000	$\lambda_{max} = 3.065$	C.I. = 0.032					C.R. = 0.056

Analytic Hierarchy Process, Table 6 Synthesis

Distributive Mode										
	1 (0.173)	2 (0.054)	3 (0.188)	4 (0.018)	5 (0.031)	6 (0.036)	7 (0.167)	8 (0.333)		
A	0.754	0.233	0.754	0.333	0.674	0.747	0.200	0.072		0.396
B	0.181	0.055	0.065	0.333	0.101	0.060	0.400	0.650	=	0.341
C	0.065	0.713	0.181	0.333	0.226	0.193	0.400	0.278		0.263
Ideal Mode										
A	1.00	0.327	1.00	1.00	1.00	1.00	0.500	0.111		0.584
B	0.240	0.007	0.086	1.00	0.150	0.080	1.00	1.00	=	0.782
C	0.086	1.00	0.240	1.00	0.335	0.258	1.00	0.428		0.461

Also, the financing is unsatisfactory because it would have to be financed through a bank at a high rate of interest.

House B: This house is a little smaller than House A and is not close to a bus route. The neighborhood gives one the feeling of insecurity because of traffic conditions. The yard space is fairly small and the house lacks the basic modern facilities. On the other hand, its general condition is very good. Also an assumable mortgage is obtainable, which means the financing is good with a rather low interest rate. There are several copies of B in the neighborhood.

House C: House C is very small and has few modern facilities. The neighborhood has high taxes, but is in good condition and seems secure. The yard space is bigger than that of House B, but is not comparable to House A's spacious surroundings. The general condition of the house is good, and it has a pretty carpet and drapes. The financing is better than for A but not better than for B.

Table 5 gives the matrices of the houses and their local priorities with respect to the elements on level two.

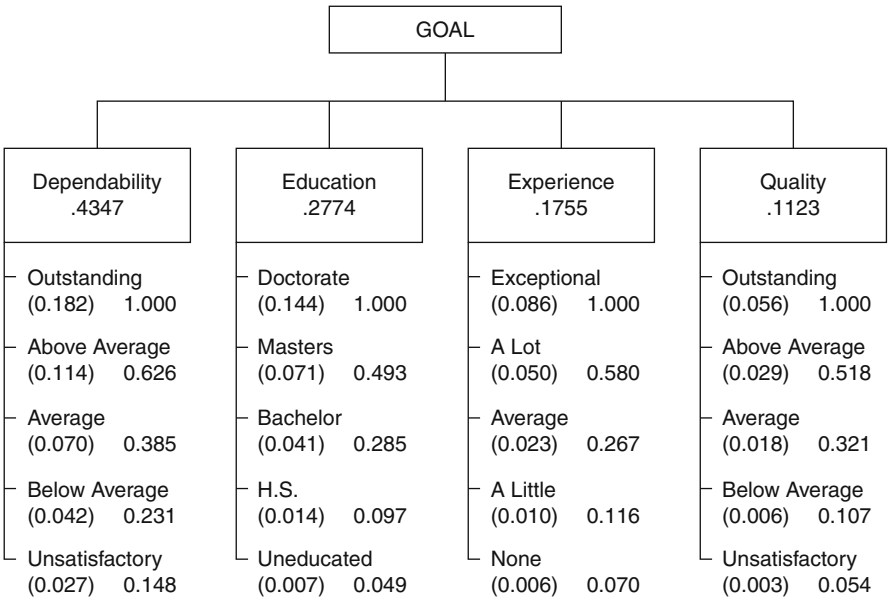
The next step is to synthesize the priorities. In order to establish the composite or global priorities of the houses, lay out in a matrix (Table 6) the local priorities of the houses with respect to each criterion and multiply each column of vectors by the priority of the corresponding criterion and add across each row, which results in the composite or global priority vector of the houses. Under the distributive mode, House A is preferred if, for example, copies of B matter. Under the ideal mode, House B is the preferred house if the family wanted the best house regardless of other houses and how many copies of it there are in the neighborhood. In a large number of situations with ten criteria and three alternatives, the two modes gave the same best choice 92% of the time (Saaty 1994).

Absolute Measurement: Evaluating Employees for Raises

Employees are evaluated for raises. The criteria for Dependability, Education, Experience, and Quality. Each criterion is subdivided into intensities, standards, or subcriteria as shown in Fig. 5. Priorities are set for the criteria by comparing them in pairs, and these priorities are then given in a matrix. The intensities are then pairwise compared according to priority with respect to their parent criterion (as in Table 7) and their priorities are divided by the largest intensity for each criterion (second column of priorities in Fig. 5). Finally, each individual is rated in Table 8 by assigning the intensity rating that applies to him or her under each criterion. The scores of these subcriteria are weighted by the priority of that criterion and summed to derive a total ratio scale score for the individual. This approach can be used whenever it is possible to set priorities for intensities of criteria, which is usually possible when sufficient experience with a given operation has been accumulated.

Concluding Remarks

The Analytic Hierarchy Process is widely applied in business and government on a global scale. Applications include strategic planning, R&D and innovation, capital planning, IT and product portfolio management, trade studies, vendor selection, and site selection, to name a few. Notable government organizations applying AHP include the US Joint Chiefs of Staff; US Navy, Air Force, and Army;



Analytic Hierarchy Process, Fig. 5 Employee evaluation hierarchy

Analytic Hierarchy Process, Table 7 Ranking intensities

	Outstanding	Above average	Average	Below average	Unsatisfactory	Priorities
Outstanding	1.0	2.0	3.0	4.0	5.0	0.419
Above average	1/2	1.0	2.0	3.0	4.0	0.263
Average	1/3	1/2	1.0	2.0	3.0	0.630
Below average	1/4	1/3	1/2	1.0	2.0	0.097
Unsatisfactory	1/5	1/4	1/3	1/2	1.0	0.062
Inconsistency Ratio = 0.015						

Analytic Hierarchy Process, Table 8 Ranking alternatives

	Dependability 0.4347	Education 0.2774	Experience 0.1775	Quality 0.1123	Total
1. Adams, V	Outstanding	Bachelor	A little	Outstanding	0.646
2. Becker, L	Average	Bachelor	A little	Outstanding	0.379
3. Hayat, F	Average	Masters	A lot	Below average	0.418
4. Kesselman, S	Above average	H.S.	None	Above average	0.369
5. O'Shea, K	Average	Doctorate	A lot	Above average	0.605
6. Peters, T	Average	Doctorate	A lot	Average	0.583
7. Tobias, K	Above average	Bachelor	Average	Above average	0.456

every major intelligence agency; and US Department of Agriculture, among other civilian agencies. In the commercial arena it is used by Johnson & Johnson across a number of their operating businesses, Siemens, Pfizer, Boeing, and the NFL (player selection), among others.

See

- ▶ Analytic Network Process
- ▶ Decision Analysis
- ▶ Multi-attribute Utility Theory
- ▶ Utility Theory

References

Falcone, D., De Felice, F., & Saaty, T. L. (2009). *Il decision making e i decisionali Multicriterio: Le metodologie AHP e ANP*. Milano: HOPELI.

Saaty, T. L. (1986). Axiomatic foundation of the analytic hierarchy process. *Management Science, 32*, 841–855.

Saaty, T. L. (1993). What is relative measurement? the ratio scale phantom. *Mathematical and Computer Modelling, 17* (4–5), 1–12.

Saaty, T. L. (1994). *Fundamentals of decision making and priority theory*. Pittsburgh, PA: RWS Publications.

Saaty, T. L. (1996). *The analytic network process: Decision making with dependence and feedback*. Pittsburgh, PA: RWS Publications. Revised (2001).

Saaty, T. L. (1997). *Decision making with dependence and feedback: The analytic network process (ANP) and super decisions software, guide, manual and examples*. Pittsburgh, PA: RWS Publications.

Saaty, T. L. (2000a). *Fundamentals of decision making with the analytic hierarchy process, paperback*. Pittsburgh, PA: RWS Publications. Original edition 1994.

Saaty, T. L. (2000b). *The brain, unraveling the mystery of how it works: The neural network process*. Pittsburgh, PA: RWS Publications.

Saaty, T. L. (2005). *Theory and applications of the analytic network process*. Pittsburgh, PA: RWS Publications.

Saaty, T. L. (2010). *Principia Mathematica Decernendi: Mathematical principles of decision making*. Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Alexander, J. (1989). *Conflict resolution*. New York: Praeger.

Saaty, T. L., & Cillo, B. (2008). *The encyclicon* (Vol. 2). Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Foreman, E. H. (1993). *The Hierarchon – a dictionary of hierarchies*. Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Kearns, K. P. (1985). *Analytical planning—the organization of systems* (International series in modern applied mathematics and computer science, Vol. 7). Oxford: Pergamon Press.

Saaty, T. L., & Ozdemir, M. (2005). *The encyclicon*. Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Peniwati, K. (2008). *Group decision making: Drawing our and reconciling differences*. Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Vargas, L. G. (1982). *The logic of priorities: Applications in business energy health, transportation*. Norwell, MA: Kluwer.

Saaty, T. L., & Vargas, L. G. (1991). *Prediction, projection and forecasting*. Norwell, MA: Kluwer.

Saaty, T. L., & Vargas, L. G. (1993). A model of neural impulse firing and synthesis. *Journal of Mathematical Psychology, 37*, 200–219.

Saaty, T. L., & Vargas, L. G. (1994). *Decision making in economic, political, social and technological environments: The analytic hierarchy process, paperback*. Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Vargas, L. G. (2000). *Models methods, concepts and applications of the analytic hierarchy process*. Boston, MA: Kluwer.

Saaty, T. L., & Vargas, L. G. (2006). *Decision making with the analytic network process: Economic, political, social and technological applications with benefits, opportunities, costs and risks*. New York: Springer.

Saaty, T. L., France, J. W., & Valentine, K. R. (1991). Modeling the graduate business school admissions process. *Socio-Economic Planning Sciences, 25*, 155–162.

Tversky, A., Slovic, P., & Kahneman, D. (1990). The causes of preference reversal. *American Economic Review, 80*, 204–215.

Analytic Network Process

Thomas L. Saaty
University of Pittsburgh, Pittsburgh, PA, USA

Introduction

There are at least five important criteria that a reliable decision theory should satisfy. They are: (1) having the potential to cope with full fledged complexity, including scenarios, stakeholders, criteria, subcriteria and the like; (2) validation in practice through prediction of decision outcomes in complex situations; (3) measuring intangible factors along similar lines as the theory does with tangibles; (4) including the possibility to deal with dependence and feedback among the elements of the decision; and (5) allowing for group decision making within the assumptions and mathematical workings of the theory by including the power of each member of a group in a scientifically justified way and not simply through consensus or other arbitrarily chosen criteria.

The Analytic Network Process (ANP) is a new theory that extends the Analytic Hierarchy Process (AHP) to cases of dependence and feedback and generalizes on the supermatrix approach introduced by Saaty for the AHP. It allows interaction and feedback within clusters (inner dependence) and between clusters (outer dependence). Feedback can better capture the complex effects of interplay in human society. The ANP provides a thorough framework to include clusters of elements connected in any desired way to investigate the process of deriving ratio scale priorities from the distribution of influence among elements and among clusters. The AHP becomes a special case of the ANP. Although many decision problems are best studied through the

ANP, it is not true that forcing an ANP model always yields better results than using the hierarchies of the AHP. There are examples to justify the use of both. It is not yet known when the shortcut of the hierarchy is justified, not simply on grounds of expediency and efficiency, but also for reasons of validity of the outcome.

The ANP is a coupling of two parts. The first consists of a control hierarchy or network of criteria and subcriteria that control the interactions in the system under study. The second is a network of influences among the elements and clusters. The network varies from criterion to criterion and a supermatrix of limiting influence is computed for each control criterion. Finally, each of these supermatrices is weighted by the priority of its control criterion and the results are synthesized through addition for all the control criteria.

In addition, a problem is often studied through a control hierarchy or system of benefits, a second for costs, a third for opportunities, and a fourth for risks. The synthesized results of the four control systems are combined by taking the quotient of the benefits times the opportunities to the costs times the risks to determine the best outcome.

Feedback Networks

Many decision problems cannot be structured hierarchically because they involve the interaction and dependence of higher-level elements on lower-level elements. Not only does the importance of the criteria determine the importance of the alternatives as in a hierarchy, but also the importance of the alternatives themselves determines the importance of the criteria.

The feedback structure does not have the linear top-to-bottom form of a hierarchy but looks more like a network, with cycles connecting its clusters of elements, which can no longer be called levels, and with loops that connect a cluster to itself. A decision problem involving feedback arises often in practice. It typically has many interactions, which in the limit converge toward the goal. Our minds need a tool to manage this complexity. The ANP provides that tool.

At present, in their effort to simplify and deal with complexity, people who work in decision making use mostly very simple hierarchic structures consisting of a goal, criteria, and alternatives. Yet, not only are decisions obtained from a simple hierarchy of three levels different from those obtained from a multilevel hierarchy, but also decisions obtained from a network may be different from those obtained from a more complex hierarchy. The question is: How much would one like to trade off the effort in creating and following through an elaborate structure against the desired degree of accuracy of the outcome? We one cannot collapse complexity artificially into a simplistic structure of two levels, criteria and alternatives, and hope to capture the outcome of interactions in the form of highly condensed judgments that correctly reflect all that goes on in the world. We one must learn to decompose these judgments through more elaborate structures and organize our reasoning and calculations in sophisticated but simple ways to serve our understanding of the complexity around us. Experience indicates that it is not very difficult to do this, although it takes more time and effort. Indeed, one must use feedback networks to arrive at the kind of decisions needed to cope with the future.

We will lay out the theoretical foundations for the kinds of structures and matrices of derived ratio scales associated with feedback networks from which we obtain the priorities for a decision. Let us summarize what we will do in anticipation of what will be coming later on. Each ratio scale, derived from a paired comparison matrix, is appropriately introduced as part of a column in a matrix to represent the impact of elements in a component on an element in another component (outer dependence) or on elements of the component itself (inner dependence). Not every element of a component need impact an element in another component. In that case those elements that make no impact are given a zero value for their contribution. The resulting matrix of components with their elements displayed vertically on the left side of the matrix and horizontally at the top of the matrix must be stochastic (each column sums to one) to obtain meaningful limiting results. To ensure that this matrix, called the supermatrix, is stochastic, we need to compare the components themselves (rather than their elements) that are on the left with respect to their impact on each component at the top according to an attribute represented in a separate control hierarchy for that system. The resulting priorities of the

components are then used to weight the column vectors. Each block of column vectors defines an entry of the supermatrix. All the column vectors in the block are multiplied by the single priority of the corresponding component on the left. The process is repeated by deriving a vector of impact of all components on the left on each component at the top. The columns of the supermatrix corresponding to the impacts on the elements of the component at the top now sum to one. The resulting supermatrix is column stochastic. What is desired, is the long-run or limiting priority of impact of each element on every other element.

Contributions to this impact are obtained in many ways. They are be obtained directly from the matrix or indirectly for any two elements by taking the impact of the first on some third element and then multiplying it by the impact of that element on the second. One must consider every such possibility of a third element. All such possibilities are obtained from the square of the matrix. Again the impact can be obtained by considering a third element that impacts a fourth element, which in turn impacts the second element. All such impacts are obtained from the cubic power of the matrix, and so on. Thus we have an infinite number of impact matrices: the matrix itself, its square, its cube, etc. If we sum all these matrices, does the result converge? Does the limit exist, and how do we compute it to obtain the desired priorities? The supermatrix may not be positive and may have zeros in certain positions where there is no direct impact of an element on another. Alternatively, the matrix may be positive or may become positive after raising it to powers. What theory do we have to deal with this problem?

Note that if the matrix is positive or if, after raising it to some power, it becomes positive, it turns out that one can obtain a unique answer. But when no power of the matrix is strictly positive, we need to examine what happens closely because even in those situations where every element can be reached from every other element, we may not have a unique limit. For example, powers of the matrix may oscillate, and different limits are obtained. Also, if it is not possible to reach every element from every other, then the graph representing the connections of the components and even the elements them-selves may be divided into subgraphs, in some of which every element can be reached from every other, but not in others. How then does one obtain the desired results? The graph of a decision system must always be connected. It cannot be divided into two or more disjoint parts.

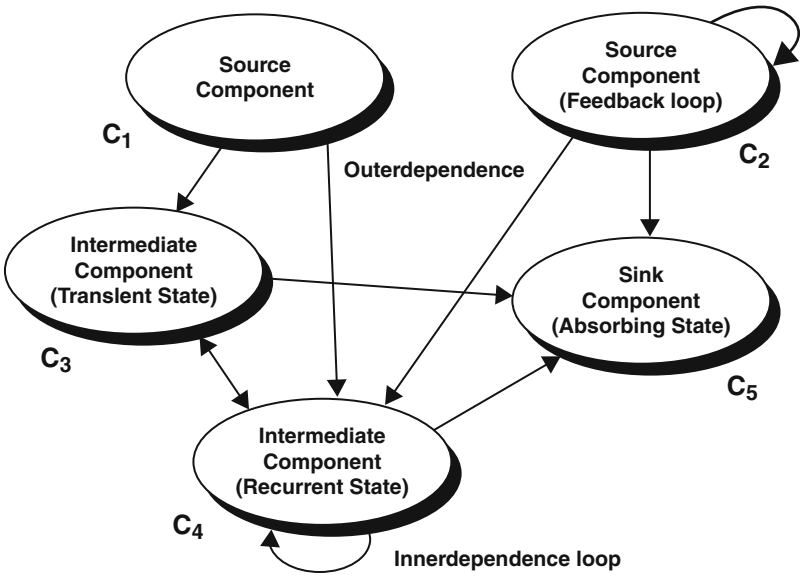
When the criteria do not depend on the alternatives, the latter may be kept out of the supermatrix and evaluated according to the ideal or distributive modes of the AHP, after the limiting priorities of the criteria are obtained from the limiting supermatrix. Otherwise, if some criterion depends on the alternatives or if there is inner dependence among the alternatives, they must be included in the supermatrix.

To test for the mutual independence of the criteria, one proceeds as follows: construct a zero-one matrix of criteria against criteria using the number one to signify dependence of one criterion on another, and zero otherwise. A criterion need not depend on itself, just as an industry, for example, need not use its own output. For each column (criterion) of this matrix, construct a pairwise comparison matrix only for the dependent criteria, derive an eigenvector, and augment it with zeros for the excluded criteria. If a column consists of only zeros, then as-sign a zero vector. The question in the comparison would be: For a given criterion, which of two criteria depends more on that criterion with respect to the goal or with respect to a higher-order control criterion?

The Supermatrix of a Feedback System

This section introduces different examples of graphs and their matrices. Assume that there is a system of N clusters or components whereby the elements in each component interact or have an impact on or are influenced by some or all of the elements of another component with respect to a property governing the interactions of the entire system, such as energy or capital or political influence. Assume that component h, denoted by C_h, $h = 1, \ldots, N$, has n_h elements, denoted by $e_{h1}, e_{h2}, \ldots, e_{hn\ k}$. The impact of a given set of elements in a component on another element in the system is one represented by a ratio scale priority vector derived from paired comparisons in the usual way.

In Fig. 1, no arrow feeds into a source component, no arrow leaves a sink component, and arrows both leave and feed into a transient component. Each such priority vector is now introduced in the appropriate



Analytic Network Process, Fig. 1 Feedback network

position as a column vector in a supermatrix of impacts displayed as follows:

$$
W = \begin{array}{c|cccc}
 & C_1 & C_2 & \cdots & C_N \\
 & e_{11}e_{12}\cdots e_1n_1 & e_{22}e_{22}\cdots e_{2n_2} & & e_{N1}e_{N2}\cdots e_Nn_N \\
\hline
 & e_{11} & & & \\
 & e_{12} & & & \\
C_1 & \vdots \quad W_{11} & W_{12} & \cdots & W_{1N} \\
 & e_{1n_1} & & & \\
 & e_{21} & & & \\
C_2 & e_{22} \quad W_{21} & W_{22} & \cdots & W_{2N} \\
 & \vdots & & & \\
 & e_{2n_2} & & & \\
\vdots & \vdots \quad \vdots & \vdots & \vdots & \vdots \\
 & e_{N1} & & & \\
 & e_{N2} & & & \\
C_N & \vdots \quad W_{N1} & W_{N2} & \cdots & W_{NN} \\
 & e_{Nn_N} & & &
\end{array}
$$

where the i, j block of this matrix is given by

$$
W_{ij} = \begin{bmatrix}
w_{i1}^{(j_1)} & w_{i1}^{(j_2)} & \cdots & w_{i1}^{(j_{n_j})} \\
w_{i2}^{(j1)} & w_{i2}^{(j2)} & \cdots & w_{i2}^{(j_{n_j})} \\
\vdots & \vdots & \vdots & \vdots \\
w_{in_i}^{(j_1)} & w_{in_i}^{(j_2)} & \cdots & w_{in_i}^{(j_{n_j})}
\end{bmatrix}
$$

each of whose columns is a principal eigenvector that represents the impact of all the elements in the i th component on each of the elements in the j th component.

The discussion of this section will focus on deriving limiting priorities for the supermatrix. It must first be reduced to a matrix, each of whose columns sums to unity. As already mentioned, such a matrix is known as a column stochastic or simply a stochastic matrix. If the matrix is stochastic, the limiting priorities depend on the reducibility, primitivity, and cyclicity of that matrix. Interaction in the supermatrix may be measured according to several different criteria. To display and relate the criteria, there is a need for a separate control hierarchy that includes these criteria with their priorities. For each criterion, a different supermatrix of impacts is developed, and in terms of that criterion the components are compared according to their relative impact (or absence of impact) on each other component at the top of the supermatrix, thus developing priorities to weight the block matrices of eigenvector columns under that component in the supermatrix. The resulting supermatrix would then be a stochastic matrix. Hereafter, W is usually a stochastic matrix.

In general, the supermatrix is rarely stochastic because, in each column, it consists of several eigenvectors that each sum to one, and, hence, the entire column of the matrix sums to an integer greater

than one. The natural thing to do is to determine the influence of the clusters on each cluster with respect to the control criterion. This yields an eigenvector of influence of all the clusters on each cluster. The priority of a component of such an eigenvector is used to weight all the elements in the block of the supermatrix that corresponds to the elements of both the influencing and the influenced cluster. The result is a stochastic supermatrix. This is not a forced way to make the matrix stochastic. It is natural. Why? Because the elements are compared among themselves and one needs information about the importance of the clusters to which they belong, to determine their relative overall weight among all the elements in the other clusters. Normalization would be meaningless, and such weighting does not call for normalization. Here is an example of why it is necessary to weight the priorities of the elements by those of their clusters: If one shouts into a room, "Ladies and Gentlemen, the President," everyone is alerted and somewhat awed to expect to see the president of the United States because he is in the news so often. But if the announcement is then followed by, "of the Garbage Collection Association," the priority immediately drops according to the importance of the group to which that president belongs. There is a need for such consideration.

The Control Hierarchy

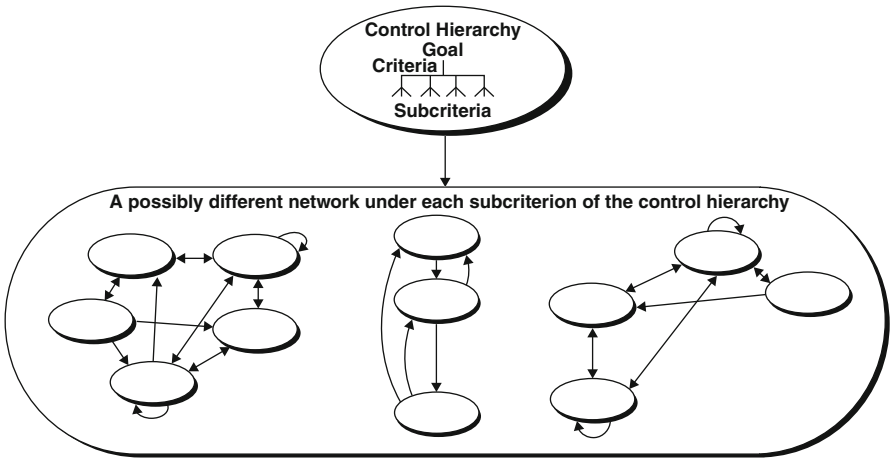
Analysis of priorities in a system can be thought of in terms of a control hierarchy with dependence among its bottom-level subsystem arranged as a network (Fig. 2). Dependence can occur within the clusters and between them. A control hierarchy at the top may be replaced by a control network with dependence among its clusters. More generally, one can have a cascading set of control networks, the outcome of one used to synthesize the outcomes of what it controls. For obvious reasons relating to the complexity of exposition, apart from a control hierarchy, we will not discuss such complex control structures here. A control hierarchy can also be involved in the network itself with feedback involved from the criteria to the elements of the network and back to the criteria to modify their influence. This kind of closed-circuit interaction between the operating parts and the criteria that drive the parts is likely to be prevalent in the brain.

A component or cluster in the AHP is a collection of elements whose function derives from the synergy of their interaction and, hence, has a higher-order function not found in any single element. A component is like the audio or visual component of a television set or like an arm or a leg, consisting of muscle and bone, in the human body. A mechanical cluster has no synergy value but is simply an aggregate of elements and is not what is meant by a component or cluster. The clusters of the system should generally be synergistically different from the elements themselves. Otherwise, they would be a mechanical collection with no intrinsic meaning.

The criteria in the control hierarchy that are used for comparing the components are usually the major parent criteria whose subcriteria are used to compare the elements in the component. Thus the criteria for comparison of the components need to be more general than those of the elements because of the greater functional complexity of the components. Sometimes for convenience, interactions of both components and elements are examined in terms of the same criteria in the control hierarchy. Although one does that to economize on the effort spent, it is more meaningful to compare the clusters with respect to control criteria and to compare the elements with respect to subcriteria of the control criteria. Otherwise the process can lead to asking difficult questions in making the paired comparisons.

The control hierarchy, critical for ANP analysis, provides overriding criteria for comparing each type of interaction that is intended by the network representation. There are two types of control criteria (subcriteria). A control criterion may be directly connected to the structure as the goal of a hierarchy if the structure is in fact a hierarchy. In this case the control criterion is called a comparison-linking criterion. Otherwise a control criterion does not connect directly to the structure but induces comparisons in a network. In that case the control criterion is called a comparison-inducing criterion.

An example of dependence between components is the input-output of materials among industries. The electric industry supplies electricity to other industries including itself. But it depends more on the coal industry than on its own electricity for operation and also more on the steel industry for its turbines.



Analytic Network Process, Fig. 2 A control hierarchy

To summarize, a control hierarchy is a hierarchy of criteria and subcriteria for which priorities are derived in the usual way with respect to the goal of the system being considered. The criteria are used to compare the components of a system, and the subcriteria are used to compare the elements. The generic question is: Given an element (in the same component or in another component) of the system or given a component of that system, how much more does a given element (component) of a pair influence that element (component) with respect to a control subcriterion (criterion)? The weights of the components are used to weight the blocks of the supermatrix corresponding to the component being influenced. The limiting priorities in each supermatrix are weighted by the priority of the corresponding subcriterion and the results are synthesized for all the subcriteria. If it should happen that an element or a component has no input, a zero is entered in the corresponding priority vector.

In each block of the supermatrix, a column is either a normalized eigenvector with possibly some zero entries, or all of its elements are equal to zero. In either case it is weighted by the priority of the corresponding cluster on the left. If it is zero, that column of the supermatrix must be normalized after weighting by the cluster's weights. This operation is equivalent to assigning a zero value to the cluster on the left when weighting a column of a block with zero entries and then renormalizing the weights of the remaining clusters.

Figures 3 and 4 and their accompanying supermatrices represent a hierarchy and a holarchy whose bottom level is connected to its top level of criteria and has no single element goal as in a hierarchy. Note the difference between the two.

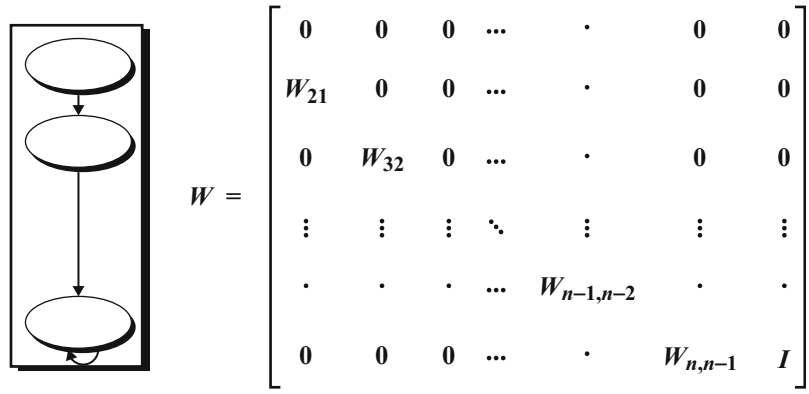
Note from Fig. 3 that the entry in the last row and column of the supermatrix of a hierarchy is the identity matrix I. The limiting results obtained by raising W to powers and by following the theoretical route described here turn out to be the same.

For Fig. 4 a system may be generated from a hierarchy by increasing its connections gradually so that pairs of components are connected as desired and some components have an inner dependence loop. What follows is an illustration of how feedback and the supermatrix work in a pharmaceutical marketing decision problem.

Drug-Marketing Decision

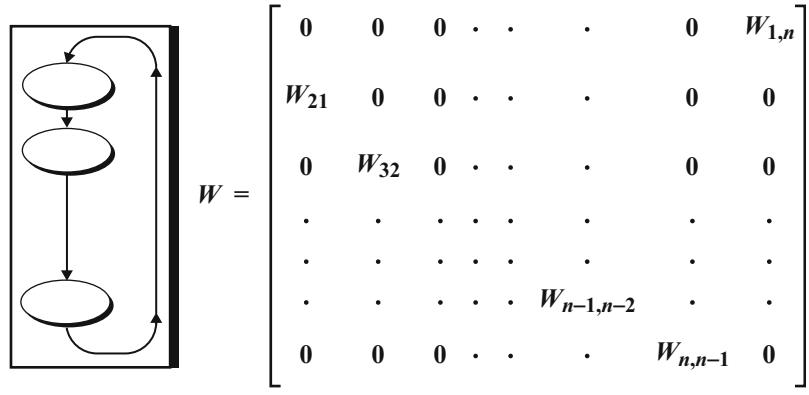
The problem faced by a large pharmaceutical company was how should the company market a new drug, given the pending patent expiration of an already existing drug it had been marketing? The ANP model used to make the decision illustrates two important ideas. First, it uses a very simple control hierarchy involving the aggregate criteria of benefits, costs, and risks (Fig. 5). For each of these controlling considerations, a separate network model of interactions was created. Thus there were three submodels, one for each control criterion. Second, the outcomes of the three sub-models were computed.

This ANP model was a real case in that the company was actually in the process of making the decision



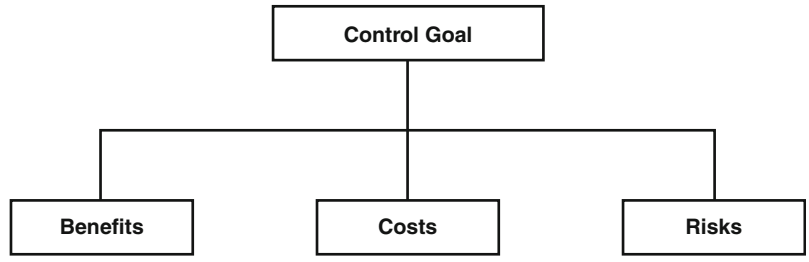
$$W = \begin{bmatrix} 0 & 0 & 0 & \cdots & \cdot & 0 & 0 \\ W_{21} & 0 & 0 & \cdots & \cdot & 0 & 0 \\ 0 & W_{32} & 0 & \cdots & \cdot & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \cdot & \cdot & \cdot & \cdots & W_{n-1,n-2} & \cdot & \cdot \\ 0 & 0 & 0 & \cdots & \cdot & W_{n,n-1} & I \end{bmatrix}$$

Analytic Network Process, Fig. 3 The structure and supermatrix of a hierarchy



$$W = \begin{bmatrix} 0 & 0 & 0 & \cdot & \cdot & \cdot & 0 & W_{1,n} \\ W_{21} & 0 & 0 & \cdot & \cdot & \cdot & 0 & 0 \\ 0 & W_{32} & 0 & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & W_{n-1,n-2} & \cdot & \cdot \\ 0 & 0 & 0 & \cdot & \cdot & \cdot & W_{n,n-1} & 0 \end{bmatrix}$$

Analytic Network Process, Fig. 4 The structure and supermatrix of a holarchy

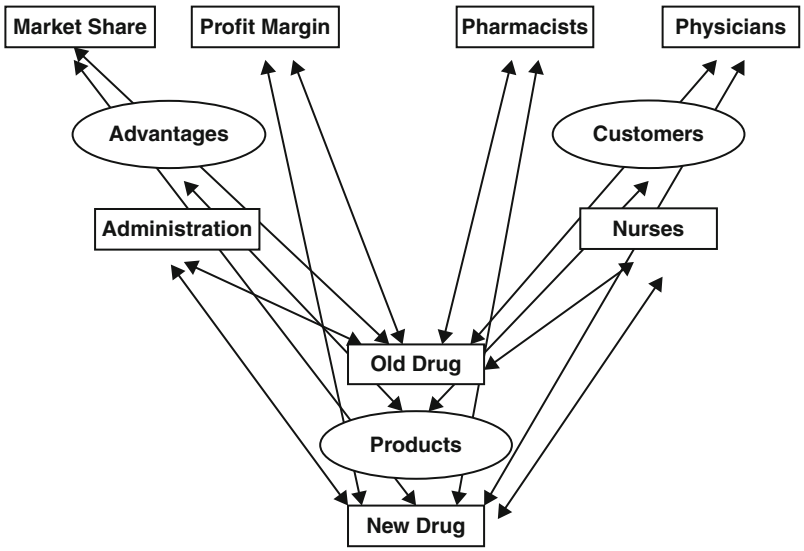


Analytic Network Process, Fig. 5 The control hierarchy

when the study was conducted. With the pending expiration of the current market offering, their objective was clearly to launch the new drug successfully before the patent for the old drug expired.

However, the company had to decide whether to market the new drug over the old. The questions were: Should the old drug be phased out slowly or entirely replaced by the new drug? How much of the marketing budget should be devoted to each strategy? Other important questions were also considered. Although the old drug would be facing stiffer competition, its brand recognition would continue to bring in revenue. The new drug was chemically improved over the old but needed greater promotion to be successful. There was also the risk that the new drug would be rejected by the HMOs and by customers. Moreover, with the added costs of promotion and the existing competition, the company could be facing slimmer profit margins, making these marketing decisions more critical.

The first step in building the model was to identify the stakeholders: HMO administrators, physicians, pharmacists, and nurses. Interviews with these stakeholders facilitated identifying the relevant clusters and the elements in each cluster and their



Analytic Network Process, Fig. 6 The benefits submodel

interdependencies. The interviews also yielded invaluable information and knowledgeable judgments for the model.

The second step was to partition the decision by using the control model approach. This allowed the participants to focus on each aspect of the larger problem. When considering benefits alone, for example, factors such as market share and profit margin were identified. Their importance may vary for the different stakeholders, and these are precisely the dependencies and priorities the model seeks and measures. For the costs side, the factors identified included marketing expenses, production costs, and market share loss. Factors for the risks model included product acceptance, production backlog, patient expiration, and sales incentives. Figure 6 illustrates the submodel corresponding to the benefits control criterion (note there that an oval represents a cluster and a rectangle represents an element belonging to a cluster).

The last step was to synthesize the ANP submodels. The best outcome derived from the ANP was the decision to proceed to market the new drug but also continue to market the old one. The new drug would incur higher marketing expenses compared with the old, but the brand recognition of the old drug would continue to provide a healthy return even with a lower marketing effort.

The actual decision of the company was to proceed to market the new drug and continue to market the old drug (Table 1). However, the company decided to eliminate the sales incentives and marketing expenses for the old drug and to increase the sales incentives and marketing expenses for the new drug. These decisions were consistent with the ANP model of Fig. 5.

Analytic Network Process, Table 1 The ratio of benefits/(costs × risks)

Market share	0.4845
Physicians	0.5901
Profit margin	0.3416
Administration	0.1739
Pharmacists	0.2905
Nurses	0.1193
Old drug	1.8180
New drug	2.6692

The final outcome is given in Table 1. It suggests marketing the new product in the ratio of 2.6692 to 1.8180, or 1.47 to 1, obtained by dividing the ratio scale numbers for the two drugs.

See

- ▶ Analytic Hierarchy Process
- ▶ Decision Analysis
- ▶ Multi-attribute Utility Theory
- ▶ Utility Theory

References

Saaty, T. L. (1977). A scaling method for priorities in hierarchical structures. *Journal of Mathematical Psychology, 15*(3), 234–281.

Saaty, T. L. (1982). *Decision making for leaders*. Belmont, CA: Wadsworth.

Saaty, T. L. (1986). Axiomatic foundation of the analytic hierarchy process. *Management Science, 32*, 841–855.

Saaty, T. L. (1994). *Fundamentals of the analytic hierarchy process*. Pittsburgh, PA: RWS Publications.

Saaty, T. L. (1996). *The analytic network process*. Pittsburgh, PA: RWS Publications.

Saaty, T. L. (2005). *Theory and applications of the analytic network process*. Pittsburgh, PA: RWS Publications.

Saaty, T. L. (2008). The analytic hierarchy and analytic network processes: Applications to decisions under risk. *European Journal of Pure and Applied Mathematics, 1*(1), 122–196.

Saaty, T. L. (2010). *Principia mathematica decernendi: Mathematical principles of decision making*. Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Alexander, J. (1989). *Conflict resolution: The analytic hierarchy process*. New York: Praeger.

Saaty, T. L., & Cillo, B. (2008). *The encyclicon* (Vol. 2). Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Ozdemir, M. (2005). *The encyclicon*. Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Peniwati, K. (2008). *Group decision making: Drawing out and reconciling differences*. Pittsburgh, PA: RWS Publications.

Saaty, T. L., & Sodenkamp, M. (2008). Making decisions in hierarchic and network systems. *International Journal of Applied Decision Sciences, 1*(1), 24–79.

Saaty, T. L., & Vargas, L. G. (1991). *Prediction, projection and forecasting*. Norwell, MA: Kluwer.

Saaty, T. L., & Vargas, L. G. (2006). *Decision making with the analytic network process: Economic, political, social and technological applications with benefits, opportunities, costs and risks*. New York: Springer.

Analytics

Term used to describe data-driven modeling and analysis for decision making; uses tools and methodologies from operations research.

Animation

► Visualization

ANP

► Analytic Network Process

Ant Colony Optimization

A population-based metaheuristic search approach predominantly for combinatorial optimization based on ideas from ant foraging (randomized search) and pheromone communication (information exchange) in forming favored paths.

See

► Metaheuristics
► Particle Swarm Optimization
► Swarm Intelligence

References

Dorigo, M., & Stützle, T. (2004). *Ant colony optimization*. Cambridge, MA: MIT Press.

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics—Part B, 26*(1), 29–41.

Anticycling Rules

Rules that prevent simplex-type algorithms from cycling.

See

► Bland's Anticycling Rules
► Cycling
► Degeneracy

Antithetic Random Variates

► Variance Reduction Techniques in Monte Carlo Methods

Applied Probability

The application of probability theory to the biological, physical, social, and engineering sciences.

See

► Stochastic Model

Approximate Dynamic Programming

Michael C. Fu
University of Maryland, College Park, MD, USA

Introduction

Stochastic dynamic programming models provide a rich framework for modeling sequential decision making under uncertainty. However, for many problems of practical interest, finding a tractable solution of the resulting models faces many potential challenges, including

- large state space;
- large action space;
- unknown state transition function/probabilities;
- costs or rewards that must be estimated.

The so-called curse of dimensionality refers to the computational intractability resulting from a large state space or a large action space (or both). Traditional methods for solving dynamic programming assume that the model is completely specified and that the corresponding expectations can be computed exactly. Even in cases where the state transition function/probabilities and probability distributions on costs/rewards are fully specified, analytical calculations may be computationally prohibitive if the state or action space is large.

Approximate dynamic programming (ADP) is the name given to approaches for providing practically implementable computational solutions to dynamic programming problems facing one or more of the challenges just described (Powell 2011; Si et al. 2004). In the artificial intelligence (AI) community, ADP methods are usually referred to as reinforcement learning (Sutton and Barto 1998), and in the stochastic control community, ADP is often called neuro-dynamic programming (Bertsekas and Tsitsiklis 1996). Simulation-based ADP approaches are described in Chang et al. (2007) and Gosavi (2003). ADP generally does not refer to the process of trying to estimate the transition probabilities directly. For expositional purposes, a discounted reward Markov decision process (MDP) model will be used throughout this entry. However, the state and action spaces will not be restricted to be finite or even countable, as the most general ADP approaches handle uncountable settings.

Problem Setting

The MDP is defined by the respective state and action spaces S and A, along with the one-stage rewards $r(s, a)$, $s \in S$, $a \in A$, which will be assumed to be stationary, and a probabilistic mechanism (which could be a simulation model) for transitioning from a state given an action taken. For the usual MDP with a finite or countable state space, this is specified in the form of state transition probabilities $\{p_{ij}(a), i, j \in S, a \in A\}$ that give the probability of going from state i to state j when action a is taken. Given an initial state (or more generally a probability distribution over the initial state), the optimization or control problem is to sequentially choose actions in each stage to maximize the expectation of the total discounted reward given by

$$\sum_{t=0}^{T} \beta^t r(s_t, a_t), \tag{1}$$

where $0 < \beta < 1$ is the discount factor, T is the number of stages (horizon length, so taking $T \to \infty$ corresponds to the infinite-horizon problem), a_t is the action taken in stage t, and s_t is the state reached in stage t. To simplify notation, it is assumed that all actions are feasible in all states; otherwise, one would write $A(s)$, $s \in S$ for the feasible action space in state s.

The action a_t taken in each stage is assumed to be specified by a Markov policy $\pi = \{\pi_0\ \pi_1 \ldots \pi_T\}$ that specifies for each possible state the action to be taken, i.e., a policy is a sequence of decision rules π_t: $S \to A$, where t is the stage. A stationary policy does not

depend on the stage, in which case π will be used to indicate both the policy and decision rule, which is the same for any stage. In the infinite-horizon setting, the usual objective is to find an optimal stationary policy (if one exists), for which there are three main approaches: value iteration, policy iteration, and linear programming.

Value Function and the Optimality Equation

Critical to ADP is the concept of a value function, which is defined for a given policy π from any stage n and state s_n to the end of the horizon:

$$V_n^{\pi}(s_n) = E\left[\sum_{t=n}^{T} \beta^{t-n} r(s_t, \pi_t(s_t)) \middle| s_n\right]. \quad (2)$$

The optimal value function achieves the maximum:

$$V_n^* = \sup_{\pi} V_n^{\pi},$$

and the problem of maximizing the expectation of (1) becomes that of finding $V_0^*(s_0)$ for a given starting state s_0, with an optimal policy given by

$$\arg\sup_{\pi} V_0^{\pi}(s_0).$$

According to the principle of optimality in dynamic programming, the optimal value function satisfies the Bellman optimality equation:

$$V_n^*(s) = \sup_{a \in A} \left\{E[r(s,a)] + \beta E[V_{n+1}^*(s')]\right\},\ s \in S, \quad (3)$$

where s' represents the state (a random variable) reached in stage $n + 1$ when taking action a from current state s in stage n, and the reward $r(s, a)$ is also allowed to be random. The form of the optimality equation given by (3) is convenient for the ADP setting, generalizing the usual form for the countable state space MDP with nonrandom rewards given by

$$V_n^*(s) = \sup_{a \in A} \left\{r(s,a) + \beta \sum_{j \in S} p_{sj}(a) V_{n+1}^*(j)\right\},\ s \in S.$$

Two Approaches

The computational challenges that may arise in (stochastic) dynamic programming are evident from inspection of the optimality equation in the form given by (3). First, the equation has to be solved for every state in the state space S over the action space A. Thus, for a given state, the optimization problem over a huge action space could be intractable, and/or the number of optimization problems that have to be solved could be impractical. Furthermore, the expectations in (3) may be analytically intractable and must be estimated from sampled simulation output or from actual (e.g., online) data. This may arise even in the finite state space nonrandom reward setting, e.g., if the transition probabilities are not readily available, so that state transitions can only be sampled using a simulation model or observing a real system.

This entry briefly describes the two most frequently employed ADP approaches:

- Value (or cost-to-go) function approximation;
- Q-learning.

The former addresses the state space curse of dimensionality, whereas the latter addresses the setting where the expectations in (3) must be estimated. Other approximations not described here include purely greedy (myopic) algorithms, one-step look-ahead algorithms, rollout algorithms, linear programming methods, policy parameterization, and numerous aggregation (state and/or action) methods.

For further expositional simplicity, the infinite-horizon case will be considered, so that the value function (for a given policy π) and optimal value function are defined by

$$V^{\pi}(s) = E\left[\sum_{t=0}^{\infty} \beta^t r(s_t, \pi(s_t)) \middle| s\right], \quad (4)$$

$$V^*(s) = \sup_{\pi} V^{\pi}(s), \quad (5)$$

where a stationary policy π is assumed. The optimal value function V^* in (5) then satisfies the following Bellman optimality equation:

$$V^*(s) = \sup_{a \in A} \left\{E[r(s,a)] + \beta E[V^*(s')]\right\},\ s \in S, \quad (6)$$

which is a fixed point equation over the state space S (again s' represents the random state reached when taking action a from current state s). Value function approximation uses a compact representation for the value function (4), which should satisfy (6) if optimal. Applying value function approximation thus involves two steps if used to find optimal policies: approximating the optimal value function and then finding an approximately optimal policy.

A common value function approximation is a linear combination of a set of basis functions:

$$\widehat{V}^{\pi}(s,\mathbf{r}) = \sum_i r_i \phi_i(s) = \mathbf{r}\boldsymbol{\phi},$$

where $\mathbf{r} = [r_1\ r_2\ \ldots]$ is a (row) vector (set) of parameters and $\boldsymbol{\phi} = [\phi_1\ \phi_2\ \ldots]^T$ is a (column) vector (set) of basis functions. The basis functions and parameterization are usually chosen based on some domain knowledge of the problem setting, which can be expressed in terms of features of the problem. However, in many settings this may be more art than science.

Instead of working with the value function directly, Q-learning works with the so-called Q-factors, which are the functions being optimized in the Bellman equation (6), i.e.,

$$Q(s,a) = E[r(s,a)] + \beta E[V^*(s')], \tag{7}$$

which depends on both the state and action. If all the Q-factors are known, then the value function can be found simply by taking

$$V^*(s) = \sup_{a \in A} Q(s,a),$$

so that (7) becomes

$$Q(s,a) = E[r(s,a)] + \beta E[\sup_{a' \in A} Q(s',a')]. \tag{8}$$

Note that in (8) the optimization operator (supremum) is inside the expectation, whereas it appears outside the expectation in (6). However, now the curse of dimensionality in both the state and action spaces comes directly into play, since a two-dimensional lookup table is required to keep track of the Q-factors that will be estimated using Q-learning. Similar to value function approximation, one could also seek a compact (approximate) parametric representation of the Q-factors to reduce the dimensionality.

The basic idea of Q-learning is to use outputs from the model (real or simulated) to iteratively update the estimate of the Q-factors as follows:

$$\begin{aligned} Q_{n+1}(s,a) &= Q_n(s,a) \\ &\quad + \alpha_n \left[(r(s,a) + \beta \sup_{a' \in A} Q_n(s',a')) - Q_n(s,a) \right] \\ &= (1-\alpha_n) Q_n(s,a) + \alpha_n \left[r(s,a) + \beta \sup_{a' \in A} Q_n(s',a') \right], \end{aligned}$$

where n denotes the iterate number and α_n is called the learning rate. Q-learning is a form of the AI machine learning approach called temporal-difference (TD) learning, because the Q-factor update in the first expression uses the difference between the predicted Q-factor value based on a sampled version of (8) for a single next state s' (simulated or from real online data) and the current Q-factor value. In the second form of the update iteration, the Q-factors are expressed as a convex combination of these two values. The desire is that with increasing n, the estimated Q-factors approach the true Q-factors, i.e., $Q_n(s, a) \to Q(s, a)$ for all states $s \in S$ and actions $a \in A$. Since the first form of the iterative updating also puts Q-learning in the class of stochastic approximation algorithms, for which there is a large body of research, convergence analysis can use machinery developed for those methods, and the choice of the learning rate parameter sequence $\{\alpha_n\}$, which greatly affects the practical performance of the algorithm, can also find guidance there.

Successful Applications

The earliest success stories for ADP are applications in computer programs for playing games, from checkers (Samuel 1959) to backgammon (Tesauro 1995). The program TD-Gammon, which reached the level of the best human backgammon players at the time, used an artificial neural network to approximate the value function, where the state space is estimated to have over 10^{20} states (close to the estimated number of grains of sand in the world's beaches). ADP ideas were also implemented in the computer system

"Watson" that was designed to play the television quiz show *Jeopardy!* and in 2011 beat the two best human players; Tesauro was a member of the IBM team that developed "Watson."

Other successful applications of ADP have been found in financial engineering, although the term ADP is not generally used or recognized there. For example, the pricing of financial derivatives for American-style options is an optimal stopping problem, where at each decision stage there are two possible actions: "Hold" or "Exercise" (and stop). Generally the exercise value can be calculated easily, so calculating the price of the derivative reduces to calculating the value of holding (taking the "Hold" action), which is also called the continuation value and can be represented by $Q(s, \text{Hold})$ using the Q-factor notation introduced previously, with

$$V^*(s) = \max\left\{Q(s, \text{Hold}), \text{Value of Exercising at } s\right\}.$$

Now either of the two ADP approaches described in the previous section could be used to estimate the continuation values $Q(s, \text{Hold})$: Q-learning or function approximation. Regression based on a large number of simulated paths is a popular way of carrying out the function approximation. See (Glasserman 2004, Chapter 8) for more details on this approach.

A real-world large-scale application of ADP received the 2010 Daniel H. Wagner Prize for Excellence in Operations Research Practice, which "emphasizes the quality and coherence of the analysis used in practice" (as stated on the INFORMS Web site, accessed October 2010). The application addressed logistics and transportation planning for a large trucking company, reported in Simão et al. (2010): "Schneider National needed a simulation model that would capture the dynamics of its fleet of over 6,000 long-haul drivers to determine where the company should hire new drivers, estimate the impact of changes in work rules, find the best way to manage Canadian drivers, and experiment with new ways to get drivers home. It needed a model that could perform as well as its experienced team of dispatchers and fleet managers. In developing our model, we had to simulate drivers and loads at a high level of detail, capturing both complex dynamics and multiple forms of uncertainty. We used approximate dynamic programming to produce realistic, high-quality decisions that capture the ability of dispatchers to anticipate the future impact of decisions. The resulting model closely calibrated against Schneider's historical performance, giving the company the confidence to base major policy decisions on studies performed using the model. These policy decisions helped Schneider to avoid costs of $30 million by identifying problems with a new driver-management policy, achieve annual savings of $5 million by identifying the best driver domiciles, reduce the number of late deliveries by more than 50 percent by analyzing service commitment policies, and save $3.8 million annually by reducing training expenses for new border-crossing regulations."

Concluding Remarks

Historically, ADP can be traced all the way back to the very roots of dynamic programming itself, from function approximations in Bellman and Dreyfus (1959) to the machine learning-based checkers-playing program of Samuel (1959). Q-learning was introduced by Watkins in his 1989 Ph.D. dissertation, with a convergence proof published in Watkins and Dayan (1992). An online and continually updated version of the research-oriented Chap. 6 on ADP in Volume II of the two-volume book by Bertsekas (2007) is made freely available for download by the author at his MIT Web site.

Although the setting considered here is stochastic, the idea of approximating the value function in dynamic programming can also be applied to completely deterministic settings, which might be appropriate when the state and/or action space is huge.

See

- ▶ Dynamic Programming
- ▶ Markov Decision Processes
- ▶ Stochastic Approximation

References

Bellman, R. E., & Dreyfus, S. E. (1959). Functional approximations and dynamic programming. *Mathematical Tables and Other Aids to Computation, 13*(68), 247–251.

Bertsekas, D. P. (2007). *Dynamic programming and optimal control* (3rd ed., Vol. 1&2). Belmont: Athena Scientific.
Bertsekas, D. P., & Tsitsiklis, J. N. (1996). *Neuro-dynamic programming*. Belmont: Athena Scientific.
Chang, H. S., Fu, M. C., Hu, J., & Marcus, S. I. (2007). *Simulation-based algorithms for Markov decision processes*. London: Springer.
Glasserman, P. (2004). *Monte Carlo methods in financial engineering*. New York: Springer.
Gosavi, A. (2003). *Simulation-based optimization: Parametric optimization techniques and reinforcement learning*. Boston, MA: Kluwer.
Powell, W. B. (2011). *Approximate dynamic programming: Solving the curses of dimensionality* (2nd ed.). New York: Wiley.
Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development, 3*, 211–229.
Si, J., Barto, A. G., Powell, W. B., & Wunsch, D. W. (Eds.). (2004). *Handbook of learning and approximate dynamic programming*. Piscataway, NJ: IEEE Press.
Simão, H. P., George, A., Powell, W. B., Gifford, T., Nienow, J., & Day, J. (2010). Approximate dynamic programming captures fleet operations for Schneider National. *Interfaces, 40*(5), 342–352.
Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge: MIT Press.
Tesauro, G. (1995). Temporal difference learning and TD-gammon. *Communications of the ACM, 38*(3), 58–68.
Watkins, C. J. C. H., & Dayan, P. (1992). Q-learning. *Machine Learning, 8*(3), 279–292.

Arc

An edge (link, path) connecting two nodes in a graph or network. The term arc usually means that it is directed.

See

- ▶ Digraph
- ▶ Graph Theory
- ▶ Network Optimization

Archimedean Axiom

The property of real numbers that for any positive numbers a and b, there is a positive integer n such that $a < nb$.

ARIMA

Autoregressive Integrated Moving Averages.

See

- ▶ Time Series Analysis

Army Operations Research

- ▶ Military Operations Research

Arrival Process

A random point process or marked point process with marks denoting some aspects of the stream of customers arriving to a queue or some aspects of the queue itself at the times of arrival, with points representing the precise instants of arrivals. For example, in the marked point process $(\boldsymbol{X}^a, \boldsymbol{T}^a)$, the $\boldsymbol{X}^a$ process may represent the sequence of customer priority classes arriving to a queue, while the $\boldsymbol{T}^a$ process would be the sequence of actual arrival times.

See

- ▶ Queueing Theory

Arrival-Point Distribution

- ▶ Customer Distribution
- ▶ Queueing Theory

Arrow Diagram

A graphic use of arrows to represent component jobs of a project and the manner in which they are

interconnected. An arrow diagram is sometimes also called a network diagram.

See

► Network Planning

Artificial Intelligence

Peter I. Cowling[1], Harvey J. Greenberg[2] and Kenneth De Jong[3]
[1]University of Bradford, Bradford, UK
[2]University of Colorado-Denver, Denver, CO, USA
[3]George Mason University, Fairfax, VA, USA

Introduction

Both artificial intelligence (AI) and operations research (OR) have roots in the early years of computer science, both matured during the 1950s and 1960s, and both have undergone major changes in the last few decades as a result of the explosive power and affordability of computers. Operations research is an interdisciplinary approach to problem solving, generally using mathematical models to represent a system. Artificial intelligence involves making computers perform functions that are generally believed to require intelligence. Although the meaning of intelligence is subject to debate, one ingredient, which relates to OR, is being able to solve complex problems.

Recent years have seen considerable overlap between OR and AI with a number of conferences and journals devoted specifically to areas of overlap such as heuristic search and data mining. Indeed some similar techniques have followed parallel paths, for example, A* search within AI and branch and bound within OR. Problem areas involving big data that arise particularly through the development of the Internet give rise to new opportunities and problems for both AI and OR, with data mining and machine learning techniques, which are generally regarded as AI approaches, becoming increasingly prominent within the OR community.

In developing computer systems capable of complex problem solving, the AI community has adopted a wide variety of approaches that range from very strong cognitive models of human problem solving to very strong computational models that have little internal resemblance to human problem solving. The more cognitively oriented models focus on knowledge representation and reasoning, while the more computationally oriented methods focus on efficient techniques for representing and searching large, complex spaces.

The AI techniques that are likely to be most useful to the OR researcher and practitioner are those of weak AI, which consists of those AI techniques which are targeted at solving specific problems, and form the primary focus of this article. Strong AI research aims to find a more general-purpose computer intelligence at a similar level to that of human or at least animal intelligence. This appears to be an extremely difficult goal and problems requiring this general-purpose intelligence are often referred to as AI-hard. To illustrate this, consider the Deep Blue chess computer, which built upon decades of research on computer hardware, software and chess AI to beat the human world champion, Garry Kasparov, in 1997. While Deep Blue was arguably the strongest chess playing entity on the planet at that time, Deep Blue was incapable of doing almost all of the everyday activities of a human, such as talking about the weather or filling in a tax return. One of the ongoing problems of AI research is the so-called AI effect that once a precise algorithm is known for performing a task that appeared to require intelligence, then it can be argued that it does not require intelligence. Some AI methods, however, such as neural networks, can be encoded on a computer but the precise mechanism by which solutions are generated is not readily understood.

In the past there has been overoptimistic extrapolation of the triumphs of weak AI research to the possibility of general intelligence. Recent success in AI approaches to steering a road or air vehicle in real time point to a substantial increase in the generality of problems which AI can solve, but still leave the goals of strong AI some way off. One exciting possibility for strong AI is the computational neuroscience approach to directly modeling the processing methods of the human brain (Feng 2003). While this approach is still in its infancy, it provides a method which might lead to very flexible and powerful problem-solving approaches in the future.

AI as Search

It would seem at first glance that the obvious way to build intelligent systems is to use as a starting point

properties of human problem solving. Articulating and implementing human capabilities, however, has proven to be an exceedingly difficult task. This is perhaps best illustrated by the efforts to build game playing programs for chess (Hsu 2004) and Go (Lucas 2011), where approaches that aim to emulate the thought processes of a human expert are much weaker than those that are able to search through millions of alternative move sequences before selecting a move.

Unlike OR, AI has the dual mission of producing a solution and providing a laboratory tool to test explanatory theories of intelligent behavior. This dual role creates a difference in how AI approaches search. An OR design always seeks an optimal solution with a minimum of computational effort and, for complex problems, must trade-off solution quality with computational effort. An AI design is additionally concerned with the meaning of the heuristic in relation to human reasoning, and the trade-off between solution quality and computational effort can be different. For example, humans generally construct satisfactory solutions to problems such as "what route will I take home tonight," and do not spend much time thinking about optimality. Historically, OR has focused on computational efficiency by exploiting mathematical structures in a relatively narrow class of problems, while AI has focused on heuristics for broader class of problems. In recent years, however, OR and AI approaches have become more intermingled.

Modern AI approaches (Russell and Norvig 2010) generally objectify a computational problem-solving approach as an agent. Often the computational decision making approach is closely integrated into a physical system, or robot. The agent receives sensory information about the current state, and chooses between the set of possible actions at that state to move to a new state, and to continue selecting actions until a goal state is reached. Such a problem is known as a planning problem in the AI literature. A planning problem may be represented as a directed graph with states as nodes, actions as arcs and goal states as optimal solutions (or at least solutions of acceptable quality). This mapping allows search approaches from AI and OR to be used interchangeably for a wide range of problems.

The difficulty with modeling complex problems as spaces to be searched is that most such spaces are known to present NP-hard search problems, that is, there are no known polynomial-time algorithms for finding the answer. As a consequence considerable effort has gone into the development of efficient heuristic search procedures capable of finding solutions in acceptable amounts of search time. A heuristic is simply a rule of thumb that expresses some problem-specific knowledge that can be used to improve search efficiency.

A heuristic search strategy consists of a problem representation, a database of points in the search space, a set of heuristic rules and a control strategy. At any point during the search, the database contains subsets of candidate solutions. Using this database the control strategy selects a heuristic that generates a new candidate to be tested. Examples of control strategies are depth-first and breadth-first search. Testing a candidate, such as a partial solution to a traveling salesman problem, can be as simple as evaluating the length of the sub tour. A heuristic can be as simple as choosing the next link to be a nearest neighbor of an endpoint of the partial solution, or it can be more computationally intensive, such as solving a linear-programming relaxation problem that provides a bound on the completion of the partial solution.

An important AI paradigm for node selection in a search tree is the A* algorithm. This is a family of algorithms designed to find an optimal or high-quality solution guided by a heuristic function. For each node, n, in the search tree, such as at a partial solution to a traveling salesman problem, the cost to reach that node is denoted by $g(n)$, and the estimated minimum cost to complete the solution is the heuristic function, denoted $h(n)$. The control strategy selects the node having the minimum value of $g + h$.

In this family of algorithms, if the heuristic function is admissible: i.e., $h(n) \leq h^*(n)$, where $h^*(n)$ is the actual minimum cost to complete the solution from node n, then the resulting solution is guaranteed to be optimal. In words, if the heuristic function $h(n)$ is optimistic, then the resulting A* search finds an optimal solution. If h is admissible, the search tree can be pruned below node n when $g(n) + h(n) \geq g(n')$ for some complete solution n' already found (in OR, this is called fathoming a node).

A special case is breadth-first search, where $h \equiv 0$ and $g(n)$ is defined to be the depth of node n. Another special case is OR's branch-and-bound for general integer

programming, where $g + h$ is the objective value of the linear relaxation at a node.

When using a non-admissible heuristic estimate h, optimality is no longer guaranteed, although the time taken to find a solution of sufficient quality may be very greatly reduced. For example, this approach may be taken in applications such as finding paths for a very large number of units in a computer simulation or game, and is used in a variety of search local approaches which do not have optimality guarantees.

Models for AI problems are often less well-defined than in OR, in areas such as machine vision or vehicle guidance, and it is difficult to define a priori a heuristic search approach for such a problem. As a consequence, an important subarea of AI focuses on algorithms that construct models from sampled data provided a priori or dynamically acquired during the problem solving process. These techniques are generally labeled as machine learning algorithms (Mitchell 1997; Alpaydin 2004) and have been successfully applied to a variety of difficult problems including classification (e.g., support vector machines for image processing) and sequential decision problems (e.g., reinforcement learning techniques for robot control).

Machine learning approaches have proven their worth in conjunction with OR heuristics and algorithms, for example in finding good parameters for another search approach or in choosing which heuristic approach to use at each step of search. The use of machine learning approaches to choose between other heuristics was christened a hyperheuristic in (Cowling et al. 2001), and this approach has gained in popularity as an approach for solving difficult optimization problems (Chakhlevitch and Cowling 2008). Hyperheuristics are examples of a more general approach to machine learning where an ensemble of learning methods are combined using some kind of voting method to yield an approach which is greater than the sum of its parts (Polikar 2006). Recent years have seen considerable success for ensemble approaches, particularly for classification problems arising in data mining (Nisbet et al. 2009).

Another theme in AI is the design and application of biologically-inspired techniques. A well-developed subarea focuses on artificial neural networks (ANNs) (Bishop 1995). ANNs are very loosely based on a connectionist model of the mammalian brain and have proven extremely useful when dealing with problems where it is necessary to provide a mechanism for approximating a complex function. An ANN is a weighted digraph with identified sets of input, hidden, and output nodes. Values are applied to the input nodes which are then propagated though the ANN to generate values at the output nodes which approximate the given function. The values at each hidden node and the output nodes are obtained by applying a nonlinear squashing function to the sum of the inputs at each node, weighted using the arc weights. These values are then propagated forward to subsequent hidden nodes and ultimately to output nodes. The function to be approximated may be highly complex, for example to estimate the probability that a luggage contains an explosive device by using an ANN to analyze the pixels of an X-ray photograph of the luggage. Appropriate weights are determined by supervised or unsupervised learning. In supervised learning a large number of examples where the function value is known are considered and the variance between ANN outputs and true outputs is minimized by treating the problem of changing weights as a nonlinear optimization problem. Back propagation is often used to improve the weights, proceeding backwards from output nodes to input nodes, iteratively adjusting weights so as to reduce the error arising for each example. Unsupervised learning is also used, where the arc weights are often adjusted using an Evolutionary Algorithm (EA) and the ANN is evaluated by considering its performance directly in an application (e.g., the time taken to drive around a track for an ANN which controls vehicle steering).

Evolutionary Algorithms (EAs) (Michalewicz and Fogel 2004; De Jong 2006) are another well-developed area of biologically-inspired techniques. EAs use analogies with natural evolutionary processes. They maintain a population of candidate solutions, and local search operators which modify single solutions (mutation) and pairs of solutions (crossover). EAs attempt to find a solution with a high objective function value (generally known as fitness within an EA) by using selection processes modeled on Darwin's "survival of the fittest" principal. EAs have proven effective in a number of problems where it is difficult or impossible to use a large amount of problem knowledge, so that they may be regarded as relatively general-purpose approaches.

In game theoretic problems involving more than one participant, machine learning and evolutionary algorithm approaches have proven effective. In particular, Monte Carlo Tree Search (MCTS) (Lucas 2011) has recently provided much stronger players for the game of Go, a game that has for a long time been considered as one of the more intractable problems of AI research. Since 2009, MCTS approaches using highly parallel algorithms on many computer cores have started to compete at, or near human world champion level for small-board Go. MCTS algorithms selectively build a search tree, starting from the current state, by trading off exploration of unseen areas of the tree, and exploitation of known good areas, by considering the decision as to where to progress at each node/state as a multi-armed bandit problem. It seems likely that MCTS approaches will become increasingly important in a wide variety of decision applications.

There are many other approaches to heuristic search which have been explored by researchers and practitioners in OR and AI, known as metaheuristics (Michalewicz and Fogel 2004; Gendreau and Potvin 2010). Generally these approaches guide a local search operator to avoid poor local optima. Metaheuristics are often inspired by natural metaphors. They include: simulated annealing, inspired by the second law of thermodynamics; tabu search, inspired by connections between intelligence and memory; and swarm intelligence algorithms, inspired by the foraging behavior of insects.

AI as Logic

At the other end of the AI spectrum are techniques for solving problems by capturing the knowledge of human experts and using a reasoning approach analogous to that of a human problem-solver. Generally, this involves expressing a problem as a theorem to be proved using a particular system of logic and providing a computational procedure for proving such theorems. This is generally referred to as logic programming.

The most elementary form uses propositional logic. A set of propositions and logical expressions is given. The satisfiability problem is to find an assignment of truth values for the propositions such that the logical expressions are true. In logical inference, the given expressions are facts about how propositions relate to each other. Consider, for example, a project selection problem, where proposition $P_j = TRUE$ means project j is selected, and $P_j = FALSE$ means project j is not selected. Projects might be related by a simple precedence constraint, $P_i \rightarrow P_j$, which says that if project i is selected, project j must also be selected (or the selection of project i must precede the selection of project j).

A feasible truth assignment exists when the facts, which comprise one form of a knowledge base, are consistent. Redundant facts occur when some logical expression is implied by the others. For example, if the knowledge base contains $P_i \rightarrow P_j$ and $P_j \rightarrow P_k$, then $P_i \rightarrow P_k$ is a redundant fact (from transitivity of implication). The knowledge base contains circular reasoning if it contains the implication $P_i \rightarrow P_i$ through a chain of implications. For example, the expressions, $P_i \rightarrow P_j$, $P_j \rightarrow P_k$ and $P_k \rightarrow P_i$, are circular. This means that projects i, j and k form an equivalence class: all are selected, or all are rejected. In managing a knowledge base, one wishes to know if it is consistent, non-redundant, and non-circular, particularly when new facts are entered into it. In some cases, violations can mean an error in the rule entry. If, for example, precedence constraints are supposed to form a partial ordering, as in job scheduling, a circular chain implies that there is no feasible schedule. Similarly, a redundancy can be due to a subtle implication that the users of the rule base should know to avoid a false perception of what an inference means.

The satisfiability problem can be represented by a system of linear inequalities with binary-valued variables. Let x_j be 1 or 0, according to whether P_j is true or false, respectively. A simple implication, $P_i \rightarrow P_j$, is true if, and only if, $x_j \geq x_i$. More complex logical expressions can also be represented by linear inequalities, but the effort to derive the inequalities is, itself, a difficult problem unless special forms are assumed.

The difficulty with propositional logic is that it not sufficient to express many important pieces of knowledge such as "all men are mortal." To do so requires logics with more expressive power such as the first-order predicate calculus. Unfortunately, this increase in expressibility comes at a high price: the inability to implement efficient computational theorem

provers. One of the important developments in this area was the development of PROLOG, a logic programming system based on a restricted form of the predicate calculus that maintained considerable expressive power while providing an efficient theorem prover (inference engine). This was achieved by restricting the semantics of negation and restricting propositions to one special form, a Horn clause. This is where the antecedent is a conjunction of propositions and the consequent is only one proposition: $P_1 \wedge P_2 \wedge \ldots \wedge P_n \wedge P_0$. This can be represented by $x_0 \geq x_1 + x_2 + \ldots + x_n - n + 1$.

To see if a particular proposition can be inferred from a set of facts, the logic programming problem becomes one of combinatorial optimization. Suppose the truth value of P_i for i in I is given, where I is some index reference set over the propositions, in order to determine if P_j can be inferred from the knowledge base (where j is not in I). x_i is set to 1 or 0, according to whether P_i is true or false, respectively, for $i \in I$ and minimum and maximum values of x_j are determined subject to the logical constraints. The maximum of x_j is 1 if, and only if, P_j can be true, and the minimum of x_j is positive if, and only if, P_j must be true. Then, the facts imply P_j if P_j can be true (i.e., max $x_j = 1$) and P_j cannot be false (i.e., min $x_j > 0$). The facts are inconsistent if P_j cannot be true (i.e., max $x_j < 1$) and P_j cannot be false.

Logic programming has been successfully used to build many AI systems, particularly expert systems designed to give advice about a particular problem, in areas such as medical diagnosis. A substantial effort has been made to encode common sense in the database of a very large, multi-contextual knowledge base and inference engine developed by Cycorp using the rules of logical inference to provide a general way to make everyday decisions. Evidence so far suggests that such an expert system is effective for specific tasks, but not likely to work as a method for attacking the strong AI problems of general intelligence.

Knowledge representation in an expert system can include forms other than logical expressions and uncertainties can be represented by a variety of calculi. Alternative logics arise naturally in both human reasoning and in AI. For example, it may be accepted that birds fly and deal with special cases, like penguins, without disturbing the main value of logical reasoning (key terms to investigate are non-monotonic and default logics).

One area that bring together search and logic-based approaches to problem solving is constraint programming (Van Hentenryck and Michel 2009). Constraint programming software allows a user to express a rich variety of constraints, which link decision variables. These include constraints such as "all these variables take distinct values from a given set" which are often difficult to model using other techniques. Powerful solution approaches are then often able to find solutions which satisfy all constraints (and which may also be required to maximize an objective function). Constraint programming techniques have been used with considerable success in a variety of planning and scheduling applications. Their drawback is that they often cannot guarantee to find a solution at all, so that metaheuristics are often used instead of, or in conjunction with, constraint programming approaches for very difficult problems.

Further Reading

A broad selection of AI textbooks are available. One widely used AI text is Russell and Norvig (2010). Michalewicz and Fogel (2004) introduces evolutionary algorithms and metaheuristic methods alongside OR techniques in a problem-solving style, based on the classic Polya book. Traditional heuristic search is well covered by Pearl (1984). A number of journals are specifically devoted to the links between Computer Science and OR, where AI is the aspect of Computer Science most strongly represented, including the *INFORMS Journal on Computing* and *Computers and Operations Research*. The Conference on Integration of AI and OR (CPAIOR) brings together AI and OR researchers with a common interest in constraint programming. More generally a great number of AI and OR journals and conferences specifically ask for contributions which bridge the gaps between these disciplines.

See

- ▶ A* Algorithm
- ▶ Agent
- ▶ Computational Complexity
- ▶ Computational Intelligence

- Computer Science and Operations Research Interfaces
- Constraint Programming
- Data Mining
- Evolutionary Algorithms
- Expert Systems
- Game Theory
- Genetic Algorithms
- Graph Theory
- Heuristics
- Horn Clause
- Inference Engine
- Integer and Combinatorial Optimization
- Logic Programming
- Machine Learning
- Metaheuristics
- Monte Carlo Methods
- Monte Carlo Simulation
- Neural Networks
- Simulated Annealing
- Swarm Intelligence
- Tabu Search

References

Alpaydin, E. (2004). *Introduction to machine learning*. Cambridge: MIT Press.

Bishop, C. (1995). *Neural networks for pattern recognition*. Oxford: Oxford University Press.

Chakhlevitch, K., & Cowling, P. I. (2008). Hyperheuristics: Recent developments. *Springer SCI, 136*, 3–29.

Cowling, P. I., Kendall, G., & Soubeiga, E. (2001). A hyperheuristic approach to scheduling a sales summit. *Springer LNCS, 2079*, 176–190.

De Jong, K. (2006). *Evolutionary computation: A unified approach*. Cambridge: MIT Press.

Feng, J. (2003). *Computational neuroscience: A comprehensive approach*. Boca Raton: Chapman and Hall.

Gendreau, M., & Potvin, J.-Y. (2010). *Handbook of metaheuristics* (2nd ed.). New York: Springer.

Hsu, F.-H. (2004). *Behind deep blue: Building the computer that defeated the world chess champion*. Princeton: Princeton University Press.

Lucas, S., (2011). *IEEE transactions on computational intelligence and AI in games*. Special issue on Monte Carlo Tree Search and Computer Go.

Michalewicz, Z., & Fogel, D. B. (2004). *How to solve it: Modern heuristics* (2nd ed.). Berlin: Springer.

Mitchell, T. (1997). *Machine learning*. New York: McGraw-Hill.

Nisbet, R., Elder, J., & Miner, G. (2009). *Handbook of statistical analysis and data mining applications*. New York: Academic.

Pearl, J. (1984). *Heuristics*. Reading, MA: Addison-Wesley.

Polikar, R. (2006). Ensemble-based systems in decision making. *IEEE Circuits and Systems Magazine*, *6*(3), 21–45.

Russell, S., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). New York: Prentice Hall.

Van Hentenryck, P., & Michel, L. (2009). *Constraint-based local search*. New York: MIT Press.

Artificial Variables

A set of nonnegative variables added temporarily to a linear program to obtain an initial basic (artificial) feasible solution. If the original constraints are $\boldsymbol{Ax} = \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}$, than adding an artificial variable y_i to each equation yields the system $\boldsymbol{Ax} + \boldsymbol{Iy} = \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}, \mathbf{y} \geq \boldsymbol{0}$, where $\boldsymbol{y}$ is a column vector of artificial variables. Assuming the vector $\boldsymbol{b} \geq \boldsymbol{0}$, this system has an obvious basic (artificial) feasible solution, with $y_i = b_i$ being the basic variables and the x_i the nonbasic variables. To obtain a basic solution to the original constraints, the artificial variables must be driven to zero. One way to do this is to solve an auxiliary linear program (known as Phase I) where the objective is to minimize the sum of the artificial variables. If the new system has no solution with all artificial variables equal to zero, then the original constraints are infeasible.

See

- Big M Method
- Phase I Procedure
- Phase II Procedure
- Simplex Method (Algorithm)

Assignment Problem

The problem of optimally assigning m individuals to m jobs, so that each individual is assigned to one job, and each job is filled by one individual. The problem can be formulated as a linear-programming problem

with the objective function measuring the (linear) utility of the assignment as follows:

$$\text{Maximize} \sum_i \sum_j c_{ij} x_{ij}$$

subject to

$$\sum_i x_{ij} = 1 \qquad j = 1, \ldots, m$$

$$\sum_j x_{ij} = 1 \qquad i = 1, \ldots, m$$

$x_{ij} = 1$ if person i is assigned to job j

$x_{ij} = 0$ if person i is not assigned to job j

$c_{ij} =$ utility of person i assigned to job j

The problem is a special form of the transportation problem and, as such, has an optimal solution in which each variable is either zero or one. The problem can be solved by the simplex method, but special assignment problem algorithms tend to be computationally more efficient.

See

- ▶ Hungarian Method
- ▶ Transportation Problem
- ▶ Transportation Simplex (Primal-Dual) Method

References

Burkard, R., Dell'Amico, M., & Marterllo, S. (2009). *Assignment problems*. Philadelphia: SIAM.

Kuhn, H. W. (1995). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly, 2*, 83–97.

Auction and Bidding Models

Ronald M. Harstad
University of Missouri-Columbia, Columbia, MO, USA

Introduction

Models to assess and guide bidding and auction design have proliferated, broadened, and become more game-theoretic. Usage of auctions for major sales by governments, most notably of rights to the electromagnetic spectrum for digital transmission, have been a major impetus for these developments; Internet auctions and combinatorial procurement auctions have also stimulated and supported modeling advances. In 1997, William Vickrey shared the Nobel Prize in Economic Science for his pioneering development of auction theory; the Prize in 2007 was awarded to Leo Hurwicz, Eric Maskin, and Roger Myerson for closely related work in Mechanism Design.

Why hold an auction? The literature has focused on two reasons: because a bid-taker lacks the information needed to fix an apt price, and because a method is needed to harness competition. Research has also pointed to a need to organize the allocation of interrelated, heterogeneous assets or contracts so as to accommodate bidder synergies across packages of assets or contracts (Harstad and Pekec 2008). Auctions are also employed to reduce the flexibility of an agent assigned to attain a sale or procurement, or when legitimacy of a transaction and its pricing needs to be secured or to be made transparent (Rothkopf and Harstad 1994). All these reasons, and indeed, virtually all of the analyses of auctions and bidding, straightforwardly apply and adapt whether bidders are seeking to buy or to sell; for more concrete exposition, the bid-taker is here treated as a seller of an asset or assets (with but few and narrow exceptions, the literature models a seller and ignores any noncongruence of interests between an auctioneer and a consignor).

Given these reasons for auctioning, auctions and bidding have become the most extensive and most noted application of the theory of games of asymmetric information: it is because potential buyers have information about asset values that a less informed seller cannot simply fix a price and thus conducts an auction; it is because a winning bidder both uses and protects the bidder's private information against rivals who have differing information that winning an auction is on average profitable when bidding rationally, despite best efforts of rivals to compete and best efforts of a seller to extract some of this profit in the form of higher revenue.

In the face of myriad practical problems, however, models of bidding and auctions far too frequently make assumptions in service of elegant,

general results at far too high a cost in relevance; key limits to applicability are noted in the following.

Environments Modeled

Vickrey (1961) introduced auction theory under the assumptions that each bidder privately knew the auctioned asset's value, and that rivals' values were independent of each other. While appropriate for such a pioneering study, it is hard to justify the frequent tendency to continue to base theoretical analysis on this independent private-values model. Any durable asset presumably has a value to a bidder related to how useful it would be to rival bidders now or to potential purchasers in the future, and thus is not private. Even for a perishable commodity, a fancy dessert at a charity auction, a bidder rationally expects more serious competitive bids for a more mouth-watering cake—defeating the independence assumption. Independent-private-values models exhibit the simple characterization that a bidder facing more rivals bids more aggressively, and a far-from-robust revenue equivalence result that any auction in which the efficient acquirer always wins and any bidder with a zero chance of winning is indifferent over participating attains expected revenue equal to the expected opportunity cost (the second-highest value).

More useful is the polar opposite: a common-value model (Rothkopf 1969; Wilson 1977). In it, asset value is the same for all bidders, unknown to any at the time of the sale, and each bidder has privately acquired an estimate of the common value. A generalization to affiliated-values auctions allows asset values to incorporate a common resale value, as well as a private possession value. Where the common-value component is significant, facing more rivals raises the important "winner's curse" (Capen et al. 1971): if a bidder did not anticipate before bidding that winning likely implies rivals observed lower estimates, then the appropriate reassessment of asset value may well cause the winning bidder to feel miserable for winning. In particular, this leads to the conclusion that each extra rival past the first should lead the bidder to bid less aggressively for any given asset value estimate. (Even with participating in hundreds of auctions, sharp experimental subjects in economics laboratories cannot systematically learn to survive the winner's curse, the toughest problem economics experimenters have posed, see Kagel, Levin, and Harstad (1995).

Auction Forms

Many auction models consider any method of reaching an allocation—via communications from bidders to a seller who has an a priori commitment to functions specifying bidders' payments and probabilities of winning that depend on the profile of communications sent—to be an auction. In practice, auctions follow simpler rules and can be considered minor variants on a few basic forms.

Standard sealed bidding, commonly called a first-price auction, features a single round of simultaneous bids, with the highest bid winning and setting the price. It continues to be somewhat common, used for many procurement auctions, government sales of mineral rights, and in circumstances where congregating bidders in time and space is unjustifiably expensive. The bidder's problem is a complicated one, exhibiting a tradeoff between the probability of winning (enhanced by bidding more aggressively) and the expected profit in the event of winning (diminished by aggressiveness). Published solutions presume the set of assumptions {A}: a single, isolated auction; symmetric aggressiveness of a known, fixed number of rational rivals; all bidders risk-neutral; and that each bidder's information about asset value can be completely summarized by a single real number.

Vickrey auctions, that is, sealed bidding under second-price rules, in which the seller is committed to selling to the highest bidder at a price set by the highest losing bid (no matter how much higher the winner bid), are rare, but are an important benchmark to understanding auctions and bidders' incentives. Since a bid may only determine whether the bidder wins, but not how much the payment will be if the bidder wins, the feature that a bidder should submit the bid that separates prices at which the bidder would prefer to win from prices at which the bidder would rather lose is quite general. Outside private-values models, this pivotal bid depends on the information about asset value that can be inferred from having to pay a particular price (because the highest-bidding rival bid that price).

Most auctions feature dynamic pricing, starting at a level with more bidders than assets and becoming less attractive, with bidders repeatedly deciding whether to continue competing or cease (exit). The last remaining bidder wins, paying a price equal to, or some modest fixed increment above the last price where the bidder faced competition. Such auctions allow bidders to reevaluate as they observe diminishing competition; upon such reevaluation, the last two bidders effectively engage in a second-price auction. The literature commonly calls these English auctions; the game-theoretic model of English auctions usually considered the standard, assumes a continuous rise in prices with all exits public and irrevocable (Milgrom and Weber 1982). Under irrevocable, public exits, and assumptions {A}, expected revenue in the Bayesian Nash equilibrium of the English auction exceeds that of other auction forms. Relative to second-price auctions, this stems from the greater problem the winner has in protecting the winner's information, as revelation of exit prices, and thus inferences about losing bidders' information, makes the last rival's evaluation a better substitute for the winner's.

Auctions with dynamic pricing seldom cleanly provide the final bidders this much information. In part, this is because bidders are reluctant to let winning bids in auctions reveal their private information, let alone losing bids. In part, it is because an auctioneer does not need to learn how many bidders are still competing to raise the price; excess demand is sufficient. So it is common for an auctioneer to recognize two willing bidders, raising the price as they alternately affirm continuing excess demand. When one bidder ceases doing so, the auctioneer must find another bidder willing at that price to pair up, reestablishing the excess demand that raises the price. This aspect may slow the auction enough to reveal an exit price, but Harstad and Rothkopf (2000) find that a second-price auction model may more closely estimate revenue when some bidders' exits are silent and unobserved.

The Dutch auction starts with a high price unacceptable to any bidder and adjusts it down until a bidder ends the auction by accepting the current price (the term Dutch auction tends to be used in financial markets in a very different way). As this faces a bidder with the same probability of winning-expected profit if winning tradeoff, auction theorists who have never witnessed a Dutch auction assume it can be analyzed via the same model as the first-price auction. (As witnessed by the author, the auction's video-game speed explains its usage in flower and fish markets in the Netherlands; for flowers, each of 13 auctioneers sell a lot of flowers every 4.25 seconds).

Revenue Maximization

If n bidders will compete no matter what auction is used, a seller maximizes expected revenue via monopolistic inefficiencies that take the form of refusing to sell in some circumstances where a bidder offers more than keeping the asset is worth to seller. The reserve price that mainstream models focus on is a binding commitment never to part with the asset if no bid exceeds the reserve. Such a blunt instrument is used only in auctions of perishables; to auctioneers, a reserve price is the lowest price at which the asset will be sold today, but with later negotiations or auctions with a lower reserve possible.

A more realistic model treats the number of competitors as an endogenous variable, responding to the expected profitability of bidding. For this expected profitability to be predictable, potential bidders' private information, and their bidding aggressiveness, must be symmetric. In such a model, forcing inefficiencies is costly for a seller; the main conclusion is that aspects of different auction situations that are not modeled—e.g., costliness of congregating bidders, or bidders' preference for dynamic pricing—can be accommodated without necessarily sacrificing revenue.

Concluding Remarks

Sets of assumptions such as {A} seriously limit practical relevance by treating bidding for an asset as an isolated occurrence; this is more the exception than the rule. Firms in an industry may repeatedly bid to sell contracts or acquire key inputs, with repetition adding issues of reputation or collusion, not yet well treated by modelers. Also, the outcome of an auction may affect later negotiations or industrial competition; rational bidders anticipate such

follow-up impacts and adjust their bidding for the conjoint expected profitability of the auction and the aftereffects.

It is also common for collections of assets to be sold simultaneously or in rapid succession. When identical, as financial asset shares, attempting to have sales priced at opportunity costs (as indicated by rival bids) becomes much more complicated if an individual bidder is allowed to win multiple shares; most auction models prohibit multiple purchases, but few auctioneers do. If K identical assets are sold sequentially, expected profitability of competing in the last auction reduces bidding for the (K-1) asset, dollar-for-dollar, if winning at that stage would lead the bidder not to compete for the last asset.

Simultaneous auction of heterogeneous but related assets, rare until the mid-1990's, has arisen widely and attracted much attention, primarily in governmental sales of rights to use electromagnetic spectrum for digital transmissions, in privatization of factories and contractual responsibilities, and in firms' procurements of services, primarily logistics. Early attention focused on spectrum auctions by the U.S. Federal Communications Commission, but the key feature of potential synergies from winning particular collections of assets fits all these areas. If bidders can only compete asset-by-asset, synergies may go unbid and unattained; if bidders can make single bids on arbitrary packages of assets, the auction may be computationally unmanageable (in that both the problem of determining the revenue-maximizing collection of bids, and the bidder's problem of finding the minimal bid on a desired package to place it into the tentative winning set, become unboundedly complex). Opportunities to structure the set of permitted package bids so as to keep the auction computationally manageable have been illustrated. Package auctions have been implemented in all three situations.

Although the Internet has altered the patterns of auction usage, of general interest is the descriptive book on auctions and auctioneering by Cassady (1967), and the sociological interpretation of auctions by Smith (1990). Surveys include Wilson (1992) and Klemperer (2000); critical perspectives are given in Rothkopf and Harstad (1994); Klemperer (2002, 2003); and Harstad and Pekec (2008).

See

- ▶ Combinatorial Auctions
- ▶ Decision Analysis
- ▶ Game Theory
- ▶ Winner's Curse

References

Baye, M. R. (1996). *Advances in applied microeconomics* (Auctions, Vol. 6). Greenwich, CT: JAI Press.

Capen, E., Clapp, R., & Campbell, W. (1971). Competitive bidding in high risk situations. *Journal of Petroleum Technology, 23*, 641–653.

Cassady, R., Jr. (1967). *Auctions and auctioneering*. Berkeley: University of California Press.

Che, Y.-K., & Gale, I. (1996). Financial constraints in auctions: Effects and antidotes. In M. R. Baye (Ed.), *Advances in applied microeconomics* (Auctions, Vol. 6, pp. 97–120). Greenwich, CT: JAI Press.

Engelbrecht-Wiggans, R. (1980). Auctions and bidding models. *Management Science, 26*, 119–142.

Friedman, L. (1956). A competitive bidding strategy. *Operations Research, 4*, 104–112.

Harstad, R. M., & Pekec, A. (2008). Relevance to practice and auction theory: A memorial essay for Michael Rothkopf. *Interfaces, 38*, 367–380.

Harstad, R. M., & Rothkopf, M. H. (2000). An alternating recognition model of English auctions. *Management Science, 46*, 1–18.

Kagel, J. H., Levin, D., & Harstad, R. M. (1995). Comparative static effects of number of bidders and public information on behavior in second-price common-value auctions. *International Journal of Game Theory, 24*, 297–319.

Klemperer, P. (2000). Auction theory: A guide to the literature. In P. Klemperer (Ed.), *The economic theory of auctions*. Cheltenham, UK: Edward Elgar.

Klemperer, P. (2002). What really matters in auction design. *Journal of Economic Perspectives, 16*, 169–189.

Klemperer, P. (2003). Why every economist should learn some auction theory. In M. Dewatripont, L. Hansen, & S. Turnovsky (Eds.), *Advances in economics and econometrics: Invited lectures to 8th world congress of the econometric society*. Cambridge, UK: Cambridge University Press.

Krishna, V. (2010). *Auction theory* (2nd ed.). San Diego, CA: Academic Press.

McMillan, J. (1994). Selling spectrum rights. *Journal of Economic Perspectives, 8*, 145–162.

Milgrom, P. R., & Weber, R. J. (1982). A theory of auctions and competitive bidding. *Econometrica, 50*, 1089–1122.

Rothkopf, M. H. (1969). A model of rational competitive bidding. *Management Science, 15*, 362–373.

Rothkopf, M. H. (1994). Models of auctions and competitive bidding. In S. Pollock, A. Barnett, & M. H. Rothkopf (Eds.), *Handbooks in operations research* (Beyond the profit

motive: Public sector applications and methodology, Vol. 7). New York: Elsevier.
Rothkopf, M. H., & Harstad, R. M. (1994). Modeling competitive bidding: A critical essay. *Management Science, 40*, 364–384.
Smith, C. W. (1990). *Auctions: The social construction of value*. Berkeley: University of California Press.
Vickrey, W. (1961). Counter-speculation, auctions, and competitive sealed tenders. *Journal of Finance, 16*, 8–37.
Wilson, R. B. (1977). A bidding model of perfect competition. *Review of Economic Studies, 44*, 511–518.
Wilson, R. B. (1992). Strategic analysis of auctions. In R. Aumann & S. Hart (Eds.), *The handbook of game theory* (Vol. 1). Amsterdam: North-Holland/Elsevier.

Automation in Manufacturing and Services

Kalyan Singhal[1], Vijay K. Agrawal[2] and Matthew J. Liberatore[3]
[1]University of Baltimore, Baltimore, MD, USA
[2]The University of Nebraska at Kearney, Kearney, NE, USA
[3]Villanova University, Villanova, PA, USA

Introduction

Automation has affected both manufacturing and services. Automation's roots are in mechanization, which can be defined as the transfer of skills and manual activities to machine operation. Automation differs from mechanization in that it includes feedback for controlling the system (Odrey 1992). Automation is a dynamic technology that has been evolving for decades.

Automated reasoning is rarely included explicitly in the definition of automation, although machine learning, heuristics, and knowledge-based systems are increasing the scope of automation activities. But the industrialized world is moving towards building highly automated systems that are intelligent and operate in real time, with self-defined capabilities for carrying out pre-defined objectives over long periods of time in an uncertain environment (Kim and Chung 1991). Automation is defined here as the use of technological-based systems that replace routine physical labor and human reasoning by machines that perform operations with minimal or no human intervention. These machines should be as self-activated, self-acting, self-determining, self-regulating, and self-reliant as is practical. Automated systems can provide a heretofore unmatched level of performance along the four strategic dimensions of cost, quality, delivery, and flexibility (Singhal 1987). Parts of this presentation are based on Singhal et al. (1987) and Singhal (2011).

Automated Manufacturing

A typical automated manufacturing system can perform such operations as machining, welding, inspection, and assembly in industries as diverse as heavy machinery and light electronics. Its elements include computer-interfaced machine tools, robots, and automated material-handling and storage devices. The processing instructions stored in the computer memory of a machine tool enable it to perform customized operations automatically on each workpiece. Sensory devices in the machines and robots perform automatic inspection. The machines are linked by an automated material handling system and a central computer. The central computer provides the overall control of the system. This control includes routing and sequencing jobs, tracking their status, down-loading instructions to machine tools, and taking corrective actions. Other components of the system include a tool delivery system, and a common central buffer or local buffers at individual work stations. Applications of automation in manufacturing include a variety of technologies:

- Automatic machine tools: They include computer-numerical-controlled (CNC) tools that are controlled by software that is fairly easy to reprogram.
- Automatic materials-handling and storage systems: They include automatic storage and retrieval systems (AS/RS) and automatic guided vehicle (AGV) systems.
- Industrial robots: These are reprogrammable multi-function manipulators, often arm like in appearance, perform a variety of tasks.
- Flexible manufacturing systems (FMSs): These are manufacturing cells designed for mid-volume and mid-variety ranges of production (Stecke 1985, 1992). Typical FMSs consist of 5–25 machines that are linked together by common computer controllers and by automated material handling

devices. Because of the versatility of machine tools, for example, and the quick (seconds) cutting tool interchange capability, these systems are quite flexible with respect to the number of part types that can be produced simultaneously and in low (sometimes unit) batch sizes.

- Fixed automation: This is used when the sequence of operations is fixed, for example, high-volume assembly.
- Computer-aided design (CAD): CAD relies on computer graphics for designing products, and a full system includes databases of documentation and stored images of parts and assemblies. The designer can create new designs or modify existing designs on a CRT by using a light pen, a keyboard, or a joystick.
- Computer aided manufacturing (CAM): CAM includes various forms of manufacturing automation such as those listed above (Considine and Considine 1989; Odrey 1992).
- Computer-aided process planning (CAPP): CAPP uses computers to develop detailed process plans for producing parts or assemblies, and, as such, links CAD and CAM.

The use of CAD in contemporary engineering projects is especially important. CAD has been used to design products as simple as potato chips, kitchen cabinets, and customized swimsuits, and such complex products as the Boeing 777. Boeing's CAD system employed nine IBM mainframes, a Cray supercomputer, and 2,200 workstations and stored 3,500 billion bits of information. Boeing reduced its engineering design errors by more than 50% and designing and building a 777 plane took 10 months as compared to 18 months for Boeing's earlier planes, the 747 and the 767 (Laudon and Laudon 1998, pp. 618–621). CAD eliminates such activities as drawing blue prints and building prototypes, reducing costs, which was all quite critical in the Boeing efforts. More generally, CAD permits designers to make changes quickly and at very low cost. The designer can rotate the design to examine it from various angles, can split it apart to get a view of the inside, or zoom in on portions of the computer screen for close-ups.

An engineering perspective of automation in manufacturing is given in Chryssolouris et al. (2009). Radio frequency identification (RFID) has enormous potential for automated traceability in manufacturing systems. According to Ngai et al. (2007), its advantages include improved lead time; ability to offer customers timely information about an item's status and job completion time, improved maintenance operations; reduction in human errors, improved inventory management, automatic capture of data and resulting lower costs, and improved customer relationships.

Many automated manufacturing systems integrate several production stages resulting in interactions among many hardware and software components. Many of the benefits of automated systems accrue from this integration. For a number of reasons, this integration can dramatically increase the complexity of managing a manufacturing system. First, many automated manufacturing systems have several hierarchical levels, a large number of candidate decisions, and large data requirements. Second, management of these systems requires balancing multiple objectives with quantifiable and nonquantifiable trade-offs. Finally, the high degree of integration makes the management of automated manufacturing systems especially vulnerable to the stochastic variability of machine failures, operator absences, material shortages, and production requirements.

Since mathematical models can offer insights into the nature of the interactions among components in complex systems, OR/MS professionals can play a major role in the design, operation, and control of complex automated systems. Although these systems are complex, the central role played by computers enables low cost data collection for a wide range of events that occur within the system. The resulting availability of information greatly facilitates the use of OR/MS models in these computer-controlled systems. The contribution of models and other decision-making aids can be substantial because the large investments these systems require make the opportunity cost of suboptimal design and operation high. Time, talented professionals, and good data are also crucial to success.

Traded off against the costs of the capital investment and the human resources are a wide range of benefits attributed to automated manufacturing systems. These benefits include lower direct manufacturing costs resulting from reductions in setup time, processing time, labor requirements, lead time, inventory, and factory space; improved product conformance (quality); a greater variety of products at

little additional cost (economies of scope); the ability to respond rapidly to changes in design and demand, and flexibility in scheduling around equipment breakdowns.

Managers can sometimes obtain many of the benefits in improved quality and lead time that are attributed to automated systems by paying attention to problems in these areas, irrespective of the technology in place. Adopting total quality management, just-in-time practices, and design-for-manufacturability, for example, may achieve many of those benefits at a fraction of the capital cost. If this is the case, one must ask whether the additional benefits that can actually be attributed to automation can justify the cost. In the 1980s, an IBM printer manufacturing facility in Kentucky dramatically redesigned and simplified its product and process to pave the way for 100% automated assembly. IBM discovered that the redesigned printer was so simple that it could achieve high quality manual assembly at a lower cost than automated assembly.

It is useful to partition the issues related to analyzing and modeling automated manufacturing systems according to the following phases: choice of technology; design of the physical system; design of the production planning, scheduling, and control system; installation and start-up; and steady-state operation and improvements (Singhal et al. 1987):

The Choice of Technology — Conceptually, the decision rule for investing in an automated manufacturing system is like that for any other investment decision; the net benefits, tangible and intangible, should result in positive net present value to the firm. Because many of the benefits of automated manufacturing systems are difficult to quantify, the firms should first calculate the net present value on the basis of quantifiable benefits. If this value is negative, firms should use managerial judgment to decide whether the intangible benefits are greater than the shortfall. Analysts also use multicriteria approaches, such as the analytic hierarchy process or multiattribute utility theory, to compare different automation options. Canada and Sullivan (1989) and Liberatore (1990) reviewed models for investment in automated manufacturing systems and related issues.

It is crucial that a firm's technology acquisition decisions be consistent with its manufacturing strategy. Once the firm has established that the acquisition is congruent with its manufacturing strategy, it must consider several other issues. First, in addition to hardware costs, the investment cost includes the cost of software and training. Second, in a changeover from conventional manufacturing, the firm should include in its analysis the cost of changeover and the reduction in work-in-process and finished-goods inventory. Third, it cannot extrapolate the tangible benefits from the performance of a conventional manufacturing system; it must evaluate an initial design of the system to measure these benefits.

During the technology evaluation process, the firm should identify sources of risk and uncertainty related to the automated manufacturing system project. The stochastic variability of market demand, component supply, and competitive interaction will all affect the contribution of an automated manufacturing system to a firm. An additional source of uncertainty is organizational learning. In automated systems, as in most manufacturing systems, learning continues beyond the start-up phase into the steady-state operation phase. Learning increases equipment utilization, reduces the number of workers, increases flexibility in meeting changes in product demands and designs, improves quality, and increases variety.

Benefits that are difficult to quantify economically should be identified. These benefits include better and more consistent quality, economies of scope, shorter lead times, and flexibility in responding to changes in product designs and product demands. Some of these benefits derive from the firm's ability to do things not previously possible and to build new strategies to exploit the new capabilities.

The firm should not assume a static environment when comparing the option of investing in automated technologies with the option of retaining its old technologies. Whether or not a firm chooses to invest in automated technologies, its competitors may do so.

Another issue is whether the firm should acquire islands of automation in stages and then link them or whether it should go for complete automation in one step. With the former approach the firm can evaluate what it has implemented before proceeding with the next step. This approach also provides more scope for learning and mastering technology in stages. However, it will realize many of the benefits related to flexibility and product variety only when it links these islands. This may make the latter approach more desirable.

The Design of the Physical System — For most automated manufacturing systems, the output and flexibility requirements determine the basic design: the types and number of machines, robots, material handling devices, information processing capabilities, and human resources. Important decisions in the design of automated manufacturing systems concern the specifications of the structure of the control system and the sizes and locations of inventory and capacity buffers. Although most automated manufacturing systems are large and complex, they tend to be organized in hierarchical structures, allowing the designer to break up the computational and data requirements up so that modeling the system is possible. The higher levels of the hierarchical models have long horizons and use aggregated data. The lower levels have shorter horizons and use more detailed data. Queueing models (Buzacott and Yao 1986a, 1986b) can be used for the basic design. They are robust and computationally efficient, and they can predict the main output measures within 10% accuracy. Thus, the designer can explore a wide range of alternatives before selecting a small subset of designs for detailed evaluation.

Viswanadham and Narahari (1992, p.163) point out that "Markov chains constitute the basic model of discrete event systems and therefore of automated manufacturing systems." They further note (p. 7) that the "underlying stochastic process of most high-level models such as queues, queueing networks, and stochastic Petri nets turns out to be a Markov chain."

The detailed design of the system determines the layout, the number of pallets, the type and number of fixtures, the required accuracy of machines, the types of tool-changing systems, and the methods of feeding and locating parts at machines. The designer should integrate the planning, scheduling, and control of operations with the detailed design. Simulation, because of its versatility, is the tool most frequently used for detailed design. The designer plans the details of a common data base at this stage. In an ideal system, all decision makers in the organization have access to the same data so that engineering changes in products and processes can be entered as soon as they are finalized. The development of such a database for an automated manufacturing system is a major task requiring considerable effort and resources. For an application of analytical approaches to the design of an automated production line, see Burman, Gershwin, and Suyematsu (1998).

Design of the Production Planning, Scheduling, and Control System — Production planning consists of deciding when to produce which products; allocating machines, pallets, fixtures, operators, and tools; and determining policies for preventive maintenance and inspection. Scheduling problems include establishing work-releasing, sequencing, and priority rules. Control problems include determining policies for defect detection, equipment breakdown, repair, and real-time allocation of resources. The operations of an automated manufacturing system are under the control of a computer system that makes decisions, such as which parts to load into the system next and what workstations a particular batch is to visit next. Human intervention becomes necessary only when unusual or unanticipated events take place, such as machine failures, non-availability of materials, human errors, unscheduled maintenance, and changes in the operating environment. Some of the resulting resource/reallocation decisions require a combinatorial number of complex computations that need to be done quickly. In most cases, these computations cannot be done optimally in real time with the present speed of computers and current OR/MS models and artificial intelligence (AI). Combining OR/MS models and AI approaches seems promising. Crama, Oerlemans, and Spieksma (1996) review models for planning and scheduling.

The objectives for the models at each level of the control system hierarchy depend on the operating characteristics of the system and, thus, cannot always be specified without evaluating the alternatives at lower levels. Hence, system designers use detailed simulations to determine the appropriate planning, scheduling, and control system before its implementation.

Installation and Start-up – Many shortcomings in design and planning are exposed during installation and start-up. The systems may be incompatible with product design, the database may be inadequate, and the operators, managers, and support staff may not have the required skills. The models developed for steady-state operation of an automated manufacturing system are usually not applicable to its transient behavior during start up and shutdown. Equipment breakdowns may also cause transient behavior that interrupts steady-state operation. During start up,

however, one can expect frequent shutdowns and one can expect devote a good deal of effort to debugging and tuning the system.

Steady-State Operation and Improvements – There is considerable scope for learning during steady-state operation, and learning is facilitated if those who design and install the system also operate it. The learning objectives will be to increase equipment utilization, to reduce labor costs, to increase flexibility in meeting changes in product design and demand, to improve quality, to discover opportunities for improving product design, and to increase product variety. The firm can achieve these objectives by improving production planning, scheduling, and control; by developing preventive maintenance and repair policies; by devising strategies to cope with the consequences of equipment breakdowns; and by integrating the various functions in the firm, especially manufacturing and engineering. Two major determinants of learning are planned efforts and the existing skills of the operators, managers, and support staff. The firm can use total quality management and business process redesign to make some of these improvements.

Automated technologies are becoming capable of learning from experience and making decisions with little or no human intervention to optimize operations and minimize costs. For example, in manufacturing automation, artificial intelligence (AI) will become increasingly important in the move toward a true computer-integrated-manufacturing (CIM) environment, in which all of the organization's activities are linked through computers: order entry and customer billing, product design, manufacturing planning, and manufacturing control.

Services

The impact of automation on service is visible in everyday life: TV remote control, automated telephone answering systems, automated teller machines (ATMs), the World Wide Web, electronic commerce, and so forth. Such technologies as electronic imaging, electronic data interchange (EDI), and expert systems are having a major effect on work-flow automation. Electronic imaging involves scanning and digitizing documents (e.g., routine reports, expense-account reimbursements, and purchase orders) so that they can be stored in a database and retrieved. EDI allows information to be transmitted and shared electronically. Expert systems are programs that incorporate knowledge of experts concerning a particular set of decisions. Some examples of automation in services are noted below.

Food Service — In Arby's, Inc., an automated ordering machine asks the customers whether they intend to dine or take out and what they want, keeps a running tally of the cost, and if the customer does not order a drink, it suggests one. The machine has reduced order times by about 50% and improved accuracy. In McDonald's restaurants, an automated fry-maker drops fries into the basket, lowers the basket into the cooking oil, shakes them intermittently, and dumps the finished fries into a tray.

Retail Sales – Bar codes save time at the cash register and automatically update the store's inventory records as items are sold. Electronic commerce on the internet saves customers' time and reduces administrative and other retailing costs.

Financial Services – ATMs improve customer service and reduce the costs of transactions; electronic fund transfer systems have made possible direct payroll deposit and debit cards; and optical scanning in credit card and check processing operations reduces processing time and improves accuracy. Other applications include expert systems for loan applications, insurance underwriting, and security analysis.

Interorganizational Coordination – EDI speeds the exchange of information between locations by eliminating the time taken by regular mail and for data entry. Buyers in a retail chain, for example, decide which items are to be purchased and the automated system can them sort them and place orders with suppliers. Srinivasan, Kekre, and Mukhopadhyay (1994) reported that EDI technology facilitates the accurate, frequent, and timely exchange of information to coordinate the movement of materials between trading parties. Organizations implementing EDI must cooperate closely and establish coordination between the organizations (Cohen and Apte 1997).

Office automation – Automation includes word processing, spreadsheets, optical scanning, electronic mail, teleconferencing, and voice mail.

United States Postal Service – Optical character systems can read and sort 5,000 pieces of mail per hour as compared to 800 per hour for a human sorter and 1,650 per hour for a mechanical sorter.

Healthcare – CAT scanners, laboratory diagnosis machines, pacemakers, and expert systems help physicians to make diagnosis and prescribe treatments.

Education – Automation includes distance learning, bibliographic databases in libraries, and computer assisted instructions.

Hospitality Industry – Automation includes revenue management, electronic reservation systems, and message and wake-up call systems.

Most parts of the framework described earlier for automated manufacturing are also applicable to automated services, as are the OR approaches used for automated manufacturing. As in manufacturing, it is crucial that a firm's decisions about process design and technology acquisition are consistent with its strategy. The firm must make sure that its systems and procedures are properly designed or redesigned before introducing automation. A redesigned process is not always simpler than its predecessor (Davenport 1993). For example, a new underwriting process developed by Phoenix Home Life Mutual Insurance (Hartford, Connecticut) substantially reduced under writing time by making activities parallel rather than serial. But the new system turned out to be much more complex than the one it replaced.

In making decisions about automating service, one must consider both tangible and intangible benefits and risks and uncertainties associated with them. To justify investment in electronic imaging, the firm must assess its potential impact on work flow and the performance of the overall system. Firms use a number of multicriteria approaches to evaluate costs and benefits. The United States Postal Service used simulation to evaluate facilities and equipment for automating postal operations (Cebry et al. 1992) and a decision tree to choose between two alternatives for one of its automation plans (Ulvila 1987). A number of OR approaches, including queueing theory and mathematical programming, are used for designing the physical systems and operations planning, scheduling, and control systems in automated services. Kolesar (1984) described a queueing analysis of ATMs. Other examples of OR studies on automation in the service sector include studies to measure the effect of ATMs on branch labor productivity, to manage investment portfolios, to trade fixed-income securities, and to process loan applications. The continuing importance of improving productivity in the service sector will lead to further automation and to OR studies to improve automated systems.

Implementation: Human and Organizational Dimensions

To implement automation, firms must change their technology, their operations, the design of their organization, and the tasks people perform. An organization adopting and implementing automation must often retrain workers, invest in computer hardware and software, and deal with start-up problems until it stabilizes the manufacturing process and the organization.

Workers operating automated systems must be highly skilled. The organization must be designed to facilitate interaction among many departments – manufacturing, engineering, purchasing, marketing, and accounting. In a changeover from conventional system, the firm must plan for the redeployment of its workforce and the restructuring of the organization. For such pervasive organizational changes, the firm must bring all parties, including top management, all departments, professional staff, union leaders, and workers, into the discussion.

Upton and McAfee (1998) suggested that automation should be designed and implemented to broaden the roles of workers instead of constraining them. The implementation of automated technologies in manufacturing inflicts stress on workers and this can hinder the process. Karuppan and Schniederjans (1995) suggested a number of steps for three different categories of workers in automated design and automated manufacturing. For CAD/CAM workers, they suggest increased participation in project teams and early and thorough training, preferably cross-training in designing different products or different tools and in programming a variety of numerically controlled (NC) machines. This can improve workers' flexibility, enhance their ability to handle fluctuating workloads, and break the monotony in most work situations. For manufacturing-control-system employees, they suggest broadening the scope of workers' responsibilities (articulating critical success factors

for software enhancements and hardware upgrades and redesigning the layout of the manufacturing floor according to work flow), making their workstations more private and personal, teaching them proper posture in front of the screen, and investing in equipment with ergonomic features. For NC machine and robot operators, they suggest educating workers about safety, ensuring proper ventilation, providing adequate rest areas, cross training for different NC machines or robots, and broadening their responsibilities (machine programming, maintenance, quality control, keeping and analyzing breakdown records, evaluating the present equipment, and formulating criteria for improvements). Many of the steps suggested for implementation of automation in manufacturing are also useful in facilitating implementation of automation in services.

Concluding Remarks

Analysts routinely combine OR with the methods of AI approaches to take advantage of their synergy. AI helps automate the selection, development, and use of OR tools and models through intelligent modeling and decision support systems. For example, AI can be used to determine whether a mathematical programming model has a special structure that can be exploited by a more efficient algorithm. In addition, AI can be used to develop new models when knowledge about operations is complex and qualitative, e.g., by combining expert systems (intelligent computer programs that can solve difficult problems using knowledge and inference) with OR approaches in such areas as process planning and scheduling.

See

- ▶ Analytic Hierarchy Process
- ▶ Artificial Intelligence
- ▶ Decision Analysis
- ▶ Health Care Strategic Decision Making
- ▶ Multi-attribute Utility Theory
- ▶ Networks of Queues
- ▶ Operations Management
- ▶ Retailing
- ▶ Scheduling and Sequencing
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Burman, M., Gershwin, S. B., & Suyematsu, C. (1998). Hewlett-packard uses operations research to improve the design of a printer production line. *Interfaces, 28*(1), 24–36.

Buzacott, J. A., & Yao, D. D. (1986a). Flexible manufacturing systems: A review of analytical models. *Management Science, 32*, 890–905.

Buzacott, J. A., & Yao, D. D. (1986b). On queueing network models of flexible manufacturing systems. *Queueing Systems, 1*, 5–27.

Canada, J. R., & Sullivan, W. G. (1989). *Economic and multiattribute evaluation of advanced manufacturing technologies*. Englewood Cliffs, NJ: Prentice-Hall.

Cebry, M. E., deSilva, A. H., & DiLisio, F. J. (1992). Management science in automating postal operations: Facility and equipment panning in the United States postal service. *Interfaces, 22*(1), 110–130.

Chryssolouris, G., Mavrikios, D., Papakostas, N., Mourtzis, D., Michalos, G., and Georgoulias, K. (2009) *Digital manufacturing: History, perspectives, and outlook*. Proceedings of the Institution of Mechanical Engineers – Part B – Engineering Manufacture, May, Vol. 223, pp. 451–462.

Cohen, M. A., & Apte, U. M. (1997). *Manufacturing automation*. Chicago: Irwin.

Considine, D. M., & Considine, G. D. (Eds.). (1989). *Standard handbook of industrial automation*. New York: Chapman and Hall.

Crama, Y., Oerlemans, A. G., & Spieksma, F. C. R. (1996). *Production planning in automated manufacturing*. Heidelberg: Springer.

Davenport, T. H. (1993). *Process innovation*. Boston, MA: Harvard Business School Press.

Karuppan, C. M., & Schniederjans, M. C. (1995). Sources of stress in an automated plant. *Production and Operations Management, 4*(2), 108–126.

Kim, T. G., & Chung, M. (1991). *Embedding simulation modeling in development of high autonomy systems*. Proceedings of the Second Conference on Al, Simulation and Planning in High Autonomy Systems.

Kolesar, P. J. (1984). Stalking the endangered CAT: A queueing analysis of congestion at automatic teller machines. *Interfaces, 14*(6), 16–26.

Laudon, K. C., & Laudon, J. P. (1998). *Management information systems*. Upper Saddle River, NJ: Prentice Hall.

Liberatore, M. J. (Ed.). (1990). *Selection and evaluation of advanced manufacturing technologies*. Berlin: Springer.

Ngai, E. W. T., Cheng, T. C. E., Lai, K., Chai, P. Y. F., Choi, Y. S., & Sin, R. K. Y. (2007). Development of an RFID-based traceability system: Experiences and lessons learned from an aircraft engineering company. *Production and Operations Management, 16*, 554–568.

Odrey, N. G. (1992). *Maynard's industrial engineering handbook*. New York: McGraw-Hill.

Singhal, K. (1987). Introduction: The design and implementation of automated manufacturing systems. *Interfaces, 17*(6), 1–4.

Singhal, K. (2011). *Design of automated service systems*. Working paper, Baltimore: Merrick School of Business, University of Baltimore.

Singhal, K., Fine, C. H., Meredith, J. R., & Sun, R. (1987). Research and models for automated manufacturing. *Interfaces, 17*(6), 5–14.
Srinivasan, K., Kekre, S., & Mukhopadhyay, T. (1994). Impact of electronic data interchange technology on JIT shipments. *Management Science, 40*, 1291–1304.
Stecke, K. E. (1985). Design, planning, scheduling, and control problems of flexible manufacturing systems. *Annals of Operations Research, 3*, 3–12.
Stecke, K. E. (1992). Flexible manufacturing systems: Design and operating problems and solutions. In W. K. Hodson (Ed.), *Maynard's industrial engineering handbook* (4th ed.). New York: McGraw-Hill.
Ulvila, J. W. (1987). Postal automation (ZIP + 4) technology: A decision analysis. *Interfaces, 17*(2), 1–12.
Upton, D. M., & McAfee, A. P. (1998). Computer integration and catastrophic process failure in flexible production. *Production and Operations Management, 7*, 265–281.
Viswanadham, N., & Narahari, Y. (1992). *Performance modeling of automated manufacturing systems*. New Jersey: Prentice Hall.

Availability

Igor Ushakov
Qualcomm Inc., San Diego, CA, USA

Availability is a property of a system requiring it to be ready for performing its required operation or task at time t. (It is often also referred to as readiness.) This is clearly related to the main system property, reliability. The main measure of availability is the availability coefficient, $A(t)$, which is equal to the probability of finding the system in the operational state at the needed moment of time t.

If the process of the system's functioning is described in terms of an alternating sequence of lifetimes (i.e., times to failure) $\{X_i\}$ and repair times $\{Y_i\}$, then at any moment t, the availability coefficient can be determined as

$$A(t) = \Pr\{t \in X_i, i = 1, 2, \ldots\}.$$

For stationary processes, i.e., where t goes to ∞, the stationary availability coefficient is defined as

$$A = \lim_{t \to \infty} A(t) = \frac{E[X]}{E[X] + E[Y]}.$$

See

▶ Reliability of Stochastic Systems

References

Kozlov, B. A., & Ushakov, I. A. (1970). *Reliability handbook*. New York: Holt, Rinehart and Winston.
Ushakov, I. A. (Ed.). (1994). *Handbook of reliability engineering*. New York: Wiley.

Averch-Johnson Hyphothesis

▶ Economics and Operations Research

B

Backward Chaining

An approach to reasoning in which an inference engine endeavors to find a value for an overall goal by recursively finding values for subgoals. At any point in the recursion, the effort of finding a value for the immediate goal involves examining rule conclusions to identify those rules that could possibly establish a value for that goal. An unknown variable in the premise of one of these candidate rules becomes a new subgoal for recursion purposes.

See

► Expert Systems

Backward Kolmogorov Equations

In a continuous-time Markov chain with state $X(t)$ at time t, define $p_{ij}(t)$ as the probability that $X(t+s) = j$, given that $X(s) = i$, $s, t \geq 0$, and r_{ij} as the transition rate out of state i to state j. Then Kolmogorov's backward equations say that, for all states i, j and times $t \geq 0$, the derivatives $dp_{ij}(t)/dt = \sum_{k \neq i} r_{ik}\, p_{kj}\,(t) - v_i\, p_{ij}(t)$, where v_i is the transition rate out of state i, $v_i = \sum_j r_{ij}$.

See

► Markov Chains
► Markov Processes

Backward-Recurrence Time

Suppose events occur at times $T_1, T_2, \ldots$ such that the interevent times $T_k - T_{k-1}$ are mutually independent, positive random variables with a common cumulative distribution function. Choose an arbitrary time t. The backward recurrence time at t is the elapsed time since the most recent occurrence of an event prior to t.

Balance Equations

(1) In probability modeling, steady-state systems of equations for the state probabilities of a stochastic process found by equating transition rates. For Markov chains, such equations can be derived from the Kolmogorov differential equations or from the fact that the flow rate into a system state or level must equal the rate out of that state or level for steady state to be achieved. (2) In linear programming (usually referring to a production process model), constraints that express the equality of inflows and outflows of material.

See

► Markov Chains

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Balking

When customers arriving at a queueing system decide not to join the line and instead go away because they anticipate too long a wait.

See

▶ Queueing Theory

Bandit Model

▶ Multi-armed Bandit Problem

Banking

Stavros A. Zenios
University of Cyprus, Nicosia, Cyprus
University of Pennsylvania, Pennsylvania, PA, USA

Introduction

OR/MS techniques find applications in numerous and diverse areas of operation in a banking institution. Applications include the use of data-driven models to measure the operating efficiency of bank branches through data envelopment analysis, the use of image recognition techniques for check processing, the use of artificial neural networks for evaluating loan applications, and the use of facility location theory for opening new branches and placing automatic teller machines (e.g., Harker and Zenios 1999). A primary area of application is that of financial risk control in developing broad asset/liability management strategies. Papers that summarize these areas are Zenios (1993), Jarrow et al. (1994), and Ziemba and Mulvey (1998). This work can be classified into three categories: (1) pricing contingent cashflows, (2) portfolio immunization, and (3) portfolio diversification.

Pricing Contingent Cashflows

The fundamental pricing equation computes the price of a contingent cashflow as the expected net present value of the cashflows, discounted by an appropriate discount rate. In discrete time the pricing equation takes the form

$$P_T = E_S\left\{\sum_{t=0}^{T}\frac{C_{t+1}^S}{1+r_t^S}\right\} \tag{1}$$

where E denotes expectation over the set of scenarios indicated by index s, C_t^S denotes the cashflow received at period t under scenario s, r_t^S is the spot rate for the same period under the scenario s, and T denotes the maturity date. The vector (r_t) is known as the term structure of interest rates. For risk-free cashflows, the appropriate discount rate is the rate implied by the Treasury yield curve. At any given point in time, vector (r_{0t}^S) can be obtained using market data; this is the current term structure scenario. However, the temporal variation of the term structure is stochastic. This stochastic interest rate behavior, together with potential uncertainties in the level of the cashflows (i.e., the scenarios C_t^S) are the primary challenging issues behind the evaluation of (1).

One major strand of research is devoted to the development of stochastic models for the term structure of interest rates. Cox, Ingersoll and Ross (1985) first described the interest rate dynamics via the (continuous) diffusion process

$$dr = \kappa(\mu - r)dt + \sigma\sqrt{r}d\omega \tag{2}$$

Here, μ is the mean and σ the variance of the stochastic interest rate process, and $d\omega$ is the differential of a standard Wiener process. This model exhibits mean reversion with a drift factor $\kappa(\mu - r)$, and guarantees that interest rates remain positive. It is, however, a single factor model: the term structure of interest rates is represented by a single state variable, namely the spot rate, r.

A two-factor model for bond prices was developed by Brennan and Schwartz (1979). They considered two state variables, the spot rate r and a long-term (consol) rate L. The dynamics of these two variables are described by

$$\begin{cases} dr = b_1(r,L,t)dt + a_1(r,L,t)d\omega_1 \\ dL = b_2(r,L,t)dt + a_2(r,L,t)d\omega_2 \end{cases} \tag{3}$$

Here, the drift factors are denoted by the functions $b_1(r, L, t)$ and $b_2(r, L, t)$, and the variance terms are expressed by $a_1(r, L, t)$ and $a_2(r, L, t)$. The elements $d\omega_1$ and $d\omega_2$ are differentials of standard Wiener processes.

Despite the elegance of continuous-time models, since most practical applications deal with discrete time cashflows, there is interest in the development of discrete models. A popular choice of discrete models is based on binomial lattices. Such models typically assume that interest rates can move to one of two possible states, up or down, from period t to $t + 1$. The probability and magnitude of each step are calibrated using the Treasury yield curve and the volatility implied by the prices of traded option instruments. Ho and Lee (1986) and Black, Derman and Toy (1990) proposed some fundamental models. For example, the Black, Derman and Toy model described the spot rates by the process

$$r_t^{\sigma} = r_t^0(\kappa_t)^{\sigma}.$$

Here r_t denotes the spot rate that takes values r_t^{σ} with possible states $\sigma = 0,1, \ldots, t$; r_t^0 is the ground state; and κ_t is the volatility of the spot rate in period t.

Models such as those described above generate the discount rates used in the pricing of riskless cashflows. For risky contingent cashflows (e.g., cashflows with credit, default, lapse, prepayment, and other such risks), the discount rates must be adjusted with a suitable riskpremium. Such premiums can be computed from the observed market prices of actively traded securities with comparable risks through the use of option adjusted analysis (Babbel and Zenios 1992).

Another important modeling issue in evaluating (1) is the forecasting of the cashflow stream (C_t). Statistical analysis and econometric modeling can be used in this context, especially when dealing with the various complex securities that have emerged in the 1980s and 1990s, like callable corporate bonds, mortgage and other assetbacked securities, and a range of insurance products. This kind of modeling was represented for insurance products by Asay, Bouyoucos and Marciano (1993), and for mortgage-backed securities by Kang and Zenios (1992).

Portfolio Immunization

This is a portfolio management strategy for locking in a fixed rate of return during a prespecified horizon. It assumes that all risk in the returns of the securities is systematic, that is, all risks are due to some common underlying factor(s). Portfolio immunization aims at eliminating this systematic risk. In the case of fixed-income securities, systematic risk is primarily due to changes in the term structure. Portfolio immunization traditionally deals with this type of risk.

The actuary F.M. Reddington (1952) was the first to introduce the notion of immunization, and also specified conditions for immunization. Portfolio immunization became a popular strategy in the 1970s at the aftermath of interest rate deregulation in the U.S. and the volatility of the fixed-income markets that followed. Fisher and Weil (1971) defined immunization as follows:

> A portfolio of investments is immunized for a holding period if its value at the end of the holding period, regardless of the course of rates during the holding period, is at least as large as it would have been had the interest rate function remained constant throughout the holding period.

A portfolio of assets used to fund a stream of liabilities can be immunized if the following conditions are met: (1) The present value of the assets is equal to the present value of the liabilities, and (2) the duration of the assets is equal to the duration of the liabilities. The first condition guarantees that the target liabilities are funded if the interest rates remain constant throughout the target period. The second condition guarantees that assets and liabilities have identical sensitivities to parallel shifts of the interest rates. Hence, the target liabilities will be funded even if the term structure experiences parallel shifts. A general overview of portfolio immunization was given in Fabozzi (1991). Linear programming formulations are often used to structure immunized portfolios, as in Zenios (1993).

Briefly, let r_i be the yield of the ith security, and C_{it} be the cashflow of security i at time t. From the fundamental pricing (1), obtain the price of the ith security by

$$P_i = \sum_{t=1}^{T} C_{it}(1 + r_i)^{-t}.$$

The sensitivity of the price — or *dollar duration* — of security i is obtained by differentiating with respect to cashflow yield, $(\partial P_i / \partial r_i)$, to get

$$k_i = -\sum_{t=1}^{T} tC_{it}(1 + r_i)^{-(t+1)}.$$

Given the present value P_L and dollar duration k_L of its liabilities, an immunized portfolio can be structured by solving the linear program

$$\begin{aligned} \text{Maximize} \quad & \sum_i k_i r_i x_i \\ \text{s.t.} \quad & \sum_i P_i x_i = P_L \\ & \sum_i k_i x_i = k_L \\ & x_i \geq 0 \end{aligned}$$

The objective function above maximizes an approximation to the portfolio yield, obtained as the dollar duration-weighted average yield of the individual securities in the portfolio. Several variations exist on the theme of portfolio immunization. One extension is to structure a portfolio that matches not only present value and duration of assets with those of the liabilities, but that also matches convexity, i.e., second partial derivatives $(\partial^2 P_i / \partial r_i^2)$ as well. Another approach is to compute the sensitivity of the prices to more than one factor, than just to parallel shifts of interest rates. The precise form of these factors (i.e., parallel shifts, steepening of the term structure, or term structure inversions) can be obtained using factor analysis of market data. Factor analysis of the term structure was first proposed for the U.S. market by Litterman and Scheinkman (1988). The use of linear programming for factor immunization was proposed by Dahl (1993) and D'Ecclesia and Zenios (1994).

Portfolio Diversification

The principle of diversification — based on the adage "do not put all your eggs in one basket" — remains a universal strategy for portfolio management. It provides a systematic way for dealing with residual risk, assuming that residual risk is accurately represented by a function of the mean and variance in the return of the securities. It also assumes that investors have an (implied) utility function over the mean and variance of portfolio returns, favoring portfolios with higher means and lower variances. The efficient portfolios for an investor are those that achieve the highest expected return for a given level of variance or the smallest possible variance for a given level of return. Such portfolios are called mean-variance efficient portfolios. Mean-variance optimization models were proposed by Markowitz in the 1950s; Ingersoll (1987) gives an advanced textbook treatment.

Minimum variance portfolios, i.e., portfolios with the lowest level of variance for a given target expected return, can be structured using nonlinear quadratic programming. Define

Q as the covariance matrix $\{q_{ij}\}$ between securities i and j,
μ_i as the expected return of security i,
μ_p as the target expected return of the portfolio, and
X_i as the fraction of the portfolio in security i.

Assuming that no short sales are allowed ($x_i \geq 0$ for all i), formulate the problem as

$$\begin{aligned} \text{Minimize} \quad & x^T Q_x \\ \text{s.t.} \quad & \sum_i \mu_i x_i = \mu_p \\ & \sum_i x_i = 1 \\ & x_i \geq 0 \end{aligned}$$

Other constraints, like limits on portfolio turnover, on minimum holdings, or limits of investments in different market segments, etc., can be captured with more complex formulations. These issues have been addressed by Perold (1984). See also the articles in Zenios (1993) and Ziemba and Mulvey (1998).

The major area of investigation in implementing minimum variance models in practice is in the estimation of the covariance matrix. Factor models that relate the returns and variances of individual securities to a set of common factors are widely used in practice (Elton and Gruber 1984).

Mean-variance models have traditionally been used in managing portfolios of equities and for strategic asset allocation. By contrast, fixed-income portfolio management has traditionally been based on the principles of portfolio immunization. In the 1980s,

however, there was a convergence of portfolio management tools towards the ideas of portfolio diversification. More complex fixed income securities (e.g., corporate callable bonds, high-yield bonds, mortgages and other asset-backed securities) have very volatile returns. The notion of duration, as a measure of sensitivity, is extremely restrictive for such instruments. Mulvey and Zenios (1993) advocated the use of diversification models for fixed-income portfolios, indicating how pricing models can be developed to generate scenarios of holding period returns in order to calibrate the models, and illustrating that such models produce better results than traditional portfolio immunization strategies.

Another development deals with the asymmetric returns of fixed-income securities, especially those with embedded options. Mean-variance models are valid assuming a symmetric distribution of return. Furthermore, they penalize both upside and downside deviations from a target return. Development of more practical models for dealing with asymmetric returns and penalizing differentially upside from downside risk include the mean-absolute deviation model of Konno and Yamazaki (1991), the expected utility optimization models of Grauer and Hakkanson (1985), and the dynamic, multiperiod models of Kallberg, White and Ziemba (1982), Mulvey and Vladimirou (1992), and Golub et al. (1995).

See

- ▶ Data Envelopment Analysis
- ▶ Facility Location
- ▶ Financial Engineering
- ▶ Financial Markets
- ▶ Linear Programming
- ▶ Neural Networks
- ▶ Portfolio Theory: Mean-Variance Model
- ▶ Quadratic Programming
- ▶ Utility Theory

References

Asay, M. R., Bouyoucos, P. J., & Marciano, A. M. (1993). An economic approach to valuation of single premium deferred annuities. In S. A. Zenios (Ed.), *Financial optimization* (pp. 100–135). Cambridge: Cambridge University Press.

Babbel, D. F., & Zenios, S. A. (1992). Pitfalls in the analysis of option-adjusted spreads. *Financial Analysts Journal, 48*, 65–69.

Birge, J., & Linetsky, V. (Eds.). (2007). *Handbooks in operations research and management science: Financial engineering*. Maryland Heights, MO: Elsevier Science.

Black, F., Derman, E., and Toy, W. (1990). A one-factor model of interest rates and its application to treasury bond options. *Financial Analysts Journal*, 33–39.

Brennan, M. J., & Schwartz, E. S. (1979). A continuous time approach to the pricing of bonds. *Banking and Finance Journal, 3*, 133–155.

Cornuejols, G., & Tutuncu, R. (2007). *Optimization methods in finance*. Cambridge: Cambridge University Press.

Cox, J. C., Jr., Ingersoll, J. E., & Ross, S. A. (1985). A theory of the term structure of interest rates. *Econometrica, 53*, 385–407.

Dahl, H. (1993). A flexible approach to interest-rate risk management. In S. A. Zenios (Ed.), *Financial optimization* (pp. 189–209). Cambridge: Cambridge University Press.

D'Ecclesia, R., & Zenios, S. A. (1994). Factor analysis and immunization in the Italian bond market. *Journal of Fixed Income, 4*, 51–58.

Elton, E., & Gruber, M. (1984). *Modern Portfolio theory and investment analysis*. New York: Wiley.

Fabozzi, F. J. (Ed.). (1991). *The handbook of fixed-income securities*. Homewood, IL: Business One Erwin.

Fisher, L., & Weil, R. (1971). Coping with the risk of interest-rate fluctuations: Returns to bondholders from naive and optimal strategies. *Journal of Business, 44*, 408–431.

Golub, B., Holmer, M., McKendall, R., Pohlman, L., & Zenios, S. A. (1995). Stochastic programming models for money management. *European Journal of Operational Research, 85*, 282–296.

Grauer, R. R., & Hakansson, N. H. (1985). Returns on levered actively managed long-run portfolios of stocks, bonds and bills. *Financial Analysts Journal, 41*, 24–43.

Harker, P. T., & Zenios, S. A. (1999). *Performance of financial institutions: Efficiency, innovation, regulation*. Cambridge: Cambridge University Press.

Ho, T. S. Y., & Lee, S.-B. (1986). Term structure movements and pricing interest rate contingent claims. *Journal of Finance, 41*, 1011–1029.

Ingersoll, J. E., Jr. (1987). *Theory of financial decision making. Studies in financial economics*. Lanham, MA: Row-man and Littlefield.

Jarrow, R., Maksimovic, M., & Ziemba, W. (Eds.). (1994). *Handbooks in operations research and management science: finance*. Amsterdam: North Holland.

Kallberg, J. G., White, R. W., & Ziemba, W. T. (1982). Short term financial planning under uncertainty. *Management Science, 28*, 670–682.

Kang, P., & Zenios, S. A. (1992). Complete pre-payment models for mortgage backed securities. *Management Science, 38*, 1665–1685.

Konno, H., & Yamazaki, H. (1991). A mean-absolute deviation portfolio optimization model and its applications to the Tokyo stock market. *Management Science, 37*, 519–531.

Litterman, R., & Scheinkman, J. (1988). *Common factors affecting bond returns*. Technical report, Goldman, Sachs & Co., Financial Strategies Group, September.

Markowitz, H. (1952). Portfolio selection. *Journal of Finance, 7*, 77–91.
Mulvey, J. M., & Vladimirou, H. (1992). Stochastic network programming for financial planning problems. *Management Science, 38*, 1643–1664.
Mulvey, J. M., & Zenios, S. A. (1994). Capturing the correlations of fixed-income instruments. *Management Science, 40*, 1329–1342.
Perold, A. F. (1984). Large-scale portfolio optimization. *Management Science, 30*, 1143–1160.
Reddington, F. M. (1952). Review of the principles of life-office valuations. *Journal of Institute Actuaries, 78*, 286–340.
Scott, F.R., Roll, R. (1989). Prepayments on fixed-rate mortgage-backed securities. *Journal of Portfolio Management*, Spring, 73–82.
Zenios, S. A. (Ed.). (1993). *Financial optimization*. Cambridge: Cambridge University Press.
Ziemba, W. T., & Mulvey, J. M. (1998). *Worldwide asset and liability modeling*. Cambridge: Cambridge University Press.

Bar Chart

▶ Gantt Charts
▶ Quality Control

Barrier Functions and their Modifications

Roman A. Polyak
George Mason University, Fairfax, VA, USA

Introduction

In the mid-1950s and the early 1960s, Frisch (1955) and Carroll (1961) proposed the use of Barrier Functions (BFs) for constrained optimization. Since then, the BFs have been extensively studied, with particularly major work in the area due to Fiacco and McCormick (1968) who developed the Sequential Unconstrained Minimization Technique (SUMT). Currently, methods based on barrier functions make up a considerable part of modern optimization theory.

Barrier Functions

Consider the constrained optimization problem

$$x^* \in X^* = \arg\min\{f(x) | x = \in \Omega\} \tag{1}$$

where $\Omega = \{x: g_i(x) \geq 0, i = 1, \ldots, m\}$, $f: \Re^n \to \Re$ is convex, all $g_i: \Re^n \to \Re$ are concave, $m > n$, and X^* is the set of values minimizing $f(x)$ on Ω. Frisch's logarithmic barrier function F: int $\Omega \times \Re_{++} \to \Re$ is defined by formula

$$F(x, \mu) = f(x) - \mu\Sigma \ln g_i(x) \tag{2}$$

and Carroll's hyperbolic barrier function C: int $\Omega \times \Re_{++} \to \Re$ is defined as

$$C(x, \mu) = f(x) + \mu\Sigma \ln g_i^{-1}(x).$$

Assume that X^* is bounded and $\ln t = -\infty$ for $t \leq 0$; then for any $\mu > 0$, there exists a minimum of $F(x, \mu)$ in $\Re^n$, denoted by

$$(x, \mu) = \arg\min\{F(x, \mu) | x \in \Re^n\}. \tag{3}$$

Therefore

$$\begin{aligned} \nabla_x F(x, \mu), \mu) &= \nabla f(x(\mu)) - \Sigma\mu g_i^{-1}(x(\mu))\nabla g_i(x(\mu)) \\ &= \nabla f(x(\mu)) - \sum \lambda_i(\mu)\nabla g_i(x(\mu)) \\ &= \nabla_x L(x(\mu), \lambda(\mu)) = 0 \end{aligned} \tag{4}$$

where $L(x, \lambda) = f(x) - \sum \lambda_i g_i(x)$ is the Lagrangian for the problem (1). Also $g_i(x(\mu)) > 0, i, = 1, \ldots, m$ and

$$\lambda_i(\mu) = \mu g_i^{-1}(x(\mu)) > 0, \quad i = 1, \ldots, m. \tag{5}$$

Hence $x(\mu) \in$ int Ω, $\lambda(\mu) = (\lambda_i(\mu), i, = 1, \ldots, m) \in \Re^m{}_{++}$ and due to (4)

$$L(x(\mu), \lambda(\mu)) = \min\{L(x, \lambda(\mu) | x \in \Re^n\}.$$

Consider the dual problem to (Eq. 1)

$$\lambda^* \in L^* = \text{Arg}\max\{d(\lambda) | \lambda \in \Re_+^m\} \tag{6}$$

where $d(\lambda) = \min\{L(x, \lambda) \mid x \in \Re^n\}$ and L^* is the set of maxima of $d(\lambda)$ on $\Re^m$. The vector $x(\mu)$ is interior primal, the vector $\lambda(\mu)$ is interior dual, and due to (Eq. 5) the primal-dual gap is

$$\begin{aligned} \Delta(\mu) = f(x(\mu)) - d(\lambda(\mu)) &= f(x(\mu)) - L(x(\mu), \lambda(\mu)) \\ &= \Sigma\lambda_i(\mu)g_i(x(\mu)) = m\mu. \end{aligned}$$

Therefore

$$\mu \to 0 \Rightarrow \Delta(\mu) \to 0 \Rightarrow f(x(\mu)) \to f(x^*) \text{ and } d(\lambda(\mu)) \to d(\lambda^*).$$

The primal barrier trajectory $\{x(\mu)\}$ and the primal-dual trajectory $\{x(\mu), \lambda(\mu)\}$ are critical elements in both SUMT (Fiacco and McCormick 1968) and recent developments in Interior Point Methods (IPMs).

Interest in barrier and distance functions was revived after N. Karmarkar (1984) published his polynomial projective scaling method for linear programming (LP) calculations. The connection between Karmarkar's method and the Newton log barrier method for LP calculations was discovered by Gill, Murray, Saunders, Tomlin and Wright (1986). Since then the interest to BFs grew dramatically and IPMs became the main stream in modern optimization. Hundreds of papers and several books have been published recently on the matter (see Nesterov and Nemirovsky 1994; Roos et al. 1997; Wright 1997; Ye 1997).

The main idea of the path-following IPMs (see Gonzaga 1992; Renegar 1988) is to replace in a sense the unconstrained minimization problem (Eq. 3) by one Newton step for solving the system $\Delta_x F(x, \mu) = 0$. The basic path-following IPM consists of performing a Newton step toward the solution $x(\mu)$ of the system

$$\nabla_x F(x, \mu) = 0 \tag{7}$$

followed by the barrier parameter update.

For a given $\mu > 0$ one finds an approxmation x for $x(\mu)$, the so-called "warm" start. The warm start belongs to the area where Newton method for the system (Eq. 7) is well-defined (Smale 1986), that is, from x as a starting point the method

$$\hat{x} = x - (\nabla^2_{xx} F(x, \mu))^{-1} \nabla_x F(x, \mu) \tag{8}$$

converges to $x(\mu)$ quadratically. The step of the path-following method consists in replacing x by $\hat{x}$ and μ by $\hat{\mu} = \mu(1 - \alpha/\sqrt{m})$ where $\alpha > 0$ is independent on $m > n$.

In the late 1980s Nesterov and Nemirovsky (1994) discovered the self-concordant property of the function $F(x, \mu)$ for important classes of constrained optimization problems including LP, QP, and QP with quadratic constraints. A function ϕ: int $\Omega \to \Re$ is self-concordant if it is convex, three times differentiable, and for any $x \in$ int Ω any $h \in \Re^n$ on the interval $I = \{t/x + th \in \text{int } \Omega\}$, the function ϕ: $I \to R$ defined by $\phi(t) = \phi_{x,\, h}(t) = \phi(x + th)$ satisfies the following inequality

$$\varphi'''(0) \le 2(\varphi''(0))^{3/2}.$$

The self-concordant property guarantees that if x is well defined for the system $\Delta_x F(x, \mu) = 0$ then $\hat{x}$ will be well defined for the system $\nabla_x F(x, \hat{\mu}) = 0$. The polynomial complexity of the path-following method for LP follows immediately from the fact that each Newton step shrinks the primal-dual gap by $(1 - \alpha/\sqrt{m})$, where $\alpha > 0$ is independent on m.

The primal-dual algorithms have emerged as the most important and useful class of IPMs (see Wright 1997). On the computational side, the most successful implementation (see Lustig et al. 1992) is based on the Mehrotra predictor-corrector algorithm (Mehrotra 1992). The BFs became the basic tool in the IPM, but the BFs still have their inherent drawbacks: these function, as well as their derivatives, do not exist at the solution; and they grow infinitely large together with the condition number of their Hessians when the approximation approaches the solution and the area where the Newton method is well-defined shrinks to a point.

To eliminate the drawbacks, while still retaining the nice properties of the barrier functions, modified barrier functions (MBFs) were introduced in the early 1980s for both LP and NLP calculations (Polyak 1986, 1992, 1996). The MBFs are particular cases of the Nonlinear Rescaling Principle, which consists of transforming the objective function and/or the constraints into an equivalent problem and using the classical Lagrangian for the equivalent problem in both theoretical analysis and numerical methods (Polyak 1986).

Modified Barrier Functions

Consider the constrained optimization problem

$$g_i(x) \ge 0, i = 1, \ldots, m \tag{9}$$

is equivalent to $\mu \ln(\mu^{-1} g_i(x) + 1) \geq 0$, $i = 1, \ldots, m$. Therefore problem (1) is equivalent to

$$x^* \in X^* = \operatorname{Argmin}\{f(x) / \mu \ln(\mu^{-1} g_i(x) + 1) \geq 0, \quad i = 1, \ldots, m\} \tag{10}$$

where the constraints are transformed by $\psi(t) = \ln(t + 1)$, and rescaled by $\mu = 0$. The classical Lagrangian for the equivalent problem (10)

$$F(x, \lambda, \mu) = f_0(x) - \mu \Sigma \lambda_i \ln(\mu^{-1} g_i(x) + 1),$$

is the logarithmic MBF which corresponds to Frisch's log-barrier function (2). For any $\mu > 0$, the system (9) is equivalent to

$$\mu[(\mu^{-1} g_i(x) + 1)^{-1} - 1] \leq 0, i = 1, \ldots, m$$

where the constraints transformation is given by $v(t) = (t + 1)^{-1} - 1$. The classical Lagrangian for the equivalent problem is the hyperbolic MBF

$$C(x, \lambda, \mu) = f_0(x) + \mu \Sigma \lambda_i [(\mu^{-1} g_i(x) + 1)^{-1} - 1],$$

which corresponds to Carroll's hyperbolic barrier function (3).

The MBF's properties make them fundamentally different from the BFs. The MBFs, as well as their derivatives, exist at the solution, and for any Karush-Kuhn-Tucker pair (x^*, λ^*) and any $\mu > 0$, the following critical properties hold:

P1. $F(x^*, \lambda^*, \mu) = C(x^*, \lambda^*, k) = f_0(x^*)$;

P2. $\nabla_x F(x^*, \lambda^*, \mu) = \nabla_x C(x^*, \lambda^*, \mu) = \nabla_x L(x^*, \lambda^*) = 0$;

P3. $\nabla_{xx} F(x^*, \lambda^*, \mu) = \nabla_{xx} L(x^*, \lambda^*) + \mu^{-1} \nabla g^T(x^*) \Lambda^* \nabla g(x^*)$,

$\nabla_{xx} C(x^*, \lambda^*, \mu) = \nabla_{xx} L(x^*, \lambda^*) + 2\mu^{-1} \nabla g^T(x^*) \Lambda^* \nabla g(x^*)$.

where $\wedge = \operatorname{diag}(\lambda_i)$ and $\Delta g(x) = J[g(x)]$ is the Jacobian of the vector-function $g(x)^T = (g_i(x), i = 1, \ldots, m)$.

The MBF's properties resemble that of augmented Lagrangians (Bertsekas 1982; Golshtein and Tretyakov 1974; Hestenes 1969; Mangasarian 1975; Polyak and Tretyakov 1973; Powell 1969; Rockafellar 1973). One can consider the MBFs as interior augmented Lagrangians. At the same time, MBFs have some distinctive features, which make them different from both quadratic augmented Lagrangian (Rockafellar 1973) and nonquadratic augmented Lagrangian (Bertsekas 1982). The MBFs' properties lead to the following multipliers method.

Let $\mu > 0$, $\lambda^0 = e = (1, \ldots, 1) \in \Re^m$ and $x^0 \in \Omega_\mu = \{x | g_i(x) \geq -\mu, i = 1, \ldots, m\}$. The logarithmic MBF method consists of generating two sequences $\{x^s\}$ and $\{\lambda^s\}$:

$$x^{s+1} \in \arg\min\{F(x, \lambda^s, \mu) | x \in \Re^n\} \tag{11}$$

and

$$\lambda^{s+1} = \operatorname{diag}[\mu^{-1} g_i(s^{s+1}) + 1]^{-1} \lambda^s. \tag{12}$$

There is a fundamental difference between the logarithmic MBF method and SUMT or other IPM that is based on BFs. The MBF method converges to the primal-dual solution with *any* fixed $\mu > 0$ for any convex programming which has bounded optimal primal and dual solutions (Jensen and Polyak 1994). Moreover, for LP calculations, M. Powell proved that for any fixed barrier parameter, the MBF method produces such primal sequences that the objective function tends to its optimal value and constraints violations tend to zero with R-linear rate (Powell 1995).

If the second order optimality conditions hold then the primal-dual sequence converges with Q-linear rate:

$$\max\{\| x^{s+1} - x^* \|, \| u^{s+1} - u^* \|\} \leq c\mu \| u^s - u^* \| \tag{13}$$

where $c > 0$ is the condition number of the constrained optimization problem, which depends on the input data and the size of the problem, but it is independent on $\mu > 0$ (Polyak 1992).

The numerical realization of the MBF method leads to the Newton MBF. The Newton method is used to find an approximation for x^s, followed by the Lagrange multiplier update. Due to the convergence of the MBF method under the fixed barrier parameter $\mu > 0$, both the condition number of the MBF Hessian and the area where the Newton method is well defined remain stable. These properties contribute to both numerical stability and complexity, and they lead to the discovery of the "hot" start phenomenon in constrained optimization. It means that from some

point on for large classes of nondegenerate-constrained optimization problems including LP, QP and QP with quadratic constraints, the approximation for the primal minimizer will remain in the Newton area after each Lagrange multipliers update (Polyak 1992; Melman and Polyak 1996).

Due to (13) from the "hot" start on it takes only (ln ln ε^{-1}) Newton steps to improve the primal-dual approximation by a given factor $0 < q < 1$ as soon as $\mu \leq qc^{-1}$. The neighborhood of (x^*, λ^*) where the "hot" start occurs can be characterized by the condition number $c > 0$. Using the IPM with the Shifted Barrier Function (SBF) $S(x, \mu) = f(x) - \mu \sum \ln(\mu^{-1} g_i(x) + 1)$, which is self-concordant for the same classes of problem as $F(x, \mu)$, it takes $0(\sqrt{m} \ln c)$ to reach the "hot" start.

Combining the IPM based on SBF with the Newton MBF method, it is possible to improve substantially the complexity bounds for nondegenerate LP, QP and QP with quadratic constraints. In particular, for nondegenerate QP the total number of Newton step sufficient to obtain an approximation for (x^*, λ^*) with accuracy $\varepsilon = 2^{-L}$ is

$$N = 0(\sqrt{m} \ln c) + 0((L - \ln c) \ln m),$$

where L is the input length, $c > 0$ is the condition number of QP and $n < m$ (Melman and Polyak 1996).

The MBF method has an interesting dual interpretation. Assuming that the dual function $d(\lambda)$ is differentiable,

$$\Delta d(\lambda) = -g(x(\lambda))$$

where $x(\lambda) = \arg\min\{L(x, \lambda) | x \in \Re^n\}$ and $g(x(\lambda)) = (g_i(x(\lambda)), i = 1, \ldots, m)$, that is,

$$\nabla d(\lambda^{s+1}) = -g(x^{s+1}). \tag{14}$$

From the formula (12) for the Lagrange multipliers update

$$\begin{aligned} g_i(x^{s+1}) &= \mu \psi'^{-1}(\lambda_i^{s+1}/2\lambda_i^s) \\ &= \mu \psi^{*\prime}(\lambda_i^{s+1}/2\lambda_i^s), i = 1, \ldots, m \end{aligned} \tag{15}$$

where $\psi^*(s) = \inf\{st - \psi(t)\} = 1 - s + \ln s$ is the Legendre transformation of $\psi(t) = \ln(t + 1)$. Using (15), rewrite (14) as

$$\nabla d(\lambda^{s+1}) + \mu \Sigma \psi^{*\prime}(\lambda_i^{s+1}/\lambda_i^s) e_i = 0$$

where $e_i = (0, \ldots, 1, \ldots, 0)$. Hence,

$$\begin{aligned} \lambda^{s+1} &= \arg\max\{d(\lambda) + \mu \Sigma \lambda_i^s \psi^*(\lambda_i/\lambda_i^s)/\lambda \in \Re_+^m\} \\ &= \arg\max\{d(\lambda) - \mu D(\lambda, \lambda^s)/\lambda \in \Re_+^m\} \end{aligned} \tag{16}$$

where $D(\lambda, \lambda^s) = \sum \lambda_i^s \varphi(\lambda_i/\lambda_i^s)$ – a ϕ -divergence entropy-like distance with the kernel $\phi = -\psi^*$. Note that (16) is an IPM for the dual problem (see Teboulle 1993; Polyak and Teboulle 1997).

The formula (12) is in fact a method for solving the dual problem (6). It can be rewritten as

$$\lambda_i^{s+1}(1 - \mu^{-1} \nabla_{\lambda_i} d(\lambda_i^{s+1})) = \lambda_i^s, i = 1, \ldots, m. \tag{17}$$

Such a method is a well-known multiplicative image reconstruction algorithm for positron emission tomography (Eggermont 1990). On the other hand, it is nothing but the implicit Euler method for numerical solution of the following system of ordinary differential equations

$$\frac{d\lambda_i}{dt} = \mu^{-1} \lambda_i \frac{\partial d(\lambda)}{d\lambda_i}, \lambda(0) = \lambda_0, \ i = 1, \ldots, m$$

and $\lim_{t \to \infty} \lambda(t) = \lambda^*$, which is the solution of following nonlinear complementarity problem

$$\begin{aligned} \nabla d(\lambda) \leq 0, \ \lambda \leq 0 \\ \lambda^T \nabla d(\lambda) = 0. \end{aligned}$$

See

- Classical Optimization
- Computational Complexity
- Interior-Point Methods for Conic-Linear Optimization
- Nonlinear Programming

References

Bertsekas, D. (1982). *Constrained optimization and lagrange multipliers methods*. New York: Academic.

Carroll, C. (1961). The created response surface technique for optimizing nonlinear restrained systems. *Operations Research, 9*, 169–184.

Eggermont, P. (1990). Multiplicative iterative algorithm for convex programming. *Linear Algebral and its Applications, 130*, 25–42.
Fiacco, A. V., & McCormick, G. P. (1968). *Nonlinear programming: Sequential unconstrained minimization techniques*. New York: Wiley.
Frisch, K. (1955). *The logarithmic potential method of convex programming*. Technical Memorandum, May 13, University Institute of Economics, Oslo.
Gill, P., Murray, W., Saunders, M., Tomlin, J., & Wright, M. (1986). On projected barrier methods for linear programming and an equivalence to Karmarkar's projective method. *Mathematical Programming, 36*, 183–209.
Golshtein, E. G., & Tretyakov, N. V. (1974). Modified Lagrangean functions. *Economics and Mathematical Methods, 10*, 568–591 (Russian).
Gonzaga, C. (1992). Path following methods for linear programming. *SIAM Review, 34*, 167–224.
Hestenes, M. (1969). Multiplier and gradient methods. *Journal of Optimization Theory and Applications, 4*, 303–320.
Huard, P. (1967). Resolution of mathematical programming with nonlinear constraints by the method of centers. In J. Abadie (Ed.), *Nonlinear programming*. Amsterdam: North-Holland.
Jensen, D., & Polyak, R. (1994). The convergence of the modified barrier method for convex programming. *IBM Journal of Research and Development, 38*, 307–321.
Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica, 4*, 373–395.
Lustig, I., Marsten, R., & Shanno, D. (1992). On implementing Mehrotra's predictor-corrector interior point method for linear programming. *Siam Journal of Optimization, 2*, 435–449.
Mangasarian, O. (1975). Unconstrained Lagrangians in nonlinear programming. *Siam Journal of Control, 13*, 772–791.
Mehrotra, S. (1992). On the implementation of primal-dual interior point method. *Siam Journal of Optimization, 2*, 575–601.
Melman, A., & Polyak, R. (1996). The Newton modified barrier method for QP. *Annals of Operations Research, 62*, 465–519.
Nesterov, Y., & Nemirovsky, A. (1994). *Interior point polynomial algorithms in convex programming*. Philadelphia: SIAM Studies in Applied Mathematics.
Polyak, B. T., Tretyakov, N. V. (1973). The method of penalty bounds for constrained extremum problems. Zh. Vych. Mat. iMat. Fiz, 13, 34–46, *USSR Computational Methods Mathematics and Physics 13*, 42–58.
Polyak, R. (1986). *Controlled processes in extremal and equilibrium problems*. Moscow (Russian): VINITY.
Polyak, R. (1992). Modified Barrier functions (theory and methods). *Mathematical Programming, 54*, 177–222.
Polyak, R. (1996). *Modified Barrier functions in linear programming*. Research Report, Department of Operations Research, George Mason University, pp. 1–56.
Polyak, R. (1997). Modified interior distance functions. *Contemporary Mathematics, 209*, 183–209.
Polyak, R., & Teboulle, M. (1997). Nonlinear rescaling and proximal-like methods in convex optimization. *Mathematical Programming, 76*, 265–284.
Powell, M. (1969). A method for nonlinear constraints in minimization problems. In R. Fletcher (Ed.), *Optimization*. New York: Academic.
Powell, M. (1995). Some convergence properties of the modified log barrier method for linear programming. *SIAM Journal of Optimization, 5*, 695–739.
Renegar, J. (1988). A polynomial-time algorithm, based on Newton's method for linear programming. *Mathematical Programming, 40*, 59–93.
Rockafellar, R. T. (1973). The multiplier method of hestenes and powell applied to convex programming. *Journal of Optimization Theory and Applications, 12*, 555–562.
Roos, C., Terlaky, T., & Vial, J.-P. (1997). *Theory and algorithms for linear optimization: an interior point approach*. New York: Wiley.
Smale, S. (1986). A Newton method estimates from data at one point. In R. Ewing (Ed.), *The merging of disciplines in pure, applied and computational mathematics*. New York/Berlin: Springer.
Teboulle, M. (1993). Entropic proximal mappings with application to nonlinear programming. *Mathematics of Operations Research, 17*, 670–690.
Wright, S. (1997). *Primal-dual-interior point methods*. Philadelphia: SIAM.
Ye, Y. (1997). *Interior point algorithms: Theory and analysis*. New York: Wiley.

Basic Feasible Solution

A nonnegative basic solution to a set of $(m \times n)$ linear equations $\boldsymbol{Ax} = \boldsymbol{b}$, where $m \leq n$. The major importance of basic feasible solutions is that, for a linear-programming problem, they correspond to extreme points of the convex set of solutions. The simplex algorithm moves through a sequence of adjacent extreme points (basic feasible solutions).

See

- ▶ Adjacent (Neighboring) Extreme Points
- ▶ Basic Solution
- ▶ Linear Programming

Basic Solution

For a set of $(m \times n)$ linear equations $\boldsymbol{Ax} = \boldsymbol{b}$ $(m \leq n)$, with rank m, a basic solution is a solution obtained by setting $(n - m)$ variables equal to zero and solving for the remaining m variables, provided that the column vectors associated with the m variables form a linearly independent set of vectors. The m variables are called basic variables, and the remaining $n - m$

variables that were set equal to zero are called nonbasic variables. The vectors associated with the basic variables form an $(m \times m)$ basis matrix $\boldsymbol{B}$.

See

► Linear Programming

Basic Variables

The set of variables corresponding to the columns of a basis matrix in a linear system $\boldsymbol{Ax} = \boldsymbol{b}$.

See

► Basic Solution
► Basis
► Linear Programming

Basis

A nonsingular square matrix $\boldsymbol{B}$ obtained by selecting linearly independent columns of a full row rank matrix $\boldsymbol{A}$. The matrix $\boldsymbol{B}$ is then a basis matrix for the system $\boldsymbol{Ax} = \boldsymbol{b}$. The components of $\boldsymbol{x}$ associated with $\boldsymbol{B}$ are called the basic variables, and the remaining components are called the nonbasic variables. The term basis also refers to the set of indices of the basic variables.

See

► Basic Variables
► Linear Programming

Basis Inverse

The inverse of a basis matrix.

See

► Basis
► Linear Programming

Basis Vector

A column of a basis matrix.

See

► Basis
► Linear Programming

Batch Shops

► Production Management

Battle Modeling

Dean S. Hartley III
Oak Ridge National Laboratory, Oak Ridge, TN, USA

Introduction

The ideal battle model completely, accurately, quickly, and easily predicts the results of any postulated battle from the initial conditions. Several factors prevent the existence of an ideal battle model.

One factor is computational complexity. For example, medical planners could use such a battle model to determine the size of treatment facilities, the breakdown of physician skills needed, and the medical supply inventory requirements. It is reasonable to suppose a battle model would track individuals and their separate wounds for engagements of a dozen participants on a side; however, maintaining that level of detail for engagements of tens of thousands of people would be prohibitively expensive in time and hardware requirements. Thus the requirement for complete predictions competes with the requirements for generality and speed of computation.

The second factor preventing the existence of an ideal battle model is the fact that not enough is known about battle dynamics to model it accurately. Where components can be modeled accurately (e.g., firing disciplines for weapons and probabilities of kills given hits), it is not known how the components fit together (e.g., when do soldiers fire their weapons and

how do conditions modify their ideal performance). Further, it is not known when, where, and why battles are joined or when and how they stop. The ignorance is not absolute, but is relative to the desired accuracy for the battle models.

A third factor also proceeds from ignorance. It is not known which initial conditions are significant for determining battle results. In general, those battle models that deliver massive details about the model results require extremely large quantities of input data. Thus, perceived accuracy of results is a competitor of ease and rapidity of use.

Battle Model Classification

Although the ideal battle model cannot be built, many individual battle models can be built, each conceived to fulfill a particular set of objectives. These models of combat may be classified by their position along several dimensions; however, they all have one feature in common, and that is the object that is modeled is some aspect of combat. These dimensions are listed below with illustrative examples of positions along the dimension.

DOMAIN	Land; air; naval; space; combinations.
SPAN (size of conflict)	Platoon battle; division combat; theater-level combat; global combat.
SCOPE (type of conflict)	Politico-military; special operations; low intensity conflict; urban warfare; conventional warfare; theater-level nuclear, chemical, and biological conflict; strategic nuclear conflict.
SCORING (adjudication topics and methodology)	Measures of merit: attrition, movement, tons of bombs dropped, supplies delivered, victory; methodologies: weapon weights (simple or complex, as in anti-potential potential, which uses eigenvalues to value weapon by the value of the weapons it can kill), process simulations.
RANDOMNESS	Deterministic or stochastic calculations.
COMBAT ACTIVITIES AND FORCE COMPOSITION (military assets and mission areas)	Small-arms; armor; aircraft; artillery; engineer; logistics; signal; command and control; intelligence; surface navy; submarine; electronic warfare; space assets; missiles.
LEVEL OF RESOLUTION OR DETAIL (smallest item modeled as a separate entity)	Bullet; soldier; tank; platoon; company; battalion; brigade; division; corps.
ENVIRONMENT	One-dimensional terrain (pistonmodel); two-dimensional terrain (including ocean or air), latitude-longitude or hexagonal grid-based; three-dimensional terrain; weather; day-night; smoke.
PURPOSE (design purpose or users' purpose)	Training; weapon system employment; force composition decisions; operations plans testing.
LEVEL OF TRAINING (training audience)	Individual skills; platoon leaders' skills; division staff skills; commanders' skills; combinations.
MODEL TREATMENT OF TIME	Linear code with no time representation or algorithmically computed time (generally analytic combat models); time-stepped simulations; event-driven simulations; expected value models; stochastic simulations.
HUMAN INTERACTION	Data preparation and output interpretation; interruptible with modification and restart; computer-assisted human participation on one or more sides; continuous human participation on all sides.
SIDEDNESS	One-sided (e.g., strategic nuclear strike damage effects); two-sided; multi-sided; hard-coded identical properties for each side, hard-coded different properties for each side (e.g., U.S. vs Soviet-style tactics), or data-driven properties for each side.
COMPUTER INVOLVEMENT	None; moderate; complete.
SIZE COMPUTER REQUIRED	PC; mini-computer; mainframe; supercomputer; peripheral equipment required; large run-times, small run-times.
EXTERNAL INTERACTIONS (interfaces with parts of the real world)	None; distributed processing; interfaces with weapon simulators; interfaces with real equipment; sand tables, scripting.

Battle modeling started the first time someone scratched a battle plan in the dirt and tried to conceive of the consequences. Sand tables, with miniature troops and landscaping, added discipline to the modeling process; however, the modeling remained essentially qualitative. Sand table models were used as war games, in which opposing players took turns moving the pieces and used rules to

adjudicate the results of the moves. Modern war games include sand table games and computer adjudicated games.

Attrition Laws

Lanchester (1916) introduced the concept of a quantitative model of attrition. (Osipov in Russia and Fiske in the U.S. introduced similar concepts at virtually the same time; however, most Western works refer to Lanchester's Laws and Lanchestrian attrition.) Lanchester showed that one could express the value of concentration of forces precisely, using mathematics, and thus evaluate what forces would be needed for victory before a battle. Engel (1954) provided what many took to be proof that Lanchester's square law was correct.

Lanchester's simple concepts have been elaborated to the extent that Taylor (1983) required two volumes to discuss the many uses and implications of Lanchester theory. The computational power of computers has permitted this elaboration. First, heterogeneous Lanchester equations could be solved without undue manual labor. Once, heterogeneous equations were admitted, the coefficients could be represented as functions of other factors, such as weather, firing discipline, and distance to the target. Bonder and Farrel (in Taylor 1983) introduced rigorous thinking into this area by observing direct fire activities and creating a mathematical model of those activities.

Dupuy (1985) argued that there are many important factors in combat that were not being included in the physics-based combat models. Morale, training, and leadership are at least as important as force sizes according to Dupuy. He proposed a model based on quantified judgments of these and other "soft" factors. His Quantified Judgment Model (QJM) stirred considerable controversy. Regardless of the merits of the QJM itself, the quantified judgments of soft factors is currently receiving more favorable reviews. The difference in public opinion at home during the Vietnam and Gulf wars and the impact on troop morale and the outcomes of the wars provides some justification for increased emphasis on soft factors.

Computers also made the computation of stochastic processes possible. The differential equations of Lanchester attrition were viewed as approximations to a random process model of the actual killing process that should be correct for large numbers. Stochastic duels addressed the results for small numbers. Ancker and Gafarian have made significant contributions in this area (Ancker 1994).

Helmbold has made contributions to both the theoretical and the practical aspects of battle modeling. His empirical studies of attrition (1961, 1964), breakpoints (1971), and movement (1990) injected the element of reality into the sometimes rarefied atmosphere of theoretical battle modeling. Hartley (1991) continued in this vein with results indicating that the best description of attrition (using a homogeneous approximation) is not the Lanchester square or linear law, but an intermediate form between the linear law and a logarithmic law. Speight (1995, 1997) and Speight and Rowland (1999) have continued the process, introducing duels (mini-battles) and simulations of combat exercises (trials) and showing the impact of firing on dead targets on the formulation of attrition equations.

With computer battle models also came a proliferation of structural types of models. Battle models involving anti-submarine warfare have a peculiar requirement of finding the enemy before the battle can be prosecuted. Search theory must be implemented in such models, just as it is used in actual battles or exercises (Shudde 1971). In some types of war, the proper allocation of resources or mix of strategies provides an easily defined variable (e.g., strategic nuclear targeting or allocation of combat air forces to mission types). Because game theory deals with optimal strategies considering both sides' options, it provided an obvious technique for addressing the problem and providing prescriptive models (Bracken et al. 1974; HQ USAF/SAMA 1974).

Dimension, Data, and Output

In earlier times, land warfare models were one-dimensional: the forward edge of the battle area (FEBA) advanced or retreated. More sophisticated versions allowed one-dimensional structures for each sector (piston-models). More powerful computers now permit two-dimensional representations of the battlefield, using either x, y (or latitude, longitude) coordinates or (rectangular or hexagonal) grid structures. Some models are now three-dimensional, having terrain elevation and playing the effects of

flying aircraft at different altitudes. [See, for example, the Research Evaluation and Systems Analysis (RESA) model (Naval Ocean Systems 1992), which plays aircraft at different altitudes and submarines at different depths].

Most large models have extremely large input and output data sets and require sophisticated database management systems to keep track of the data. These large output data sets also stress the human ability to understand the results. Sophisticated graphics are necessary adjuncts to most large models today. The graphics are required to define realistic scenarios and to understand the process and results of the model.

Advances in computer power have resulted in the capability for human interfaces that are qualitatively different from past capabilities. Such interfaces include real-time depictions of a battlefield from a human perspective and auditory and tactile interfaces. The first full-scale example of this kind of interface, called virtual reality, in a battle model was SIMNET (HQ US Army Armor School 1987). SIMNET is a network of tank and other vehicle simulators, each participating in a shared virtual battlefield. Work is proceeding to tie virtual reality battle models to other, more conventional battle models. The success of connecting simulators has motivated recent work in connecting interactive training models. The connection of these battle models permits distributed processing and cost sharing among users.

The history of battle modeling has not been a smooth process of constant improvements. It has been beset with controversies in many areas. Some of the controversies have involved the standard resource allocation question: where do you spend the money? One of the first of these concerned documentation. Early (1960–1970s) computer models were usually undocumented and, because of frequent modifications, had virtually indecipherable code. The need for proper documentation was obvious but the need for better (or at least more complex) models appeared overriding. While the readability of the documentation of today's models may be variable, most models are documented.

Verification, Validation, and Accreditation

One controversy probably began with the first model that produced a result someone did not like: is the model right? During the 1960s and early 1970s, it was said there were two kinds of generals: those for whom computer printout was the gospel and those who would believe nothing produced by a computer. The problem in dealing with the first type was in conveying that there were caveats. All results had to be retyped manually to disguise their origin for the second type of general. Today's generals (and politicians) grew up with computers. They want to understand to what extent the results are believable. They require verification, validation, and accreditation. Although progress is being made, no one knows how to completely verify, validate, or accredit the general battle model.

Other Controversies

There have also been technical controversies in battle modeling. Notable controversies have included the proper interpretation (and thus use) of the differences between the Lanchester linear and square laws, the connection between attrition and advance rates (if any), the value of force ratios, the connection between deterministic Lanchester formulations and stochastic attrition formulations and which should be used. There is a precept that states that a force ratio of 3-1, attacker-defender is required for a successful attack. Numerous studies have criticized this precept, yet it is still heard.

There are disagreements about the proper level of detail in deterministic models, despite agreement on the principle that what is appropriate depends on the uses to be made of a model. High resolution models of large span require tremendous quantities of data and run slowly. One camp advocates small, fast "roughly right" models as better than high resolution models. Another camp protests that such models will miss the critical points that differentiate the issues in question. The stochastic process camp protests that both the large, high resolution and the small, low resolution models are not grounded in the reality of stochastic battles, and cannot thus be even roughly right.

There have also been disagreements about the proper uses of models. At one time prescriptive battle models were popular (finding optimal strategies, where the definition of optimal varied with the model). Lately they have been out of favor. Complaints about the misuse of models have ranged from the use of models designed for other purposes and failing to understand the resulting mismatch of assumptions to charges of advocacy modeling. Advocacy modeling, in the

pejorative sense, entails fiddling with input parameters until a combination is found that gives the desired result. Most large models have sufficient numbers of parameters with sufficiently tenuous connections to physical factors that plausible values can be found that generate almost any result.

One controversy involves the discovery that very simple deterministic battle models can exhibit chaos (Dewar et al. 1991). The question of the impact of chaos on the more complex models that are actually used is obvious. Most issues are settled by point estimates. For example, suppose the impact of weapon X is being investigated. Model runs with 25% X, 50% X, 75% X, and 100% X are executed. The runs with 75% X and 100% X are found to have superior results. It is assumed that such results are valid for values between 75% and 100%. If the results are chaos driven, such an assumption is unwarranted. The question has not been finally answered; however, investigations with one of the currently used complex models indicates that any uncertainty due to chaotic behavior in that model is no larger than a few percent. Because this is within the uncertainty that was already present in the model, the impact of possible chaotic behavior was claimed to be minimal (Herndon 1993).

Concluding Remarks

Despite all controversy, battle modeling remains the only method of answering some questions and is widely used. Battle models are used to inform decisions on weapons' procurement issues (balancing costs against effectiveness), to test strategies and tactics, and to train personnel. Battle training models provide inexpensive tools for training commanders because the large numbers of combat personnel maneuver in the computer rather than on the ground. As military funding is reduced, this supplement to traditional training methods has become indispensable. New models continue to be created as the requirements for greater scope arise. The insertion of information technology into combat has necessitated new models that can discriminate among the effects of different Command, Control, Communications and Intelligence (C^3 I) systems, such as the Joint Warfare System (JWARS) for analysis and the Joint Simulation System (JSIMS) for training.

See

- ▶ Cost Analysis
- ▶ Cost-Effectiveness Analysis
- ▶ Documentation
- ▶ Game Theory
- ▶ Gaming
- ▶ Lanchester's Equations
- ▶ Military Operations Research
- ▶ Model Accreditation
- ▶ Operations Research Office and Research Analysis Corporation
- ▶ RAND Corporation
- ▶ Search Theory
- ▶ Validation
- ▶ Verification

References

Ancker, C. J., Jr. (1994). *An axiom set (laws) for a theory of combat* (Technical Report). Los Angeles: Systems Engineering, University of Southern California.

Bracken, J., Falk, J. E., & Miercort, F. A. (1974). *A strategic weapons exchange allocation model, Serial T-325. School of Engineering and Applied Science*. Washington, DC: The George Washington University.

Dewar, J. A., Gillogly, J. J., & Junessa, M. L. (1991). *Non-monotonicity, chaos and combat models, R-3995-RC*. Santa Monica, CA: RAND.

Dupuy, T. N. (1985). *Numbers, predictions & war*. Fairfax: Hero Books.

Engel, J. H. (1954). A verification of Lanchester's law. *Operations Research, 2*, 163–171.

Hartley, D. S., III. (1991). *Predicting combat effects, K/DSRD-412*. Oak Ridge, TN: Martin Marietta Energy Systems.

Helmbold, R. L. (1961). *Historical data and Lanchester's theory of combat, AD 480 975, CORG-SP-128*. Fort Belvoir, VA: Combat Operations Research Group.

Helmbold, R. L. (1964). *Historical data and Lanchester's theory of combat, Part II, AD 480 109, CORG-SP-190*. Fort Belvoir, VA: Combat Operations Research Group.

Helmbold, R. L. (1971). *Decision in nattle: Breakpoint hypotheses and engagement termination data, AD 729 769*. Alexandria, VA: Defense Technical Information Center.

Helmbold, R. L. (1990). *Rates of advance in historical land combat operations, CAA-RP-90-1*. Bethesda, MD: Concepts Analysis Agency.

Herndon, S. K. (1993). *TRADOC analysis command research on VIC variability*, (Technical Document TRAC-TD-0293). Kansas: TRADOC Analysis Command, Fort Leavenworth.

HQ USAF/SAMA (1974). *A computer program for measuring the effectiveness of tactical fighter forces* (Documentation and Users Manual for TAC CONTENDER) SABER GRAND (CHARLIE).

Hq, U. S. A., & School, A. (1987). *M-1 SIMNET operator's guide*. Kentucky: Fort Knox.

Lanchester, F. W. (1916). Mathematics in warfare. In *Aircraft in warfare: The dawn of the fourth arm, constable and company*, London. (Reprinted in *The World of Mathematics*, by J. R. Newman, Ed., 1956, New York: Simon and Schuster).
Naval Ocean Systems Center (1992). *RESA Users Guide Version 5.5*, Vols. 1–8.
Shudde, R. H. (1971). Contact and attack problems. In P. W. Zehna (Ed.), *Selected methods and models in military operations research* (pp. 125–146). Alexandria, VA: Military Operations Research Society.
Speight, L. R. (1995). Modelling the mobile land battle: The Lanchester frame of reference and some key issues at the tactical level. *Military Operations Research, 1*(3), 53–56.
Speight, L. R. (1997). Modelling the mobile land battle: Lanchester's equations, mini-battle formation and the acquisition of targets. *Military Operations Research, 3*(5), 35–62.
Speight, L. R., & Rowland, D. (1999). Modelling the mobile land battle: Combat degradation and criteria for defeat. *Military Operations Research, 4*(3), 45–62.
Speight, L. R., & Rowland, D. (2010). Modelling the rural infantry battle: Group morale and the chances of attack success. *Military Operations Research, 15*(1), 31–52.
Taylor, J. G. (1980). *Force-on-force attrition modeling*. Linthicum, MD: Military Applications Society, INFORMS.
Taylor, J. G. (1983). *Lanchester models of warfare* (Vol. I and II). Linthicum, MD: Military Applications Society, INFORMS.
Taylor, B., & Lane, A. (2004). Development of a novel family of military campaign simulation models. *Journal of the Operational Research Society, 55*, 333–339.

Bayes Rule

When a decision maker receives data bearing on an uncertain event, the probability of the event can be updated by computing the conditional probability of the uncertain hypothesis given the new evidence. The derivation of the revised or a posteriori probability can be easily derived from fundamental principles and its discovery has been attributed to the Reverend Thomas Bayes (1763). The result is therefore known as Bayes rule or theorem:

$$\Pr\{H_1|E\} = \frac{\Pr\{E|H_1\}\Pr\{H_1\}}{\sum_i \Pr\{E|H_i\}\Pr\{H_i\}}$$

In this equation, H_1 refers to the specific, uncertain hypothesis entertained by the decision maker, the $\{H_i\}$ are the complete set of possible hypotheses, and E refers to the new evidence or information received.

See

► Bayesian Decision Theory, Subjective Probability, and Utility

Bayesian Decision Theory, Subjective Probability, and Utility

Kathryn Blackmond Laskey
George Mason University, Fairfax, VA, USA

Introduction

In every field of human endeavor, individuals and organizations make decisions under conditions of uncertainty and ignorance. The consequences of a decision and their value to the decision maker often depend on events or quantities which are unknown to the decision maker at the time the choice must be made. Such problems of decision under uncertainty form the subject matter of Bayesian decision theory. Bayesian decision theory has been applied to problems in a broad variety of fields, including engineering, economics, business, public policy, and artificial intelligence.

A decision-theoretic model for a problem of decision under uncertainty contains the following basic elements:

- A set of options from which the decision maker may choose;
- A set of consequences that may occur as a result of the decision;
- A probability distribution that quantifies the decision maker's beliefs about the consequences that may occur if each of the options is chosen; and
- A utility function that quantifies the decision maker's preferences among different consequences.

Subjective Probability

Decision theory applies the probability calculus to quantify a decision maker's beliefs about uncertain events or quantities, and to update beliefs upon receipt of additional information. De Finetti (1974) showed that any decision maker who acts on degrees of beliefs not

conforming to the probability calculus can be exploited by a series of gambles guaranteed to result in a net loss. Such a bet is called a dutch book. The Dutch Book Theorem and other related derivations of probability from axioms of rationality have been used to justify probability as a calculus of rational degrees of belief (De Groot 1970; Pratt et al. 1965).

Bayes Rule

When a decision maker receives information bearing on an uncertain hypothesis, degrees of belief are updated by computing the conditional probability of the uncertain hypothesis given the new evidence. The equation expressing how beliefs change with new evidence has been attributed to the Reverend Thomas Bayes (1763) and is known as Bayes Rule. The odds-likelihood form of Bayes Rule is:

$$\frac{\Pr\{H_1|E\}}{\Pr\{H_2|E\}} = \frac{\Pr\{E|H_1\}\Pr\{H_1\}}{\Pr\{E|H_2\}\Pr\{H_2\}}$$

In this equation, H_1 and H_2 refer to two uncertain hypotheses entertained by the decision maker and E refers to the new evidence or information received by the decision maker. Bayes rule quantifies how evidence is used to obtain the relative posterior probabilities $\Pr\{H_i|E\}$ of the hypotheses given the evidence. The ratio of posterior probabilities is determined by two factors. One is the ratio of prior probabilities $\Pr\{H_i\}$: all other things being equal, the stronger the prior belief in H_1 relative to H_2, the stronger the posterior belief in H_1 relative to H_2. The other is the likelihood ratio, or ratio of the probabilities $\Pr\{E|H_i\}$ of the evidence given each of the hypotheses. Again, all other things being equal, the better H_1 accounts for the evidence relative to H_2, the stronger the posterior belief in H_1 relative to H_2.

Other Interpretations of The Probability Calculus

There has been considerable debate about how to interpret the concept of probability. The term Bayesian, after Bayes Rule, is used to refer to the subjective interpretation. A subjective probability distribution represents an individual's degrees of belief about the likelihood of uncertain outcomes. Alternative interpretations of probability include the classical, the logical, and the frequentist approaches (Fine 1973). Much of standard statistical theory is based on the frequentist approach. Frequentists argue that probability models are appropriate only for repeatable phenomena exhibiting inherent randomness. For such phenomena, it is argued, there exist objectively correct probabilities intrinsic to the process producing the uncertain outcomes. Subjectivists apply probability theory to any outcomes about which a decision maker is uncertain. For subjectivists, no objectively correct probabilities need exist. Different decision makers are free to have different opinions about the probability of an outcome.

The only constraint subjective theory places on a probability distribution is that it be coherent, that is, that degrees of belief conform to the probability calculus. Within this constraint, decision makers are free to choose any probability distribution to model their uncertainty about a problem. Its inherent subjectivity has been a persistent criticism of the subjectivist approach. This is often of little practical consequence for problems that can be said to exhibit inherent randomness. The subjectivist draws inferences about the posterior distribution of the unknown parameter, while the frequentist draws inferences about the distribution of the data given different values of the unknown parameter. Nevertheless, it can be shown that when there are sufficient data to draw accurate inferences, the subjectivist and the frequentist will usually agree on the implications of the results. Thus, the major difference of practical import between the subjectivist and the frequentist is their attitudes toward problems for which there are too little data to estimate parameters accurately or for which the assumption of intrinsic objective frequencies is problematic. The frequentist maintains that probability models are in-appropriate for such problems; the subjectivist argues that probabilities are appropriate and that it is legitimate for rational people to disagree until there are sufficient data to bring them to agreement.

Utility Theory

Decision theory quantifies preferences by a utility function. It is assumed that the decision maker can

assign a numerical utility to each possible consequence of each option being entertained. Consequences with higher utilities are preferred to consequences with lower utilities. When there is uncertainty, the decision maker selects the option for which the expected value of the utility function is the largest. For some problems it is customary to deal with losses, or negative utilities. Smaller losses are preferred to larger losses.

The concept of utility appears to have been first introduced by Daniel Bernoulli (1738) in his solution to a puzzle known as the St. Petersburg Paradox. Bernoulli considered the problem of what price to pay for the opportunity to play the following gamble. A fair coin (probability 0.5 of landing heads) is tossed repeatedly until the first head appears. If the first head appears on the *n*th toss, the decision maker receives a prize of 2^n units of currency. The decision maker's expected monetary prize is

$$2(0.5) + 2^2(0.5)^2 + 2^3(0.5)^3 + \ldots,$$

which is infinite. A decision maker who maximized expected monetary value should be prepared to pay an arbitrarily large sum of money for the opportunity to play this gamble. As Bernoulli noted, most people would be willing to pay only a modest amount. Bernoulli suggested that the resolution to this apparent paradox was that a prize's worth to a decision maker was a nonlinear function of the monetary value of the prize. For example, replacing 2^n with $\log 2^n$ in the above equation yields a finite expected monetary prize.

Von Neumann and Morgenstern (1944) were the first to present a formal axiomatic development of utility theory. They defined the utility of a consequence in terms of a comparison between two options, one sure and one uncertain. The sure option is the consequence itself; the uncertain option is a lottery between two standard reference prizes, one worth more and one worth less than the consequence in question. If the reference prizes are assigned utility one and zero, then the utility of the consequence in question is defined as the probability at which the decision maker is indifferent between the two lotteries. Several similar axiom systems can be shown to lead to the maximization of expected utility as a principle of rational decision making (De Groot 1970; Pratt et al. 1965).

Concluding Remarks

It has been observed that people systematically violate the axioms of expected utility theory in their everyday behavior. Some of these violations can be reversed by informing people of the implications of their stated preferences. In other cases, many people resist changes to their original judgments. Even when the decision maker regards expected utility theory as a norm of rational behavior, it cannot be assumed that unaided judgments will be consistent with the theory. The field of decision analysis applies theories and methods from decision theory and the psychology of human information processing to construct decision theoretic models for practical decision problems (Clemen 1996).

Interest has been growing in decision theoretic formulations of statistical problems. For example, to formulate an hypothesis testing problem, one defines a prior probability for the null and alternative hypotheses. One also defines losses associated with accepting a false alternative hypothesis and rejecting a true null hypothesis. The optimal decision rule is to accept or reject the hypothesis according to which decision yields the lower posterior expected loss given the observed sample. Similarly, decisions of whether to gather information and how large a sample to draw can be formulated as decision problems that consider both the cost of gathering information and the benefit of obtaining the information. Some problems that are quite complex when viewed from a frequentist perspective become straightforward when viewed from a Bayesian perspective. Examples include hierarchical models and problems of missing data (Gelman et al. 1995).

An area of application is the field of intelligent systems (Haddawy 1999). Utility theory is being applied to planning and control of reasoning in expert systems. Diagnostic expert systems based on probability theory have achieved performance comparable to human decision makers (e.g., the Pathfinder system for diagnosing lymph node pathology, Heckerman 1991). Perhaps the most important and challenging aspect of decision analysis is the creative process of model formulation. Decision theory takes options, consequences, and their interrelationships as given. Automated decision model generation is an open research area of great importance to application of decision theory to the field of intelligent systems (Haddawy 1994).

See

- ▶ Decision Analysis
- ▶ Decision Problem
- ▶ Decision Trees
- ▶ Expert Systems
- ▶ Utility Theory

References

Bayes, T. R. (1763). An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London, 53*, 370–418 (Reprinted with biographical note by G. Barnard, 1958, in *Biometrika*, *45*, 293–315).
Bernoulli, D. (1738). Specimen Theoriae Novae de Mensura Sortis. *Commentarii Academiae Scientarium Imperalis Petropolitanae* 175–192 (Translated in L. Sommer, 1984, *Econometrica*, *22*, 23–26).
Clemen, R. (1996). *Making hard decisions*. Belmont, CA: Duxbury Press.
de Finetti, B. (1974). *Theory of probability: A critical introductory treatment*. New York: Wiley.
De Groot, M. H. (1970). *Optimal statistical decisions*. New York: McGraw Hill.
Fine, T. L. (1973). *Theories of probability*. New York: Academic.
Gelman, A., Carlin, J., Stern, H., & Rubin, D. (1995). *Bayesian data analysis*. London: Chapman and Hall.
Haddawy, P. (1994). Generating Bayesian networks from probabilistic knowledge bases. In *Proceedings of tenth conference on uncertainty in artificial intelligence* (pp. 262–299). San Mateo, CA: Morgan Kaufmann.
Haddawy, P. (1999). An overview of some recent developments in Bayesian problem solving. *AI Magazine, 20*(2), 11–19.
Heckerman, D. (1991). *Probabilistic similarity networks*. Ph.D. dissertation, Program in Medical Information Sciences. Stanford University, CA.
Pratt, J. W., Raiffa, H., & Schlaifer, R. (1965). *The foundations of decision under uncertainty: An elementary exposition*. New York: McGraw Hill.
von Neumann, J., & Morgenstern, O. (1944). *Theory of games and economic behavior*. New Jersey: Princeton University Press.

Beale Tableau

A modification of the simplex tableau arranged in an equation form such that the basic variables and the objective function value are expressed explicitly as functions of the nonbasic variables. This tableau is often used when solving integer-programming problems.

See

- ▶ Linear Programming
- ▶ Tucker Tableau

Bellman Equation

- ▶ Bellman Optimality Equation

Bellman Optimality Equation

Dynamic programming equation that the optimal value (or cost-to-go) function must satisfy, according to the principle of optimality. One simple form is the following finite-action, finite-state, finite-horizon version for a minimization problem:

$$f_n(i) = \min_a \{c_n(i,a) + \Sigma_j p_{ij}(a) f_{n+1}(j)\},$$

where $f_n(i)$ represents the optimal cost-to-go function in state i for stage (period) n, $c_n(i,a)$ is the one-period cost in stage n for state i and action a, and $p_{ij}(a)$ is the probability of transitioning from state i to state j when action a is taken.

See

- ▶ Approximate Dynamic Programming
- ▶ Dynamic Programming
- ▶ Markov Decision Processes

Benders Decomposition Method

A procedure for solving integer-programming problems that have a few integer variables. These so-called complicating variables, when given specific values, enables the resulting problem to be readily solved as a linear-programming problem.

See

- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming

Best-Fit Decreasing Algorithm

▶ Bin-Packing

Bidding Models

▶ Auction and Bidding Models

Big M Method

A method to drive artificial variables out of the basis in the simplex algorithm, by imposing a sufficiently large, finite penalty M for using these variables.

See

▶ Artificial Variables
▶ Linear Programming
▶ Phase I Procedure
▶ Phase II Procedure
▶ Simplex Method (Algorithm)

Bilevel Linear Programming

Bilevel linear programming (BLP) is a hierarchical, decentralized, multilevel mathematical programming problem in which the objective functions and constraints are linear. It can be stated in terms of upper and lower problems as follows:

$$\underset{x}{\text{Maximize}} f_1(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{c}_1\boldsymbol{x} + \boldsymbol{d}_1\boldsymbol{y}$$

where $\boldsymbol{y}$ solves:

$$\underset{y}{\text{Maximize}} f_2(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{c}_2\boldsymbol{x} + \boldsymbol{d}_2\boldsymbol{y}$$

subject to

$$\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} \leq \boldsymbol{b}$$
$$\boldsymbol{x}, \boldsymbol{y} \geq 0$$

where $\boldsymbol{c_1}, \boldsymbol{c_2}, \boldsymbol{d_1}, \boldsymbol{d_2}$, and $\boldsymbol{b}$ are constant vectors, $\boldsymbol{A}$ and $\boldsymbol{B}$ are constant matrices; $\boldsymbol{x}$ and $\boldsymbol{y}$ are vectors of the decision variables of the upper and lower problems, respectively; f_1 and f_2 are the objective functions of the upper and lower problems, respectively.

See

▶ Linear Programming

References

Bard, J. F. (1984). Optimality conditions for the bilevel programming problem. *Naval Research Logistics Quarterly, 31*, 13–16.
Ben-Ayed, O. (1993). Bilevel linear programming. *Computers & Operations Research, 20*, 485–501.
Bialas, W. F., & Karwan, M. H. (1984). Two-level linear programming. *Management Science, 30*, 1004–1020.
Colson, B., Marcotte, P., & Savard, G. (2005). Bilevel programming: A survey. *4OR, 3*, 87–107.

Binary Variable

A variable that is restricted to be equal to 0 or 1. Binary variables are often used to handle logical, nonlinear conditions associated with a problem whose constraining conditions are linear.

See

▶ Integer and Combinatorial Optimization
▶ Integer-Programming Problem

Bin-Packing

Nastaran Coleman[1] and Pearl Wang[2]
[1]Federal Aviation Administration, Washington, DC, USA
[2]George Mason University, Fairfax, VA, USA

Introduction

The bin-packing problem is concerned with the determination of the minimum number of bins that are

needed to pack a given set of input data items. The problem has numerous applications in operations research, computer science, and engineering, where the items and bins to be packed can be of varying shapes or multi-dimensional in size. These applications include industrial manufacturing, container loading, stock cutting, vehicle routing, television commercial scheduling, job scheduling on multiple processors, file backup creation in removable media, integrated circuit manufacturing and fault detection, and location testing in linear circuits. Since the bin-packing problem is known to be NP-hard (Garey and Johnson 1979), it is of interest to find efficient heuristics that obtain near-optimal solutions to the problem.

Problem Definition

The classical one-dimensional bin-packing problem (1DBPP) is defined as follows: Given a positive bin capacity C and a list of items $L = (p_1, p_2, \ldots, p_n)$, where p_i has size $s(p_i)$ satisfying $0 \leq s(p_i) < C$, determine the smallest integer m such that there is a partition of $L = B_1 \cup B_2 \cup \ldots \cup B_m$ where the sum of the sizes of the items $p_i \in B_j$ do not exceed the capacity C. Each set B_j is usually viewed as the contents of a bin of capacity C. In much of the literature, C is taken to be 1.

Several versions of two-dimensional bin-packing problems have also been studied. For example, if L is a set of rectangles p_i having heights h_i and widths w_i, one type of bin packing problem requires that the rectangles of L be packed into a single two-dimensional bin of width C and infinite height. The goal is to determine a minimum height packing of the pieces into this bin. These problems are referred to as strip-packing problems.

For an alternative form of the two-dimensional packing problem, the rectangles of L are to be packed into a minimum number of rectangular bins. A common version of the problem concerns packing a list of squares into m unit squares with the objective being to minimize m. When the rectangles to be packed are not square, restrictions might be made on the types of allowable placements of the rectangles within the bins. Depending on the application, rotations of the items may not be permitted; packings may also require that the items are placed parallel to the sides of the bins.

The items being packed in two-dimensional problems do not need to be rectangular in shape. Circular and polygonal shapes may also be packed into circular or rectangular bins.

Three-dimensional bin-packing problems have goals that are similar to their lower dimensional counterparts. For example, given the set L of rectangular prisms having widths w_i, height h_i, and depth d_j, a common problem is to pack the items into a minimum number of bins of width W, height H, and depth D. In the case of container packing, the pieces are not rotated and must be placed parallel to the sides of the bins.

Cutting stock problems are variants of bin packing problems because the amount of wasted space within stock sheets is to be minimized while the pieces are being cut from stock sheets. Similarly, if just a single bin of fixed size is to be packed and each item is characterized by both a volume and a value, the problem of maximizing the total value of a subset of items that can fit into the bin by volume is known as the knapsack problem.

Approximation algorithms for bin-packing problems were among the earliest algorithms studied in the literature. In the 1970s, it was shown that near-optimal solutions could be guaranteed for some frequently used one-dimensional packing techniques. Since then, many heuristics have been proposed for obtaining approximate solutions to both the one and two-dimensional problems for sequential and parallel models of computation. Three-dimensional problems were initially studied to a lesser degree, but recent work now appears regularly in the literature. The performance of a given heuristic (i.e., the computational time and resources needed to find a packing), as well as the quality of the packing that is constructed by the heuristic are important considerations that have been analyzed by many researchers.

Surveys of many classical bin-packing algorithms can be found in Coffman et al. (1996). A bibliography of cutting and packing research was presented by Sweeney and Paternoster (1992), while a more recent typology that characterizes cutting and packing problems is described in Wäscher et al. (2007). Recent probabilistic analyses of approaches for solving one-dimensional bin-packing problems are discussed in Coffman et al. (2000). Two-dimensional packing problems are surveyed by Lodi et al. (2002)

and meta-heuristic algorithms for strip packing problems are reviewed in (Hopper and Turton 2001). Recent work that addresses three-dimensional packing problems includes (Martello et al. 2000), (Faroe et al. 2003), and (Parreño et al. 2008). Heuristic approaches for solving irregular and polygonal packing problems are presented by Jakobs (1996) and Burke et al. (2010).

Algorithms for solving the problem on various parallel models of computation can be found in Anderson et al. (1989), Fenrich et al. (1989), Berkey (1990), and Coleman and Wang (1992). The EURO Special Interest Group on Cutting and Packing maintains a website for research activities related to cutting and packing.

Characterizations of Bin-Packing Algorithms

Many types of bin-packing algorithms have been proposed and analyzed for both sequential and parallel systems. Sequential heuristics can be classified as either on-line or off-line algorithms. On-line algorithms assign data items to bins in the same order as originally input, without utilizing any global knowledge of the data list. For example, the Next-Fit packing and Sum of Squares heuristics are on-line algorithms that perform one-dimensional packing. Off-line algorithms preprocess the data, usually by sorting. Well-known examples are the First-Fit Decreasing and Best-Fit Decreasing algorithms. Alternatively, other methods may preprocess the input data by partitioning the items by size into subintervals, and then pack the data using those sub-intervals. These techniques are described in more detail below.

Approximation algorithms for solving the one-dimensional bin-packing problem on various models of parallel computation have been reported. It has been shown that several frequently used sequential bin packing strategies such as First-Fit Decreasing are P-Complete. Thus, it is unlikely that these heuristics can be parallelized into efficient algorithms for the theoretical Parallel Random Access Machine (PRAM) model of computation. However, other well-known sequential strategies such as Harmonic packing can be parallelized efficiently. In the previous decade, experimental studies of similar heuristics were performed on Single-Instruction, Multiple-Data (SIMD) and Multiple-Instruction, Multiple-Data (MIMD) parallel computers.

Theoretical Studies

Performance metrics have been formulated as a means to compare these different packing algorithms when executed on random data. Theoretical analyses typically include worst-case and average-case packing performance of the heuristics. The asymptotic worst-case performance can be defined as the limiting ratio of an algorithm's worst instant packing to its optimal packing. For example, if $A(L)$ and $OPT(L)$ are the number of bins packed by an algorithm A and the optimal number of bins needed for a list L, respectively, then the asymptotic performance ratio can be defined as

$$R_A^{\infty} = \inf\{r \geq 1 : \text{ for some } N > 0, A(L)/OPT(L) \leq r \text{ for all } L \text{ with } OPT(L) \geq N\}$$

Two measures of average-case packing performance that have been studied are the expected values $E(R_N)$ and $E(U)$ where R_N is the ratio of the average number of bins packed by the algorithm to the average size of all data items and U is the difference between these quantities. Further, an algorithm is often said to exhibit perfect packing if $E(R) = 1$, where $E(R)$ is the limiting distribution of $E(R_N)$, or when $E(U) = O(\sqrt{N})$.

These metrics are studied analytically as well as by simulation. The input data are usually assumed to come from a uniform distribution $U[a, b]$. Coffman et al. (2000) introduced the perfect packing theorem and show that the optimal expected wasted space for a random list is either $o(n)$, $o(n^{0.5})$ or $o(1)$. These researchers have also shown that the average case can differ substantially between discrete and continuous uniform distributions.

An alternative measure of packing performance is to determine the expected waste of the packing. If $L_n(F)$ denotes a list of n items drawn according to a probability distribution F and $P_n^A(F)$ denotes a packing resulting from the application of algorithm A, then the expected waste is defined as $EW_n^A(F) = E\left[W\left(P_n^A(F)\right)\right]$ where expectation is taken over the random variable $L_n(F)$.

Theoretical studies of bin packing problems are often aimed at determining whether asymptotic approximation schemes can be constructed. In this

case, researchers seek to determine if for every $\varepsilon > 0$, there is a polynomial time algorithm A_ε having an asymptotic approximation ratio of $1 + \varepsilon$.

Some One-Dimensional Packing Heuristics

The Next-Fit algorithm packs one-dimensional items into one-dimensional bins in the simplest fashion. The data items are processed one at a time, beginning with p_1, which is put into bin B_1. If item p_i is to be packed and B_j is the highest indexed nonempty bin, then p_i is placed into bin B_j if it fits into B_j; that is, $p_i + size(B_j) \leq C$. Otherwise, a new bin B_{j+1} is started and p_i is placed into it. In this manner, each successive piece is packed into the most recently used bin, and previously packed bins are not considered. Next-Fit is a fast on-line algorithm whose time complexity is $O(n)$. Its worst-case performance ratio is bounded by 2, and its average performance by 3/2. Variants of Next-Fit have been proposed and include Next-Fit-Decreasing, Next-1-Fit, and Next-K-fit. The basic approach is also used to obtain level-oriented heuristics for solving two-dimensional bin packing problems.

The Harmonic packing algorithm begins by partitioning the unit interval into the set of intervals $I_k = (1/(k + 1), 1/k]$, $1 \leq k < m$ and $I_m = (0, 1/m]$. The bins are divided into m categories and an I_k-bin packs at most kI_k data. The packing of each I_k piece into an I_k-bin is done using the Next-Fit Algorithm. At any given time, an active list of all unfilled I_k-bins is kept. The Harmonic algorithm has a worst-case performance bound of 1.69; some modified versions of the approach have been shown to have lower performance bounds.

The Sum-of-Squares (SS) algorithm is an online method for packing items with integral sizes into bins of capacity C. It has time complexity $O(nC)$. If the amount of unpacked space in a bin is called its gap, g, and $N(g)$ is the number of bins in a current packing with gap g, then this algorithm puts an item p_i into a bin such that after placing the item, the value of $\sum_{g=1}^{C-1} N(g)^2$ is minimized.

Theoretical analysis of this algorithm demonstrates that for any perfectly packable distribution F, that $EW_n^{SS}(F) = O(\sqrt{N})$ and if F is a discrete uniform distribution $U(j, C)$ where $j < C - 1$, then $EW_n^{SS}(F) = O(1)$. For all lists L, it is further demonstrated that $SS(L) < 3OPT(L)$. Csirik et al. (2006) survey other online algorithms including randomized variants of sum-of-squares. Bender et al. (2007) propose two variants of the sum-of-squares algorithm and Seiden (2002) presents a survey as well as an online algorithm based on the Harmonic approach.

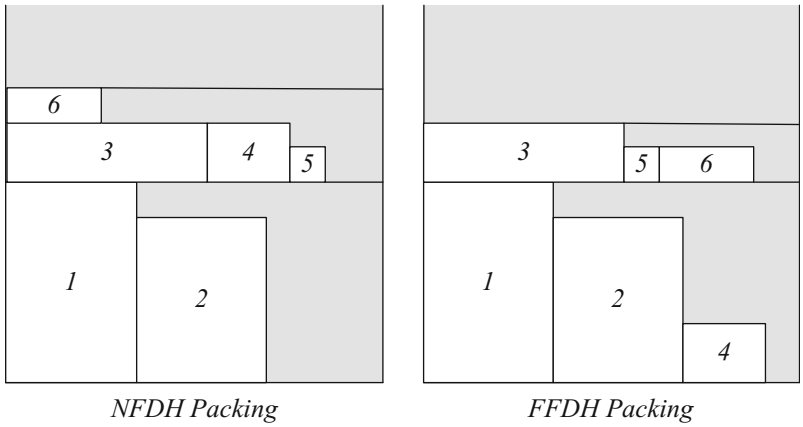
The First-Fit (FF) heuristic packs each successive data item p_i into the lowest indexed bin B_j into which it fits. When this is not possible, a new bin is created. Thus, it is necessary to maintain a list of all partially filled bins. For the worst-case, average case, and lower bound performance of First-Fit, it has been shown that the numberof bins used by this algorithm is 17/10 $OPT(L) \pm 2$, where OPT is the number of bins used by the optimal solution. Xia and Tan (2010) decreased the upper bound for the asymptotic performance ratio to 17/10 OPT + 7/10 for First-Fit and for the absolute performance ratio– to 12/7 OPT. The time complexity of First-Fit is $O(n \log n)$.

If the items are initially sorted in non-increasing order before packing proceeds, the heuristic is referred to as First-Fit Decreasing, and the performance bound decreases to 11/9 OPT + 6/9. Other algorithms that are based on this approach include Best-Fit (where the "best" bin is chosen if there is more than one possibility), Best-Fit Decreasing, Worst-Fit, Almost Worst-Fit, Revised First-Fit, and Modified First-Fit Decreasing bounded by 71/60 OPT + 1. When the data items are drawn from a uniform distribution, then $E(A(L)) - n/2 = O(n)$ for the First-Fit Decreasing and Best-Fit Decreasing algorithms. Asymptotic polynomial-time approximation schemes show that it is possible to find a solution for any $0 < \varepsilon \leq 1/2$ in polynomial time using at most $(1 + 2\varepsilon)OPT + 1$ bins.

Some Multi-Dimensional Packing Heuristics

Two-Dimensional Packing

The Two-Dimensional Bin-Packing Problem requires packing a finite set of small rectangles into the minimum number of rectangular bins without overlapping. The problem is strongly NP-hard, and has several industrial applications. Other variants of two-dimensional bin-packing problems occur in real-world applications, especially in the manufacturing industries. Additional constraints may include orientation where items can be rotated by 90°



Bin-Packing, Fig. 1 Level-oriented packings

or have to stay fixed. For example, rotation is not allowed when the items are articles to be paged in newspapers.

Researchers have applied one and two-phase algorithms that make use of upper and lower bounds on the number of bins needed to pack the input rectangles. These approaches are often integrated into greedy heuristics and tabu searches. One-phase algorithms directly pack the items into the finite bins. Two-phase algorithms start by packing the items into a single strip, i.e., a bin having width W and infinite height. In the second phase, the strip solution is used to construct a packing into finite bins. Lodi et al. (2002) survey advances obtained for the two-dimensional bin and strip packing packing problems, with emphasis on exact algorithms whose goal is to find an optimal solution, as well as effective heuristic and metaheuristic approaches.

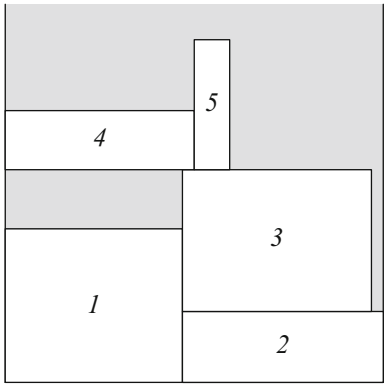
Level-oriented packing heuristics pack rectangles into a single two-dimensional bin (or strip) that has infinite height. In these approaches, the rectangles to be packed are first ordered by non-increasing height. The packing is constructed as a sequence of levels, whose heights are defined by the heights of the first rectangles placed in the respective levels. The Next-Fit or First-Fit approaches can be used to define and fill these levels of the bin. The asymptotic performance bounds of the Next-Fit Decreasing Height (NFDH) and First-Fit Decreasing Height (FFDH) heuristics are 2 and 1.7, respectively. Figure 1 illustrates these packing heuristics.

Similar approaches in which the heights of the levels are preset by a parameter yield a variety of shelf heuristics, where these levels can be packed in a similar fashion. Next-Fit Shelf and First-Fit Shelf are examples of these heuristics. Their corresponding execution times are $O(n)$ and $O(n^2)$. If the parameter that dictates the shelf heights is defined by r, then these methods have asymptotic performance bounds of $2/r$ and $1.7/r$, respectively.

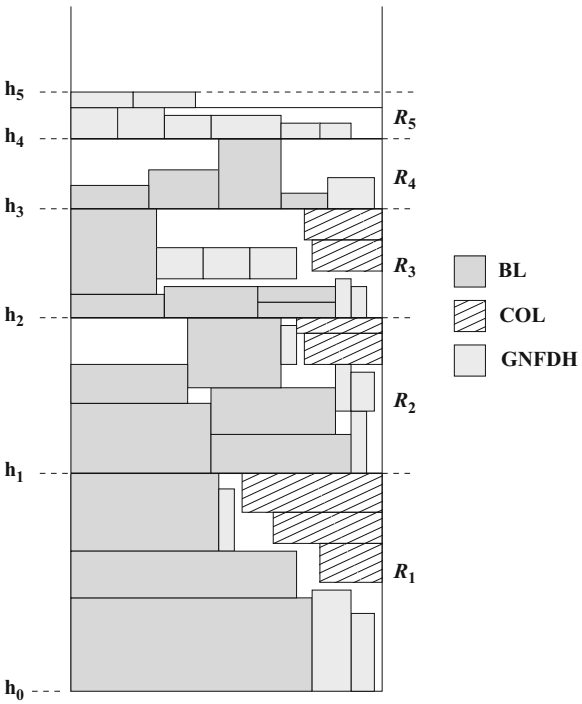
Bottom-Left (BL) packing approaches pack rectangles into an infinite height bin by successively placing each item into the bottom-most, left-most position in which it fits without overlapping any rectangles that have already been packed. If the items are preordered by non-increasing width, then the worst case bound of this heuristic indicates that the height of the packing does not exceed twice the height of an optimal packing. The algorithm can be implemented in $O(n^2)$ time and a sample packing is shown in Fig. 2.

Alternative methods may divide the set of items being packed into sublists that are used to obtain a split packing. In this case, the infinite height bin is also divided into subregions where one-dimensional heuristics are used to pack the rectangles. Classical techniques include Split-Fit, Mixed Fit, and Up-Down (see Fig. 3) which require $O(n \log n)$ time. Performance ratios of 2, 1.33, and 1.25, respectively, have been proven for these approaches. Other similar methods appear in the literature. Coffman and Shor (1993) discuss asymptotic average-case analysis for two-dimensional bin-packing.

One particular heuristic that uses a split packing approach addresses the problem of packing squares into a two-dimensional strip of unit width. The squares whose widths are greater than 1/2 are first

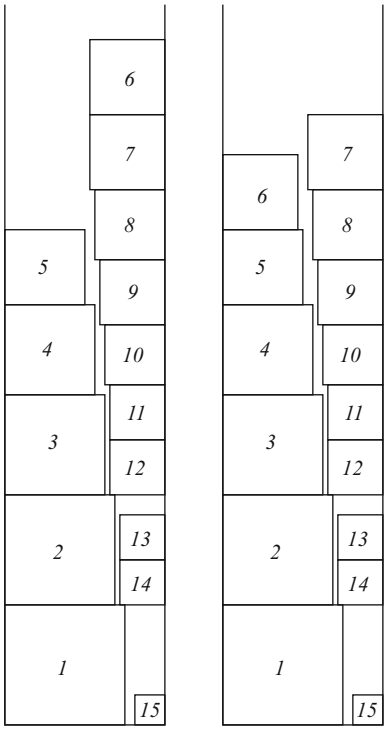


Bin-Packing, Fig. 2 BL packing



Bin-Packing, Fig. 3 Up-down packing

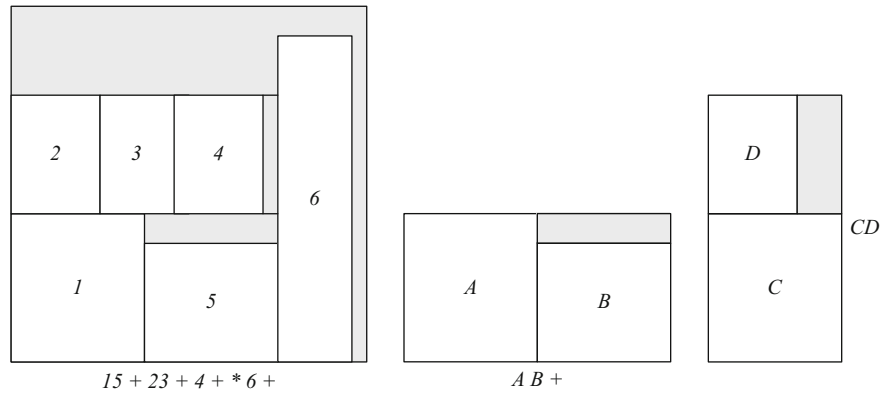
stacked along the left edge of the strip in order of decreasing width. Starting at the height, $H_{1/2} = \sum_{w_i > 1/2} h_i$, where the sum of the sizes of packed squares exceeds 1/2, the remaining squares are stacked along the right edge of the strip in order of decreasing width. This stack slides downward until



Bin-Packing, Fig. 4 Packing squares

it either rests on the bottom of the strip, or a square in the right stack comes in contact with a square in the left stack, whichever occurs first. Finally, all the squares lying entirely above $H_{1/2}$ are repacked into two stacks, one against the left edge of the strip and the other against the right edge. This is done in decreasing order of size, placing each successive square on the shorter of the two stacks already created. A sample packing is shown in Fig. 4. It can be shown for this algorithm, that $E(A(L)) = E(OPT(L)) + O(1)$.

When multi-dimensional objects are to be packed into a minimum number of multidimensional bins, the vector packing approach can be used. This technique is a direct generalization of the one-dimensional problem. For example, if rectangles are to be packed into square bins, then the only types of packing that are permitted are those where the rectangles are diagonally placed corner-to-corner across the bins. In general, if a vector packing algorithm is such that no two nonempty bins can be combined into a single bin, then the ratio of the number of bins packed to the optimal solution does not exceed $d + 1$, where d is the number of dimensions. Extensions of the First-Fit and First-Fit Decreasing



Bin-Packing, Fig. 5 Postfix GA encoding

heuristics to this multi-dimensional case have yielded approaches whose asymptotic worst case ratio is $d + 7/10$ and $d + 1/3$, respectively.

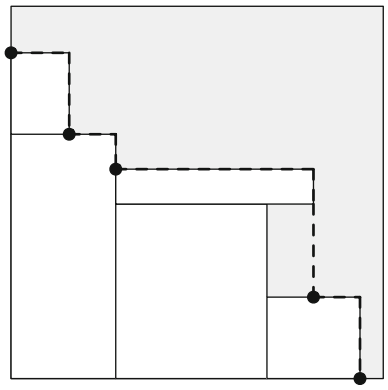
Metaheuristic algorithms have been used extensively in recent years to solve two-dimensional bin-packing problems. In short, metaheuristic methods are general frameworks that try to improve the direction of the search for the best solution, thus finding a better solution at every iteration. There are no guarantees of finding an optimal solution, but many metaheuristics implement some form of stochastic or linear optimization. Genetic algorithms, simulated annealing, and tabu search are examples of metaheuristic algorithms.

Hopper and Turton (2001) review several approaches developed to solve two-dimensional packing problems with metaheuristic algorithms. Genetic algorithms (GAs) were first used in the mid-1980s to solve strip and bin-packing problems. Many employ a two-step approach referred to as a hybrid genetic algorithm. Encoded solutions corresponding to physical layouts are manipulated by GAs that evaluate the solutions using decoding algorithms. In some cases, these decoded layouts correspond to non-overlapping packings that are obtained using a bottom-left packing heuristic. Other researchers have used a sliding principle that gives priority to the downward shifting of the rectangle being packed for the decoding routine.

Other genetic algorithms incorporate layouts directly into the encoding technique. For example, postfix strings corresponding to packing layouts can be manipulated by GAs. In the example shown in Fig. 5, the A B + and CD* substrings correspond to placements of two rectangles that are horizontally or vertically adjacent, respectively.

It is also possible for genetic algorithms to operate without encodings. An initial layout can be modified by rotating, translating, and/or relocating an item (or subset of items) in the layout. These operators correspond to hill-climbing and the mutation and recombination features of GAs. Hopper and Turton (2001) compare some meta-heuristic algorithms to two-dimensional random search and heuristic packing routines. The comparison is made in terms of the solution quality and the computation time for a number of packing instances of different sizes.

Simulated annealing, tabu search and exact algorithms have also been used to compute solutions to two-dimensional bin and strip packing problems. See (Lodi et al. 2002) for a survey of some of these approaches. A simulated annealing approach was first applied to a pallet loading problem (i.e., a three-dimensional packing problem that has been reduced to its two-dimensional footprint). Simulated annealing is a hill-climbing approach where solutions that are worse may be accepted as dictated by a cooling schedule which is determined by a given probability function. For the pallet loading problem, the number of feasible solutions for a box is equated with multiples of the item length. Neighborhoods are defined by moving each item in a solution to another position (with some restrictions). As a result, the simulated annealing heuristic would allow both legal and illegal packings as it attempted to improve the solution quality. The objective function must then minimize any overlaps that occur in the packing layout.



Bin-Packing, Fig. 6 Defining corner points

The tabu search strategy utilises a search scheme and a candidate neighborhood that is consturcted from a feasible solution: a heuristic recombines a subset of items currently packed into k different bins along with one item packed into a bin that is likely to be emptied. The value of k is also updated during the search to escape from local optima. A mechanism (i.e., the use of memory) must be built into the tabu search to prevent the heuristic from returning to recently examined packings.

Lower bounds are used to guide search strategies in exact algorithms whose goal is to find optimal solutions. For example, in one branching scheme, each node in the search tree represents a subset of packed rectangles which define a set of corner points for the bottom-left placement of unpacked items (see Fig. 6). The use of bounds to traverse a search tree corresponds to the selection of branches to investigate or ignore.

The average performance of exact algorithms and metaheuristics are typically evaluated through extensive computational experiments using benchmark data sets as described in (Parreño et al. 2008). Other more recent two-dimensional and three-dimensional examples include Bekrar and Kacem (2009) and Puchinger et al. (2010).

Three-Dimensional Packing

Algorithms for obtaining heuristic solutions to three-dimensional packing problems in which boxes are to be packed into a minimum number of identical three-dimensional bins have been characterized as either local-search or construction heuristics (Faroe et al. 2003). Analgous to the two-dimensional case, local-search methods iteratively seek better packings of the boxes by examining neighborhoods of solutions, while constructive heuristics add boxes to a packing using strategies such as First-Fit or Best-Fit. Examples of recent heuristics that have employed these methods include guided local search (GLS), a two-level tabu search (TS^2PACK), and a greedy randomized adaptive search procedure (GRASP) that is combined with a variable neighborhood descent (VND) structure.

The GLS strategy has roots in constraint-satisfaction applications and uses memory (typical of tabu search methods) to guide the search of the solution space by augmenting the objective function with penalties for previously visited solutions. It begins with an upper bound calculated from an initial greedy solution and then iteratively removes one bin from the feasible solution. Translation of boxes within one bin or between bins defines the neighborhood of the local search algorithm. To speed up the search process, some boxes are temporarily fixed in position. As before, the objective function additionally reflects the total volume of an overlap between boxes.

The TS^2PACK heuristic uses a first-level tabu search that addresses the optimality of the packing problem and a second-level tabu search that finds feasible solutions for the items assigned to the bins. An initial solution is computed using a Next-Fit Decreasing packing based on box volumes and the extreme points that are identified for a given box – these indicate positions where an additional item can be accommodated with respect to the given box. Then the TS^2PACK heuristic iteratively discards the bin with the worst fitness function value (defined as the weighted sum of the volume used by the items in the bin and the number of items). Each discarded item is packed into one of the remaining bins which yields the maximum fitness function value (i.e., minimizes the height of the new packing with bin size constraints relaxed). If this packing is not feasible due to bin size violations, a second heuristic is employed to optimize the packing with respect to the bin size constraints. This heuristic is a tabu search that uses interval graphs to represent the layout. By manipulating the graphs, alternate layouts can be generated that correspond to moving boxes by locating them in different positions.

The packing performance of these and other heuristics is compared against the GRASP/VND

approach summarized in (Parreño et al. 2008). Large sets of test cases are studied that include both two and three-dimensional problem instances. The results indicate that this method obtains comparable or better solutions to the other algorithms.

The GRASP/VND heuristic is an iterative method that combines a randomized constructive phase and an improvement phase. The constructive phase iteratively fills one bin with boxes by considering the maximal spaces created by placing boxes near corners of the bin. Boxes to be packed are selected based on best-volume or best-fit criteria. This is repeated until all boxes are packed into bins.

Attempts may be made to improve the packing by moving boxes in the bins that have first been sorted by volume. Four improvement moves were proposed: move the last k percent of boxes, move a percentage of boxes in every bin that has below average occupancy, move different parts of the bins to be emptied, or combine subsets of boxes in complementary bins and refill both with the remaining boxes.

The application of improvements (i.e., the movements of the chosen boxes) was dictated by several strategies. One of these applied the VND strategy to explore the solution neighborhood defined by the four possible moves. If the GRASP/VND heuristic appeared to be stuck at a local solution, diversification iterations are applied in the constructive process which require packing the most frequently remaining boxes first.

Recent Theoretical Studies of Multi-Dimensional Packing

Several theoretical analyses have been performed for multi-dimensional bin-packing heuristics that provide performance guarantees for packing quality as well as for algorithm execution time. One example is the recent work related to polynomial time approximation schemes (APTAS) for the three-dimensional strip packing problem. It has been shown that APTAS's exist for one-dimensional bin-packing and two-dimensional strip packing problems, but an APTAS will only exist for two-dimensional bin-packing problems if P = NP. These results are reviewed by Bansal et al. (2007) who also develop two approximation schemes: one for packing three-dimensional strips with arbitrarily sized boxes and a second for packing boxes with square bases.

Their first algorithm initially applies a Harmonic transformation (i.e., using intervals similar to those defined in the 1DBPP Harmonic heuristic) to the box widths, then it creates slabs of items to form two-dimensional strip packing instances. The two-dimensional strip is then cut into slices to produce new items that are placed on top of each other in the height dimension of a three-dimensional strip. The authors prove that this algorithm has an asymptotic approximation ratio that is arbitrarily close to the Harmonic number $T_\infty \approx 1.69$. The second algorithm A packs of set I of three-dimensional boxes with square bases so that the height of the packing does not exceed $(1 + 12\varepsilon)OPT(I) + O(K)$ where $K = \varepsilon^{-O(2^{1/\varepsilon})}$.

An APTAS for packing d-dimensional cubes into a minimum number of unit cubes has been developed by Correa and Kenyon (2004) who also present a scheme for packing rectangles into at most OPT square bins whose sides have length $1 + \varepsilon$ and OPT denotes the minimum number of unit bins required to pack the rectangles.

Parallel Algorithms

Many parallel algorithms have been proposed and studied for solving the cutting stock and knapsack variants of the bin-packing problem. Heuristics have also been proposed that obtain approximate solutions to the one-dimensional bin-packing problem on various models of parallel computation.

For the shared-memory Exclusive-Read Exclusive-Write PRAM model of computation, a heuristic based on First-Fit Decreasing has been proposed which runs in $O(\log n)$ time on $n \log n$ processors (Anderson et al. 1989). This approach divides the data items into two groups. Items in the first group are partitioned into sublists that are packed into "runs" of bins. The bins are then filled using items in the second group. The algorithm relies on parallel prefix, merging, and parenthesis matching operations, and has a worst-case performance bound of 11/9.

Practical one-dimensional bin-packing algorithms (including parallelizations of the Harmonic algorithm) have also been proposed and implemented on parallel architectures such as systolic arrays, SIMD arrays, and MIMD hypercubes. Quantitative studies

and theoretical analyses have been performed on some of these approaches. The Systolic packing algorithm, for example, has a worst-case performance bound of 1.5 and executes in $O(n)$ time. Similar results were reported in Berkey (1990).

Coleman and Wang (1992) formulated an online heuristic for massively parallel systems that used interval partitioning. The average case behavior of the heuristic could be predicted when the input have a symmetric distribution. The method is asymptotically optimal, yields perfect packings, and achieves the best possible average case behavior with high probability.

See

- Combinatorics
- Computational Complexity
- Cutting Stock Problems
- Heuristics
- Knapsack Problem
- Metaheuristics
- Parallel Computing

References

Anderson, R. J., Mayr, E. W., & Warmuth, M. K. (1989). Parallel approximation algorithms for bin packing. *Information and Computation, 82*(3), 262–271.

Bansal, N., Han, X., Iwama, K., Sviridenko, M., & Zhang, G. (2007) Harmonic algorithm for 3-dimensional strip packing problem. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, New Orleans (pp. 1197–2007).

Bekrar, A., & Kacem, I. (2009). An exact method for the 2D Guillotine strip packing problem. *Advances in Operations Research*. doi:10.1155/2009/732010. Article ID 732010.

Bender, M., Bradley, B., Jagannathan, G., & Pillaipakkamnatt, K. (2007) Sum-of-Squares Heuristics for Bin Packing and Memory Allocation. *ACM Journal of Experimental Algorithmics*, *12*, 1–19. Article No. 2.3.

Berkey, J. O. (1990). *The design and analysis of parallel algorithms for the one-dimensional bin packing problem*. Ph. D. Dissertation, School of Information and Technology, George Mason University, Fairfax, VA.

Burke, E. K., Hellier, R. S. R., Kendall, G., & Whitwell, G. (2010). Irregular packing using the line and arc no-fit polygon. *Operations Research, 58*(4), 948–970.

Coffman, E. G., Jr., Courcoubetis, C., Garey, M. R., Johnson, D. S., Shor, P. W., Weber, R. R., et al. (2000). Bin packing with discrete item sizes, Part I: Perfect packing theorems and the average case behavior of optimal packing. *SIAM Journal of Discrete Mathematics, 13*(3), 384–402.

Coffman, E. G., Jr., Garey, M. R., & Johnson, D. S. (1996). Approximation algorithms for bin-packing a survey. In D. S. Hochbaum (Ed.), *Approximation algorithms for bin packing for NP-hard problems* (pp. 46–93). Boston: PWS Publishing Company.

Coffman, E. G., Jr., & Shor, P. W. (1993). Packing in two dimensions: Asymptotic average-case analysis of algorithms. *Algorithmica, 9*, 253–277.

Coleman, N. S., & Wang, P. Y. (1992). An asymptotically optimal parallel bin-packing algorithm. In *Proceedings of the fourth symposium on the frontiers of massively parallel computation*, IEEE Society, (pp. 515–516).

Correa, J., & Kenyon, C. (2004) Approximation schemes for multidimensional packing, In *Proceedings of the fifteenth ACM-SIAM symposium on discrete algorithms*, SIAM, (pp. 186–105).

Csirik, J., Johnson, D. S., Kenyon, C., Orlin, J. B., Shor, P. W., & Weber, R. R. (2006). On the sum-of-squares algorithm for bin packing. *Journal of the ACM, 53*(1), 1–65.

Faroe, O., Pisinger, D., & Zachariasen, M. (2003). Guided local search for the three-dimensional bin-packing problem. *INFORMS Journal on Computing, 15*(3), 267–283.

Fenrich, R., Miller, R., & Stout, Q. F. (1989). Hypercube algorithms for some NP-hard packing problems. In *Proceedings of the 4th conference and hypercubes, concurrent computers, and applications* (pp. 769–776). Los Altos, CA: Golden Gate Enterprises.

Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. San Francisco: W.H. Freeman.

Hopper, E., & Turton, B. C. H. (2001). A review of the application of meta-heuristic algorithms to 2D strip packing problems. *Artificial Inteligence Review, 16*(4), 257–300.

Jakobs, S. (1996). On genetic algorithms for the packing of polygons. *European Journal of Operational Research, 88*, 165–181.

Lodi, A., Martello, S., & Monaci, M. (2002). Two-dimensional packing problems: A survey. *European Journal of Operational Research, 141*, 241–252.

Martello, S., Pisinger, D., & Vigo, D. (2000). The three-dimensional bin packing problem. *Operations Research, 48*(2), 256–267.

Parreño, F., Alvarez-Valdes, R., Oliveira, J. R., & Tamarit, J. M. (2008). A hybrid GRASP/VND algorithm for two-and three-dimensional bin packing. *Annals of Operations Research, 131*, 203–213.

Puchinger, J., Raidl, G. R., & Pferschy, U. (2010). The multidimensional knapsack problem: Structure and algorithms. *INFORMS Journal on Computing, 22*(2), 250–265.

Seiden, S. (2002). On the online bin packing problem. *Journal of the ACM, 49*(5), 640–671.

Sweeney, P., & Paternoster, E. (1992). Cutting and packing problems: A categorized, application-orientated research bibliography. *Journal of the Operational Research Society, 43*(7), 691–706.

Wäscher, G., Haußner, H., & Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research, 183*(3), 1109–1130.
Xia, B., & Tan, Z. (2010). Tighter bounds of the First Fit algorithm for the bin-packing problem. *Discrete Applied Mathematics, 158*(15), 1668–1675.

Bipartite Graph

A graph or network whose nodes can be partitioned into two subsets such that its edges connect a node in each partition.

See

- ▶ Assignment Problem
- ▶ Graph Theory
- ▶ Network Optimization
- ▶ Transportation Problem

Birth-Death Process

A stochastic counting process that satisfies the following is called a birth-death process: (1) changes from state n (sometimes written more generally as state E_n) may only be to states $n+1$ or $n-1$ (i.e., changes can only be ± 1 unit); (2) the probability of a birth (death) occurring in the "small" interval of time, $(t, t+dt)$, given that the process was in state n at the start of the interval, is $\lambda_n dt + o(dt)[\mu_n dt + o(dt)]$, where $o(dt)$ is a function going to 0 faster than dt. Such processes are in fact Markov chains in continuous time. The system size of an M/M/1 queueing system is an example of a birth-death process where $\lambda_n = \lambda(n = 0, 1, 2, \ldots)$ and $\mu_n = \mu(n = 1, 2, \ldots)$. Markov chains; Markov processes.

Bland's Anticycling Rules

A set of pivot rules, the application of which to linear-programming (degenerate) problems, prevents cycling in the simplex algorithm. Their basic principle is that whenever there is more than one eligible candidate in selection of the variable entering the basis, or the variable leaving the basis, the candidate with the smallest index is chosen.

See

- ▶ Anticycling Rules
- ▶ Cycling
- ▶ Degeneracy

References

Bland, R. (1977). New finite pivoting rules for the simplex method. *Mathematics of Operations Research, 2*(2), 103–107.
Dantzig, G. B., & Thapa, M. N. (2003). *Linear programming 2: Theory and extensions*. New York: Springer.

Blending Problem

The linear-programming problem of blending raw materials, for example, crude oils, meats, to produce one or more final products, for example, fuels, sausages, so that the total cost of production is minimized. The problem is subject to restrictions on material availability, blending requirements, quality restrictions, etc.

See

- ▶ Activity-Analysis Problem
- ▶ Stigler's Diet Problem

Block Pivoting

The process of entering several nonbasic variables simultaneously into the basis in the simplex algorithm.

See

- ▶ Simplex Method (Algorithm)

Block-Angular System

A linear system of equations for which its matrix of coefficients A can be decomposed into k separate blocks of coefficients A_i, where each A_i represents the coefficients of a different set of equations. This structure typically represents a system consisting of k subsystems whose activities are almost autonomous, except for a few top-level system constraints whose variables couple the k blocks of the subsystems. Such systems can also have a few variables external to the blocks that couple the blocks.

See

- ▶ Dantzig-Wolfe Decomposition Algorithm
- ▶ Large-Scale Systems
- ▶ Weakly-Coupled Systems

Block-Triangular Matrix

A matrix which is lower (upper) triangular except for a number of blocks along the diagonal.

See

- ▶ Triangular Matrix

Bonferroni Inequality

Result in basic probability that provides a general lower bound on the intersection of events $E_1,\ldots,E_n$:

$$P(\bigcap_{i=1}^{n} E_i) \geq 1 - \sum_{i=1}^{n} P(E_i^c).$$

Note that the events need not be independent (nor mutually exclusive).

Applied in stochastic simulation output analysis to make statements about the overall confidence level of multiple performance measures (simultaneous confidence intervals). For example, for three output performance measures each with 99% confidence levels, the overall confidence level would be at least 97%.

Bootstrapping

In forecasting, the term bootstrapping refers to models that have been developed by regressing an individual's (or group's) forecasts against the inputs that the individual used to make the forecasts.

See

- ▶ Forecasting
- ▶ Regression Analysis

Bootstrapping: Resampling Methodology

Linda Weiser Friedman[1] and Hershey H. Friedman[2]
[1]Baruch College, City University of New York, New York, NY, USA
[2]City University of New York, Brooklyn, NY, USA

Introduction

Researchers typically encounter many situations in which parametric statistical techniques are less than ideal. The t-statistic, for example, assumes that the data were sampled from a normal distribution. Of course, much real-world data follow distributions that are far from normal, and may in fact be quite skewed. Suppose a researcher is investigating data that is known to follow an exponential distribution. Clearly, it would take an extremely large sample and a great deal of manipulation (e.g., averages of averages), for the central limit theorem to apply. In many cases, there is no parametric test for the measurement of interest because the sampling distribution of that measurement may be unknown and thus there would be no tractable analytic formulas for estimating such measures, for example, the difference between two medians (Mooney and Duval 1993, p. 8).

There are a number of nonparametric statistical techniques that do not rely on distributional assumptions and often may be used in place of the more traditional parametric tests. Many nonparametric techniques, however, work only with the median as a measure of central tendency (e.g., Mann-Whitney-Wilcoxon). This may present a problem for researchers who are more interested in the mean as the measure of interest.

The bootstrap statistic (Efron 1981, 1982; Mooney and Duval 1993) is a nonparametric, computer-intensive resampling technique, which makes no distributional assumptions and may be used for estimation and hypothesis testing. The bootstrap, jackknife, and other related resampling methods are beginning to generate interest among management scientists. Indeed, these tools can be very useful for the type of data that is frequently encountered by management scientists.

The Bootstrap Method

With traditional parametric inference, a sample is taken and a statistic, often the sample mean, is computed. This statistic is assumed to follow a known distribution (normal, t-distribution, F-distribution, χ^2-distribution, etc.), which then allows the researcher to perform hypothesis tests and/or estimate confidence intervals. With bootstrapping, which was developed mainly to determine the standard error for other types of estimates (Efron and Tibshirani 1991), the sample itself is used to construct a sampling distribution by selecting from it many resamples, or pseudo-samples.

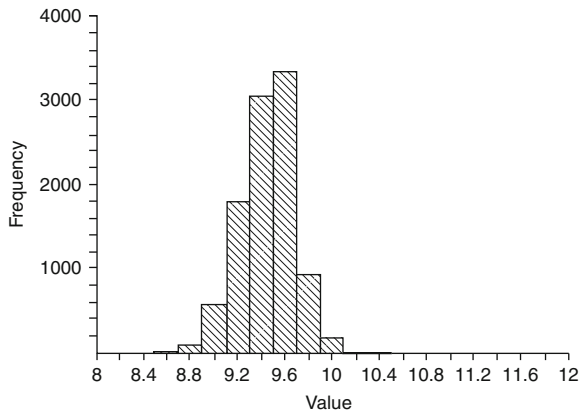
Resampling from the sample is done with replacement. Thus, it is like sampling from an infinite population with a composition that exactly matches that of the sample that was originally drawn. After resampling a great number of times one may construct a sampling distribution for a statistic of interest, such as the mean, median, or any percentile. This distribution, which is entirely based on the original sample and not on any theoretical distribution, may then be used to test hypotheses about measures of interest and to construct confidence intervals.

To illustrate the method, two illustrative examples are presented. The first is a hypothesis test for a sample from a single population; the second, for samples from two presumably different populations.

Bootstrapping: Resampling Methodology, Table 1 Sample data and statistics, Example I

Life	Frequency
8.0	3
9.0	5
10.0	6
11.0	2

$\bar{x} = 9.438$, $s = 0.964$, $n = 16$



Bootstrapping: Resampling Methodology, Fig. 1 Histogram of mean lifetimes, 10,000 resamples, Example I

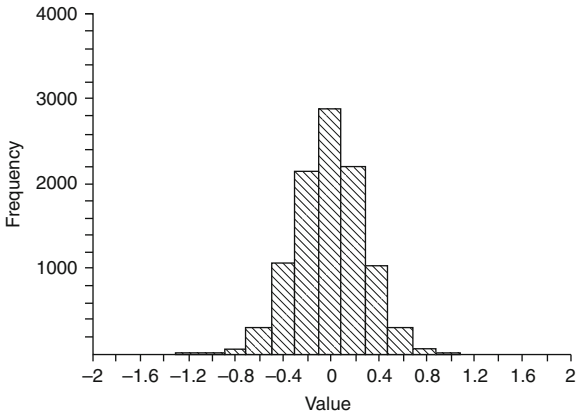
Example I A company claims that the average life of a part that it manufactures is at least 10 hours. A sample of 16 parts is taken in order to test this claim. The sampled data is summarized in Table 1.

A parametric analysis using the t-statistic would have to assume that the underlying population is normally distributed since the sample is too small to rely on the central limit theorem. Moreover, this type of data is usually not normally distributed, or even symmetrical.

Using the bootstrap method, 10,000 resamples, each of size $n = 16$, were taken from the original data. Figure 1 is a histogram of the 10,000 resampled means. One can see that the means seem to be hovering about the values 9.3 to 9.7 hours, and very few are actually above 10.0. Table 2 confirms that only a small fraction of the means were above 10.0. As a matter of fact, the 95 percent one-sided confidence interval is bounded by the value of 9.8125 hours. This means that only 5 per cent of the resamples had mean values above 9.8125. Clearly, the claim that the average life of these parts is at least 10 hours should be rejected.

Bootstrapping: Resampling Methodology, Table 2 Frequency distribution of mean lifetimes, Example I (note that each category covers all values within 0.1 of its center)

Center value	Frequency	Percent	Cum percent
8.6	7	0.1	0.1
8.8	92	0.9	1.0
9.0	583	5.8	6.8
9.2	1798	18.0	24.8
9.4	3057	30.6	55.4
9.6	3342	33.4	88.8
9.8	920	9.2	98.0
10.0	184	1.8	99.8
10.2	16	0.2	100.0
10.4	1	0.0	100.0



Bootstrapping: Resampling Methodology, Fig. 2 Histogram of mean lifetimes, 10,000 resamples, Example II

Bootstrapping: Resampling Methodology, Table 3 Sample data and statistics, Example II

Group 1	Group 2
13.8	12.6
13.3	12.4
13.7	12.9
13.6	13.3
15.2	14.2
14.4	13.0
13.6	13.4
13.3	12.9
13.6	13.5
13.8	13.6
$\bar{x}_1 = 13.83$	$\bar{x}_2 = 13.18$
$s_1 = 0.57$	$s_2 = 0.53$
$n n_1 = 10$	$n_2 = 10$

Bootstrapping: Resampling Methodology, Table 4 Frequency distribution of mean lifetimes, Example II (note that each category covers all values within 0.1 of its center)

Center value	Frequency	Percent	Cum percent
21.2	1	0.0	0.0
21.0	4	0.0	0.1
20.8	42	0.4	0.5
20.6	273	2.7	3.2
20.4	1065	10.6	13.9
20.2	2164	21.6	35.5
0.0	2876	28.8	64.3
0.2	2189	21.9	86.1
0.4	1026	10.3	96.4
0.6	305	3.0	99.4
0.8	50	0.5	99.9
1.0	5	0.1	100.0

Example II A similar type of analysis can be done for a two-sample test. Consider the data in Table 3, representing the life (in weeks) of similar parts from two different manufacturers or two different production processes. As in Example I, a parametric test would require an assumption of normally distributed lifetimes, which again may be unrealistic.

With the bootstrapping approach, first combine the two groups of data into one (i.e., under the assumption that H_0 is true). Then, this combined group is resampled to produce two groups of data items, and the mean difference of the two groups $\bar{x}_1 - \bar{x}_2$ is recorded. This resampling is done many times, and the resulting mean differences are compared with the observed mean difference in the original set of data.

In the above example, the observed mean difference is 0.65 weeks (13.83–13.18). The question is, what is the likelihood that this difference occurred by chance? Since this is a two-tailed test, consider resamples for mean differences greater than 0.65 or less than -0.65. Figure 2 contains the histogram of the mean differences of 10,000 re-samples, in which each resample produced two groups of size $n = 10$ each. Examination of this histogram and of Table 4 shows that almost all of the mean differences fall between–0.5 to 0.5. Actually, only 1.85% of the resampled mean differences were either greater than 0.65 or below–0.65. At a significance level of 0.05, reject the hypothesis that the two population means are the same.

Concluding Remarks

Bootstrapping is clearly a technique that is very useful to researchers. It should, however, be pointed out that this technique is totally dependent on the integrity of the original sample of data. If the sampled data is indeed a good representation of the underlying distribution, inferences based on resampling this data will be valid. On the other hand, if the original sample, say, over represents high values of the output distribution, then the resamples and inferences based on them cannot be trusted. If the sample is biased, the resampling technique may reflect and possibly magnify these biases.

Some areas in operations research and management science that have made use of bootstrapping and other resampling techniques include: quality control (Jeske 1997; Seppala 1995), analysis of simulation output (Friedman and Friedman 1995; Kim et al. 1993), neural networks (LeBaron 1998; Shimshoni 1998), performance evaluation (Cho 1997), and production (Jochen 1997).

Mooney and Duval (1993) describe how the bootstrap procedure may be used with SAS and RATS. Resampling Stats (Simon 1995), a simple computer package for bootstrapping, is user-friendly, relatively inexpensive, and comes with numerous examples. Fan and Jacoby (1995) describe a SAS/IML program for performing the bootstrap resampling technique in regression analysis. Bootstrapping can also be done with spreadsheets (Willemain 1994).

See

▶ Regression Analysis

References

Cho, K. (1997). Performance assessment through bootstrap. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19*, 1185–1198.

Efron, B. (1981). Nonparametric estimates of standard error: The jackknife, the bootstrap and other methods. *Biometrika, 68*, 589–599.

Efron, B. (1982). *The jackknife, the bootstrap and other resampling plans*. Philadelphia: SIAM.

Efron, B., & Tibshirani, R. (1991). Statistical data analysis in the computer age. *Science, 253*, 390–395.

Fan, X., & Jacoby, W. G. (1995). Bootsreg: An SAS matrix language program for bootstrapping linear regression models. *Educational and Psychological Measurement, 55*, 764–768.

Friedman, L. W., & Friedman, H. H. (1995). Analyzing simulation output using the bootstrap method. *Simulation, 64*, 95–100.

Jeske, D. R. (1997). Alternative prediction intervals for pareto proportions. *Journal of Quality Technology, 29*, 317–326.

Jochen, V. A. (1997). Using the bootstrap method to obtain probabilistic reserves estimates from production data. *Petroleum Engineer International, 70*, 55 +.

Kim, Y. B., Willemain, T. R., Haddock, J., & Runger, G. C. (1993). The threshold bootstrap: A new approach to simulation output analysis. *Proceedings of the 1993 Winter Simulation Conference*, 498–502.

LeBaron, B. (1998). A bootstrap evaluation of the effect of data splitting on financial time series. *IEEE Transactions on Neural Networks, 9*, 213–220.

Mooney, C. Z., & Duval, R. D. (1993). *Bootstrapping: A nonparametric approach to statistical inference*. Newbury Park, CA: Sage Publications.

Seppala, T. (1995). Statistical process control via the subgroup Bootstrap. *Journal of Quality Technology, 27*, 139–153.

Shimshoni, Y. (1998). Classification of seismic signals by integrating ensembles of neural networks. *IEEE Transactions on Signal Processing, 46*, 1194–1120.

Simon, J. L. (1995). *Resampling stats user's guide*. Arlington, VA: Resampling Stats.

Willemain, T. R. (1994). Bootstrap on a shoestring: Resampling using spreadsheets. *The American Statistician, 48*, 40–42.

Bounded Rationality

The concept that a decision maker lacks both the knowledge and computational skill required to make choices in a manner compatible with economic notions of rational behavior.

See

▶ Choice Theory
▶ Decision Analysis
▶ Multiple Criteria Decision Making
▶ Satisficing

References

Simon, H. A. (1955). A behavioral model of rational choice. *Quatrely Journal of Economics, 69*, 99–118.

Simon, H. A. (1957). *Models of man: Social and rational*. New York: John Wiley & Sons.

Bounded Variable

A variable x_j in a linear-programming problem that is required to satisfy a constraint of the form $0 \leq x_j \leq b$, $-b \leq x_j \leq 0$, or $b_1 \leq x_j \leq b_2$, where b is some positive constant and $b_1 \leq b_2$.

See

► Linear Programming

Branch

To move and analyze a new computational path (i.e., branch) based on the results obtained from a previous path.

See

► Branch and Bound

Branch and Bound

A method for solving an optimization problem, by successively partitioning (branching) the set of feasible points to smaller subsets, and solving the problem over each subset. The resulting problems are called subproblems or nodes in the enumeration tree. The idea in branch and bound is that the optimal solution to the problem is the best among the optimal solutions to the subproblems. To reduce the number of subproblems solved, best-case bounds are computed by solving relaxed problems defined at the nodes. If the best-case bound on a solution to a subproblem is worse than the best available solution, the subproblem is eliminated from consideration (fathomed). Branch and bound techniques are frequently used to solve integer-programming problems, as well as in global optimization.

See

► Global Optimization
► Integer and Combinatorial Optimization
► Integer-Programming Problem

Brownian Motion

A one-dimensional Brownian motion $\{B(t),\ 0 \leq t\}$ is a continuous-time, Markovian, real-valued stochastic process having continuous sample paths; its distribution is Gaussian with mean function $E[B(t)] = \mu t$ and covariance function $\mathrm{Cov}[B(s), B(t)] = \sigma^2 \min(s, t)$. An n-dimensional Brownian motion is a stochastic process on $\mathbb{R}^n$ whose n components are independent one-dimensional Brownian motions. Named after Scottish botanist Robert Brown. Also known as the Wiener process, named after mathematician Norbert Wiener.

See

► Markov Processes

BTRAN

The procedure for computing the dual variables in a simplex iteration, when the *LU* factors of the basis matrix are given in product form. The name BTRAN (backward transformation) derives from the fact that the eta file is scanned backwards in the solution process.

See

► Eta File

Buffer

The queue or the waiting room in a queueing system. Most often used for networks, especially tandem networks or series queues.

See

► Queueing Theory

Bulk Queues

Arrivals to a queueing system may consist of more than one customer at a time, and/or service might process more than one customer simultaneously.

See

► Queueing Theory

Bullwhip Effect

► Supply Chain Management

Burke's Theorem

The steady-state departure process of a stable M/M/*c* queueing system is a Poisson process with the same rate as the arrival process, irrespective of the service rate.

See

► Queueing Theory

Business Intelligence

Paul Gray
Claremont Graduate University, Claremont, CA, USA

Introduction

Business Intelligence (BI) systems are sophisticated analytical tools that present complex organizational and competitive information in a way that allows decision makers to decide quickly and appropriately. While the term Business Intelligence is relatively new (it was introduced in 1989, popularized in the 1990s), computer-based BI systems existed, in one guise or another, decades prior to that. BI-type functionality was available previously to varying degrees in Financial Planning Systems (4GLs), Executive Information Systems (EIS), Decision Support Systems (DSS), Data Mining, and On Line Analytic Programming (OLAP). With each new iteration, capabilities increased as enterprises grew ever-more sophisticated in their computational and analytical needs and as computer hardware and software matured. This article explores the capabilities of state-of-the-art BI, their benefits to adopters, and the role of Analytics in BI.

BI describes data-driven decision support systems (Power 2005) for managers. In its initial form, it involved business analysts who refined (mostly internal) business data to create input for management. Such systems have been marketed commercially since the 1960's, if not earlier. BI is now closely linked to Analytics, the use of quantitative methods for solving organizational problems. BI is broader than Analytics because it involves soft methodologies and information systems, as well as Operations Research (OR).

Objective and definition of BI: The objective of BI is to improve the timeliness and quality of the input to the decision process.

To achieve this objective, BI systems combine:

Data gathering		
Data storage	with	Analysis
Knowledge management		

to evaluate complex organizational and competitive information and present the results to planners and decision makers.

The first three operations are inputs, typically performed by people with information systems and data analysis skills. The skills of Analytics are brought to the table by people trained in OR, statistics, and other quantitative disciplines.

Implicit in its definition is the idea that BI systems provide actionable information and knowledge at the right time, at the right place, and in the right form.

Problems to which BI is Applied: BI aims to convert data available to the organization into

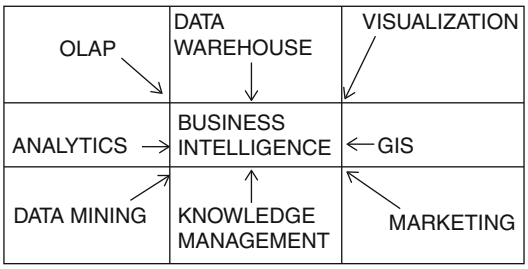
information and, through analysis, into knowledge. Among the many tasks that BI performs are:

- Examine the opportunities for
  - proposed products,
  - mergers and acquisitions,
  - acquiring new customers, and
  - locating sites for new branches.
- Create forecasts based on historical data, current performance, and estimated future performance. Futures methodologies such as Delphi and cross-impact analysis are discussed in Glenn and Gordon (2008).
- Monitor key performance indicators (KPI) both for the organization and its competitors.
- Do "what if" analysis to examine the impacts of changes and of alternative scenarios.
- Ad hoc access to data to answer specific, non-routine questions.

These examples cover both regular, repetitive scheduled reporting (e.g., monthly reports on sales by region, department, or strategic business unit), and special investigations aimed to solve specific problems. Forecasting and many of the specific problem studies involve OR modeling that uses the organization's data warehousing capabilities for the underlying information. For example, specific studies undertaken in response to a crisis or an opportunity such as a contract proposal.

BI vs. Competitive Intelligence: Business Intelligence uses technologies, processes, and applications to analyze mostly internal, structured data, and business processes, while Competitive Intelligence (discussed below) gathers, analyzes, and disseminates information from both external and internal sources to provide a framework for assessing the organization's position relative to its industry and non-industry competitors and its vulnerability to disruptive technologies.

Previous Systems: Present-day BI systems reflect a series of iterations to obtain their present functionality. These included (1) 3rd Generation financial planning languages that allowed writing relations in words rather than symbols. (For example, rather than saying S = M*MS, one could write Sales = market*market share.) (2) Executive information systems that can create PowerPoint charts to brief management on the current state of the business. (3) On-Line Analytic Processing in which data warehouses that store data in the form of 2-dimensional relational data bases are used to create multidimensional data cubes (see below). Although each of these elements is more sophisticated than the one before, they were individual systems, while the hallmark of current BIs is the integration of such systems.



Business Intelligence, Fig. 1 Software Components of Inputs to Business Intelligence

BI Input Software

BI is deeply tied to the ability to store data bases and to compute at the organizational or departmental level. Key elements include data warehouses and data marts. As shown in Fig. 1, many software capabilities are involved.

The software components used in BI include:

- A *Data Warehouse* is a collection of data bases that contain both current and historical information about the organization. The warehouse is separate from operational systems that support on-line transaction systems. It contains "a single version of the truth" and is intended to support understanding of the organizational data over time. It is particularly important for BI.

To create the single version of the truth, data goes through a process known as ETL (extract, transform, load). The ETL applies procedures that extract data from selected sources, transforms it into the format of the data warehouse that is consonant with the warehouse's rules, and then stores the data into the warehouse or mart. ETL is important for BI because it standardize the data and eliminates redundancies and inaccuracies.

Data warehouses come in two sizes:

- A data warehouse, which support an entire organization or one of its major portions.
- A data mart that is a smaller version of a data warehouse but has all features of a warehouse. It can

Business Intelligence, Table 1 Characteristics of the Data Warehouse

Characteristics	Description
Subject oriented	Data are organized by how users refer to it.
Integrated	Inconsistencies are removed in both nomenclature and conflicting information, i.e, combining of all related data around a common identifier/key
Non-volatile	Read-only data are not updated by users
Time Series	Data are time series, not current value. A typical data warehouse has 5 to 10 years of data.
Summarized	Operational data are aggregated into decision usable form where appropriate
Larger	Much more data is retained than in transaction systems because it offers time series.
Non-normalized	Data can be redundant for ease of retrieval and use.
Metadata	Data about the data are available to users and to warehouse personnel.
Input	Include both operational data and external data

Source: Gray and Watson (1998)

be dependent or independent. If dependent, it contains a subset of the data warehouse needed by specific groups, such as Analytics. But, multiple independent data marts cannot substitute for a data warehouse because data integrity is not maintained among them.

Over time, a number of specialized data warehouses have evolved. They include operational data stores, real time warehouses, prototype warehouses, and exploration warehouses discussed below.

The characteristics of the data warehouse are listed in Table 1. These characteristics assume that the data warehouse is physically separated from operational systems and that its databases are not used for on-line transaction processing (Inmon 1992).

- *OLAP* (**o**n-**l**ine **a**nalytic **p**rocessing) is used to analyze multidimensional data for a BI. It is used for such tasks as sales analysis, budgeting, forecasting, and financial reporting where it is necessary to manipulate and consolidate data from multiple sources. Data are specially configured for OLAP into data cubes to allow complex questions to be answered more quickly than for relational data bases. OLAP has subdivided into relational, multidimensional, hybrid, and other forms, which are typically referred to as ROLAP, MOLAP, and HOLAP.
- *Analytics* refers to the use of quantitative and statistical methods together with extensive computing and modeling to make sense of the data. It is the area of BI that attracts operations researchers. Mathematics is the base for Analytics. The objective is to obtain realistic and, if possible, optimal alternatives for decision making about the future. Analytics is discussed in more detail below.
- *Data Mining* [also referred to as knowledge data discovery (KDD)] is a form of predictive analytics discussed below. It is a set of analytical techniques to obtain new insights from the data in the data warehouse that an analyst or a manager had not thought to ask. It is used to find answers that reports and queries do not reveal effectively. KDD seeks to find patterns in data and to infer rules. Data mining differs from conventional hypothesis testing in that it looks at data for the relationships it contains to form hypotheses that can be tested. KDD techniques include neural networks, expert systems, fuzzy logic, intelligent agents, multidimensional analysis, data visualization, and decision trees. Data mining is used in wide range of topics, e.g., to identify where people are likely to take vacations, detect fraud, analyze loan quality, and the reported (but apocryphal) association that men who buy diapers on Friday night also buy beer.
- *Knowledge Management*. Knowledge can be tacit and explicit. Tacit knowledge is what is in one's head but cannot usually be expressed, although there are techniques for obtaining some tacit knowledge. Explicit knowledge is about what can be written down, stored, and retrieved. Knowledge management is about knowing what the organization knows and finding new knowledge that is needed when the organization does not know. It focuses on creating, sharing, and applying knowledge. In BI, the explicit information in the data warehouse and in reports is merged with the tacit knowledge in the heads of analysts and professionals.
- *Geographic Information Systems*. These systems link data bases to geographic maps of physical locations. They are used to analyze spatial phenomena. For example, they allow overlaying of customer, distribution center, retailer, and other information about a firm's and its competitor's products.

- *Marketing*. Analytics are used to understand the implications of existing and proposed policies in the marketplace. For example, data from aggregators and from the firm are used to create forecasts of market size and market size.
- *Visualization*. Visualization refers to methods to present information on-screen in a form comprehensible to non-technical managers. It does not replace Analytics; it focuses the analytic results. By exploiting visuals, it provides an overview of complex data sets and allows for identifying relationships and trends in data and in analytical results.

BI Outputs: Dashboards and Reports

Dashboards. In BI, a dashboard is way of communicating results in a form that is easily understood by managers. A dashboard is a visual screen that shows the key performance indicators. The data, drawn from internal information systems and analyses, not only summarize the current status, but also provide historical data, warning levels, next steps, and notices. It includes financial and non-financial measures.

The idea of a dashboard has been in use since the 1960's. At that time, summarized data for managers was displayed on color slides at regular management meetings. For example, the experience at AT&T from color slides was that if the dashboard slides presented the current data in the same format at each meeting, managers would rapidly find and be sensitive to changes that required action.

Introducing the computer provided an instant display device, improved visualization, and provided data on the desktop tailored to each user. For example, the VP for manufacturing and the VP for human resources can see results specifically oriented to their issues. Furthermore, the displays allow drill down; that is they start with a broad view and then let the user see greater and greater detail..

The three main applications are:

- *monitoring* information at a glance. Usually involves key performance indicators (KPI) in graphical, symbol, or symbolic form.
- *analysis* of exceptions to find root causes of problems. Summarized multi-dimensional data and drill down in "slice and dice" fashion are used.
- *Identify actions* to solve problems based on access to detailed operational data, queries, and reports.

Reports include:

- regular, repetitively scheduled documents (e.g., monthly sales by region, department, or strategic business unit),
- exception reports which are produced whenever parameters are outside pre-specified bounds,
- documents presenting the results of special investigations (often in response to requests from BI users), and
- custom data cubes based on specific requests from analysts.

Forecasting and many of the specific studies involve OR modeling that uses the organization's data warehousing capabilities for the underlying information. For example, specific studies are undertaken in response to a crisis or an opportunity such as a contract proposal.

BI Architecture

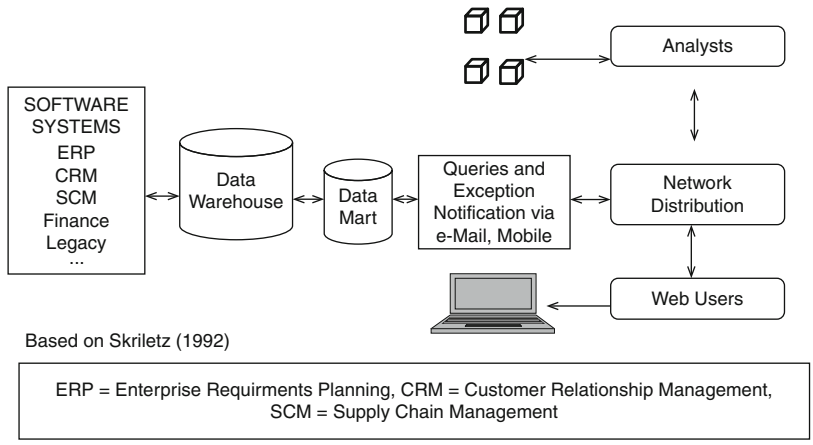
Figure 2 (based on Skriletz 2002) is typical of the architecture for a large installation that centers on the use of Web technology for distribution. As shown, the input data come from a variety of systems into the data warehouse. The specific data needed for BI is downloaded to a data mart used by planners and executives.

As shown, the specific applications used for BI include the organizational focus and the audience (Skriletz 2002).

The left side of Table 2 shows the business focus of the technologies, while the right side shows the levels of people in the organization who are the consumers of the intelligence. At the bottom of the hierarchy is transaction processing based on application-specific data in the warehouse or in ERP or in sales systems. The next level involves processing the data so that it is useful to first level managers. Here, Analytics and pattern analysis are performed and data are presented in visual form. At the top level, predictions, compilations of competitive analyses, and summary presentations for executives are created.

Tools. Many of the tools for BI are used for other applications as well. They include:

- simple querying and reporting,
- on-line analytic processing (OLAP),



Business Intelligence, Fig. 2 BI Architecture

Business Intelligence, Table 2 Business Intelligence Aplications

Organizational focus				Audience
Strategy	Predictive and Prescriptive Analytics	Competitive Intelligence	Intelligence Presentation and Visualization	**Top executives**
Operations Analysis	Data Analytics			**Operations Directors**
Operations Monitoring		Heuristic Pattern Analysis		**Operations Supervisors**
Transaction Processing	Platform for BI - Manage Through Metadata Application-Specific BI (e.g., SAS, IBM, Oracle, SAP)			**Operations staff**

- statistical analyses and data mining,
- forecasting, and
- geographic information systems and visualization.

In addition, the extraction, translation, and loading (ETL) tools of data warehousing are important for BI because they help standardize the data so it can be analyzed with accuracy and provide a single truth. When operational data is used, as from an Operational Data Store (ODS), the objective is to get dynamic data that reflects the current situation.

The key dissemination method for business technology is internet technology, whether it be an intranet within the firm or an extranet connected to suppliers and/or clients. The idea is to reach everyone who needs specific data, rather than a few at corporate headquarters.

Business Analytics

In the 20th century, most information systems were used to standardize routine business processes to minimize cost and time. Fairly sophisticated decision support systems and data warehouses were in use, but these systems rarely directly affected the ways decisions were made (Drucker 1999). BI was mostly the province of the information systems groups in organizations. It centered on providing inputs for data-based decision making. It was only after the turn of the century that it was generally realized that applying Analytics would improve to data-based decision making. This realization was reinforced by leading vendors, such as IBM, Microsoft, Oracle, and SAP, acquiring major BI software firms and investing in expanding BI software capabilities. It became clear that the analytic skills and the methods of OR analysts are needed to exploit information technology capabilities.

The definition of business Analytics is still in flux. Davenport and Harris (2007) defined it as "the extensive use of data, statistical and quantitative analysis, explanatory and predictive models, and fact-based management to drive decisions and actions." In this definition, Analytics is a subset of BI, that is, technologies and processes that use data to understand and analyze performance. A broader

definition comes from IBM who uses it to refer to both software applications and analytic solutions (Lustig et al. 2010). In their view, software includes BI, performance management, prediction, optimization, enterprise information management, content, and collaboration. Analytic solutions involve finance, risk, fraud, customer relations, human capital, and supply chain. Underlying Analytics is the idea that data and information are strategic assets.

Analytics can be divided into three categories (Lustig et al. 2010):

- Descriptive analytics
- Predictive analytics
- Prescriptive analytics

All three categories start with the underlying idea that data and information are strategic assets.

Descriptive analytics is the classic form of BI. It starts by examining, consolidating, and classifying data. Data sources include information from departments (marketing, sales, operations, accounting), enterprise systems (ERP, CRM, and Supply Chain Management SCM), as well as spreadsheets, other databases, and external data from 3rd parties. The outputs are ad hoc and exception reports, dashboards, KPI, statistical analyses, drill down, and answers to ad hoc queries about business performance. These outputs allow for the managing and monitoring of business processes. Descriptive analytics are often inputs to predictive and prescriptive analytics.

Predictive analytics combines the data within a wide variety of mathematical procedures to create models that explain and/or predict performance. It is based on inherent relations between the data inputs and outcomes. Predictive analytics uses data on what happened in the past to detect patterns and relations to make forecasts. Its methods include, among others (Lustig, et al. 2010):

Data mining	Correlations among data
Forecasting	Extrapolations of trends into the future
Monte Carlo simulation	What may happen if changes occur
Root-cause analysis	Evaluation of why things happened
Pattern recognition	Alerts when unusual situations occur
Predictive modeling	Forecasts by Delphi or other methods

Prescriptive analytics refers to mathematical techniques that provide understanding of alternative courses of actions when there are competing objectives, requirements, and constraints. It involves both static and stochastic optimization. The former leads to determining the best outcome, while the latter considers the effects of data uncertainty to improve decisions. Given the increase in computer speed and memory, improved algorithm performance, and in data quality, prescriptive analytics can be run in near-real-time so they can affect operational as well as strategic decisions.

Integrating Analytics and BI

Where traditional BI depends principally on aggregating, evaluating and manipulating the information in the data warehouse, Analytics adds modeling and optimization. Irrespective of which type of Analytics (descriptive, predictive or prescriptive) is used, the results need to be communicated to the user community.

This communication capability involves a series of steps (Shapiro 2010):

- Develop models to optimize decisions for key performance indicators.
- Select the right modeling system. It may be customized or off the shelf.
- Define the database needed for the optimization model. It may be customized or off-the-shelf
- Create the decision database. This may require new ETL routines and descriptive models
- Link the database and the outputs from the optimization model to the organization's reporting tools to be able to communicate results to users.
- For strategic and tactical decisions, reuse criteria for alerts and redo modeling studies at regular intervals. For operational decisions, exercise the operational models in real-time with current data.

Competitive Intelligence

The notion of competitive intelligence (CI) as spy vs. spy, fed by such examples as Japan and China allegedly stealing U.S. industrial secrets, is far from the real situation. That does not mean that companies do not try to find out as much as possible about their current and potential competitors. The people involved, however, claim that they do so in a legal and ethical manner. CI is defined by the Society for

Competitive Intelligence Professionals (SCIP) as the process of monitoring the competitive environment. To do so, analysts systematically gather, analyze, and manage information that can affect a company's plans, decisions, and operations.

The competitive intelligence cycle includes:

- Determine the intelligence needs of decision makers
- Collect information to meet these needs
- Analyze the data and recommend actions
- Present results to the decision makers
- Use the response to the findings to refine collection.

The focus is on determining both the current activities and the likely intentions of other firms and of governments. It also includes looking for the possible appearance of disruptive technologies and finding out about how competitors are responding to your actions.

The collected raw data (facts, statistics) are organized and then analyzed to find patterns, trends, and relationships. The tools used include:

- Simulations of alternative scenarios to test what if conditions
- Data mining of information about both competitors and the firm
- Assessing competitor technologies by tracking (and extrapolating from) patent filings.
- Attending trade shows and conferences
- Scanning publicly available data: public records, the Internet, press releases, and mass media.
- Talking with customers, suppliers, partners, industry experts

Much of the data gathering work is terribly dull and routine. To be effective, it has to be someone's (or some group's) responsibility.

For many organizations, the only basis for evaluating their competitors is by applying the SWOT technique: **S**trengths, **W**eaknesses, **O**pportunities and **T**hreats. SWOT, as taught in business schools, is often done qualitatively based on individuals intuitively assessing a particular competitor. The technique can and should be done using Analytics.

True competitive analysis goes far beyond SWOT. Table 3 shows the results of a survey of the use and effectiveness of CI analysis techniques.

Since this table was compiled, an important new source of competitive data has come to the forefront. That is the analysis of social media data. People do put things on social media (e.g., Facebook, Twitter) that they would not put in writing in e-mail or other forms.

Business Intelligence, Table 3 CI Analysis Tools

Tool	Percent Using	Tool	Effectiveness Percentage
Competitor profiles	88.9	SWOT analysis	63.1
Financial analysis	72.1	Competitor profiles	52.4
SWOT analysis	55.2	Financial analysis	45.5
Scenarios	53.8	Win/loss analysis	31.4
Win/loss analysis	40.4	Gaming	21.9
Gaming	27.5	Scenarios	19.2
Conjoint analysis	25.5	Conjoint analysis	15.8
Simulation	25.0	Simulation	15.4

Source: Powell and Allgaier (1998)
Note: The two columns of table on the *left* shows the percentage of respondents using the technique. The two columns of the table on the *right*, which list the same techniques, shows the percentage of those who believe the technique is effective.

Some companies that practice competitive analysis realize that just as they gather data about competitors, competitors are likely to gather data about them. They therefore try to protect their own information by becoming secretive about their plans. They control their press releases, approve speeches by their executives, provide security training for their employees, and more to avoid leaks about their intentions.

References

Davenport, T. H., & Harris, J. G. (2007). *Competing on analytics: The new science of winning*. Boston: Harvard Business School Press.

Davenport, T. H., Harris, J. G., & Morison, R. (2010). *Analytics at work: Smarter decisions, better results*. Boston: Harvard Business School Press.

Drucker, P. (1999). Beyond the information revolution. *The Atlantic, 284*(4), 47–54.

Eckerson, W. W. (2006). *Performance dashboards: Measuring, monitoring, and managing your business*. New York: Wiley.

Glenn, J. C., & Gordon, T. J. (2008). *Futures research methodology, version 3.0*. The millennium project, Washington, DC.

Gray, P., & Watson, H. (1998). *Decision support in the data warehouse*. Prentice Hall, NJ.
Howson, C. (2008). *Successful business intelligence: Secrets to making BI a killer app*. New York: McGraw Hill.
Lustig, I., Dietrich, B., Johnson, C., & Dzekian, C. (2010, November-December). The analytic journey. *Analytics*.
Powell, T., & Allgaier, C. (1998). Enhancing sales and marketing effectiveness through competitive intelligence. *Competitive Intelligence Review, 9*(4), 29–41.
Power, D. J., (2005). *Decision support systems: Frequently asked questions*. New York: iUniverse.
Sabherwal, R., & Becerra-Fernandez, I. (2011). *Business intelligence: Practices, technologies, and management*. New York: Wiley.
Siegel, J. (2010 September-October). Business intelligence & modeling systems synergy. *Analytics*.
Skriletz, R. (2002 April). New directions for business intelligence. *Information Management*.

Busy Period

A time interval that starts when all the servers of a queueing system become busy and ends when at least one server becomes free. May also refer to a time interval that starts when a previously completely idle system begins serving any customer and ends when the system becomes idle again. The two definitions (sometimes distinguished as full and partial busy periods, respectively) coincide for a single-server queue.

See

► Queueing Theory

C

Calculus of Variations

Stephen G. Nash
George Mason University, Fairfax, VA, USA

Introduction

The calculus of variations is the grandparent of mathematical programming. From it came such concepts as duality and Lagrange multipliers. Many central ideas in optimization were first developed for the calculus of variations, then specialized to nonlinear programming, all of this happening years before linear programming came along.

The calculus of variations solves optimization problems whose parameters are not simple variables, but rather functions. For example, how should the shape of an automobile hood be chosen so as to minimize air resistance? Or, what path does a ray of light follow in an irregular medium? The calculus of variations is closely related to optimal control theory, where a set of controls are used to achieve a certain goal in an optimal way. For example, the pilot of an aircraft might wish to use the throttle and flaps to achieve a particular cruising altitude and velocity in a minimum amount of time or using a minimum amount of fuel. The modern world is full of devices designed using optimal control — in cars, elevators, heating systems, stereos, etc.

Brachistochrone Problem

The calculus of variations was inspired by problems in mechanics, especially the study of three-dimensional motion. It was used in the 18th and 19th centuries to derive many important laws of physics. This was done using the Principle of Least Action. Action is defined to be the integral of the product of mass, velocity, and distance. The Principle of Least Action asserts that nature acts so as to minimize this integral. To apply the principle, the formula for the action integral would be specialized to the setting under study, and then the calculus of variations would be used to optimize the integral. This general approach was used to derive important equations in mechanics, fluid dynamics, and other fields.

The most famous problem in the calculus of variations was posed in 1696 by John Bernoulli. It is called the Brachistochrone ("least time") problem, and asks what path a pellet should follow to drop between two points in the shortest amount of time, with gravity the only force acting on the pellet. The solution to the Brachistochrone problem can be found by solving

$$\underset{y(t)}{\text{minimize}}\ \frac{1}{\sqrt{2g}} \int_{t_1}^{t_2} \sqrt{\frac{1+y'(t)^2}{y(t)}}\,dt$$

where g is the gravitational constant. If this were a finite-dimensional problem then it could be solved by setting the derivative of the objective function equal to zero, but seventeenth-century mathematics did not know how to take a derivative with respect to a function.

The Brachistochrone problem was solved at the time by Newton and others, but the general techniques that inspired the name calculus of variations were not developed until several decades

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

later. The first major results were obtained by Euler in the 1740s. He considered various problems of the general form

$$\underset{y(t)}{\text{minimize}} \int_{t_1}^{t_2} f(t, y(t), y'(t))dt.$$

The Brachistochrone problem is of this form. Euler solved these problems by discretizing the solution $y(t)$ — approximating the solution by its values at finitely many points. This gave a finite-dimensional problem that could be solved using the techniques of calculus. Euler then took the limit of the approximate solutions as the number of discretization points tended to infinity. This approach was difficult and restrictive, because it had to be adapted to the specifics of the problem being solved, and because there were restrictions on the types of problems for which it was successful.

Far more influential was the approach of Lagrange. He suggested that the solution be perturbed or varied from $y(t)$ to $y(t) + \varepsilon\, z(t)$, where ε is a small number and $z(t)$ is some arbitrary function that satisfies $z(t_1) = z(t_2) = 0$. For the Brachistochrone problem this latter condition ensures that the perturbed function still represents a path between the two points.

If $y(t)$ is a solution to the problem

$$\underset{y(t)}{\text{minimize}} \int_{t_1}^{t_2} f(t, y(t), y'(t))dt,$$

then $\varepsilon = 0$ will be a solution to

$$\underset{\varepsilon}{\text{minimize}} \int_{t_1}^{t_2} f(t, y(t) + \varepsilon z(t), y'(t) + \varepsilon z'(t))dt$$

This observation allowed Lagrange to convert the original infinite-dimensional problem to a one-dimensional problem that could be analyzed using ordinary calculus. Setting the derivative of the integral with respect to $\in$ equal to zero at the point $\varepsilon = 0$ leads to the equation

$$\frac{d}{dt}\frac{\partial f}{\partial y'} - \frac{\partial f}{\partial y} = 0.$$

This final condition is a first-order optimality condition for an unconstrained calculus-of-variations problem. It was first discovered by Euler, but the derivation here is due to Lagrange.

The name "calculus of variations was chosen by Euler and was inspired by Lagrange's approach in varying the function $y(t)$. The optimality condition is stated as the first variation must equal zero by analogy with the condition $f'(x) = 0$ for a one-variable optimization problem. Euler was so impressed with Lagrange's work that he held back his own papers on the topic so that Lagrange could publish first, a magnanimous gesture by the renowned Euler to the then young and unknown Lagrange.

There are additional first-order optimality conditions for calculus of variations problems. The theory is more complicated than for finite-dimensional optimization, and the necessary and sufficient conditions for an optimal solution were not fully understood until the 1870s, when Weierstrass studied this topic. A discussion of this theory can be found in Gregory and Lin (1992).

Multipliers

Constraints can be added to problems in the calculus of variations just as in other optimization problems. A constraint might represent the principle of conservation of energy, or perhaps that the motion was restricted in some way, for example that a planet was traveling in a particular orbit around the sun.

Both Euler and Lagrange considered problems of this type, and both were led to the concept of a multiplier. In the calculus of variations the multiplier might be a scalar (as it is in finite-dimensional problems) or, depending on the particular form of the constraint, it might be a function of the independent variable t. They have come to be called Lagrange multipliers; but, as with the optimality condition, Euler discovered them first.

In his book *Mécanique Analytique*, Lagrange includes an interpretation of the multiplier terms. He writes that they can be considered as representing the moments of forces acting on the moving particle, and serving to keep the constraints satisfied. This point of view is the basis for duality theory, although Lagrange does not seem to have followed up on this idea.

Duality

Duality theory did not become fully developed until early in this century, with many of the important steps

coming from the calculus of variations. At first there were only isolated examples of duality. That is, someone would notice that a pair of problems — one a maximization problem, one a minimization problem — would have optimal solutions that were related to each other. An early example of this type was published in 1755, and is described in Kuhn (1991). In the nineteenth century various other examples were noticed, such as the relationship between currents and voltages in an electrical circuit. Gradually it was understood that duality was not an accidental phenomenon peculiar to these examples but rather a general principle that applied to wide classes of optimization problems. By the 1920s techniques had been developed for obtaining upper and lower bounds on the solutions to optimization problems by finding approximate solutions to the primal and dual problems. Duality as a general idea is described in the book by Courant and Hilbert (1953).

Euler and Lagrange only considered problems with equality constraints, but later authors allowed inequality constraints as well. When specialized to finite-dimensional problems, the optimality condition is referred to as the Karush-Kuhn-Tucker condition. Kuhn and Tucker derived this result in a 1951 paper. It was later discovered that Karush had proven the same result in his master's thesis (1939) at the University of Chicago under the supervision of Bliss. There are two aspects to the result: its treatment of inequality constraints, and the assumption or constraint qualification that was used to prove it. The first idea can be traced to Weierstrass and the second to Mayer (1886), and both are outgrowths of the calculus of variations.

In the 1870s Weierstrass studied the calculus of variations and presented the results of his investigations in lectures. Weierstrass did not publish his work and it only became widely known years later through the writings of those in attendance. According to Bolza (1904), Weierstrass converted the inequality constraint

$$g(y) \leq 0$$

to an equivalent equality constraint

$$g(y) + s^2 = 0$$

using a squared slack variable s. This technique is described in many sources from 1900 onward. Bolza later became a professor at the University of Chicago, establishing a connection from Weierstrass to Bliss to Karush. Karush used this technique in his thesis.

The constraint qualification used by Karush, Kuhn and Tucker relates feasible arcs (paths of feasible points leading to the solution) and the gradients of the constraints at the solution. This same condition was used by Mayer (1886), although applied to a calculus of variations problem with equality constraints, and then in a chain of papers by various authors (including Bliss) leading to Karush's thesis. In these papers it is called a normality condition, and it is equivalent to requiring that the matrix of constraint gradients at the solution be of full rank. The implicit function theorem can be used to relate this to the condition on feasible arcs, an observation that is explicit in Mayer's work.

Concluding Remarks

The calculus of variations has influenced many areas of applied mathematics. It is a technical tool for solving optimization problems whose parameters are functions, and in this way it continues to be used in optimal control. It was the setting for the development of the most important concepts in optimization, such as duality and the treatment of constraints. And, when coupled with the Principle of Least Action, it was the vehicle for deriving the fundamental laws of physics.

See

- ▶ Control Theory
- ▶ Lagrange Multipliers
- ▶ Linear Programming
- ▶ Nonlinear Programming

References

Bliss, G. A. (1925). *Calculus of variations*. Chicago: Open Court.

Bolza, O. (1904). *Lectures on the calculus of variations*. Chicago: University of Chicago Press.

Courant, R., & Hilbert, D. (1953). *Methods of mathematical physics* (Vol. I). New York: Interscience.

Dacorogna, B. (2004). *Introduction to the calculus of variations*. London: Imperial College Press.

Goldstine, H. H. (1980). *A history of the calculus of variations from the 17th through the 19th century*. New York: Springer-Verlag.

Gregory, J., & Lin, C. (1992). *Constrained optimization in the calculus of variations and optimal control theory*. New York: Van Nostrand Reinhold.
Hestenes, M. R. (1966). *Calculus of variations and optimal control theory*. New York: John Wiley.
Kuhn, H. W. (1991). Nonlinear programming: A historical note. In J. K. Lenstra, A. H. G. Rinnooy Kan, & A. Schrijver (Eds.), *History of mathematical programming* (pp. 82–96). Amsterdam: North-Holland.
Lagrange, J. L. (1888–1889). *Oeuvres de Lagrange* (Vols. XI and XII). Paris: Gauthier-Villars.
Mayer, A. (1886). Begründung der Lagrange'schen Multiplicatorenmethode in der Variationsrechnung. *Mathematische Annalen, 26*, 74–82.

Call and Contact Centers

Vijay Mehrotra[1], Thomas A. Grossman[1] and Douglas A. Samuelson[2]
[1]University of San Francisco, San Francisco, CA, USA
[2]Infologix, Inc., Annandale, VA, USA

Introduction

All companies have direct and indirect means of contacting customers, potential customers, or other clients. The basic ways include postal mail, email, and, of course, the telephone. Of special importance and interest is the ability of a company's representatives (agents) to talk with call-in clients or called parties on a large scale, that is, via call centers. Call centers are an important channel for businesses to interact with customers and stakeholders. Such centers generate large transaction volumes and can have a significant impact on client attitudes towards a company and its products. Examples include commercial software support, outbound sales prospecting, customer service, internal company help desk services, municipal information dissemination, emergency services dispatch, and financial transaction processing.

Many call centers have expanded to become contact centers that communicate with clients and called parties through a variety of means such as voice calls, planned callbacks (sometimes through virtual queueing), voice mail, cellular text messaging, and email. Like call centers, these types of contact centers are used by organizations to provide a wide variety of services.

Historically, A. K. Erlang, by his paper, "On the rational determination of the number of circuits," written in 1924 and first published in Brockmeyer et al. (1948), is considered the founder of call center analysis. Call and contact centers (hereafter collectively referred to as centers) are a large global industry. In 2008, the U.S. had an estimated 47,000 centers and 2.7 million agents; Europe, the Middle East, and Africa had 45,000 centers and 2.1 million agents; and Canada and Latin America had 35,000 centers and 730,000 agents. Since then, the industry continued to grow rapidly worldwide.

Centers can be inbound, outbound, or blended. Inbound centers receive calls and other contacts from clients. Outbound centers, which normally rely on voice, generate calls that are usually for telemarketing or collections. Blended centers do both and typically deploy agents who perform outbound work when inbound arrival rates are low. Inbound centers provide staff based on advance predictions of call rates and duration; poor predictions can cause serious degradation in performance. Thus, inbound centers need high-quality forecasts of arrival rates and service times that are random and non-stationary.

Outbound center managers have the luxury of choosing when to initiate contact and closely map their actions to the number of agents on duty. Computers are used to generate outbound calls and are programmed to pace calls such that a called party picks up the phone just as an agent ends a call. Hence, outbound centers need predictions of the expected length of contact, and the time interval between a computer-placed call and when the called party answers the phone. Otherwise, the system generates a nuisance call by abandoning the call when the called party answers, or the called party hangs up because there is no one on the line. U.S. law prescribes penalties for generating large numbers of calls abandoned by the system.

Ongoing improvements in information technology and reductions in telecommunications costs allow multiple physical locations to be managed as a single very large virtual center, thus enhancing pooling effects. Contacts can be given complex routings depending on the client's identity, product, need, or service history. Information can be obtained from clients via interactive voice response. Centers maybe off-shored to locations with lower labor costs; they can be run in-house or outsourced to a contractor.

Contact center business and operations issues are discussed in the following surveys and related studies: Aksin et al. (2007a), reviews opportunities for OR to improve practice; Gans et al. (2003) cites 164 papers associated with call centers; an expanded Web-based bibliography by Mandelbaum includes over 450 papers, as well as case studies; and Koole and Mandelbaum (2002) survey queuing models, while L'Ecuyer (2006) surveys optimization models; multi-skill centers are reviewed by Koole and Pot (2006) and Aksin et al. (2007a, b). OR/MS models applied within this domain tend to not consider human resources issues, although these issues are reviewed by Holman (2005) and Aksin et al. (2007b).

Much of the OR/MS knowledge and research related to centers is proprietary and unpublished. Trade magazines and patent filings can be important sources of information.

Inbound Systems

Inbound centers serve clients who initiate contact with an organization to receive service. Such centers need to react to calls that arrive randomly. Typically, after initiating contact, the client is connected to an agent or placed into a queue for later connection. Upon connection, the client receives service for some random amount of time. There can be other outcomes, such as when all incoming phone lines are in use and the call is blocked (the client is given a busy signal) or the client may decide to abandon the call. In some centers, a client may be connected to another agent with different skills, or wait in that agent's queue. The client might call back or otherwise reinitiate contact if the issue was not satisfactorily resolved. See Cleveland (2006) for further details on how inbound call centers operate.

Managers of inbound call centers seek to provide high-quality client service while keeping costs under control. Cost is straightforward, with the largest expense being labor. Cost performance is generally measured using labor cost, for which agent utilization (percent of time an agent is engaged with clients) serving as a proxy. Labor costs typically constitute 60% to 80% of a call center's operating expense. Telephone (or for contact centers, the Internet) costs have been a concern when queues or service times were high. These pressures are diminishing in most developed countries due to rapidly declining telephone and Internet rates. An issue of labor costs is that centers with excessive agent utilization or low pay may experience high employee turnover with concomitant expenses for recruitment and training.

In contrast to labor cost, service quality is a more complex measure. Service quality can include issues such as agent training and professionalism, and the ability to resolve client problems on the first call. Operationally, service quality is often measured by some function of the amount of time a client waits prior to talking to an agent. As the waiting time experienced by an individual client is a random variable, performance measures are typically some function of the waiting time distribution. The two most common measures are the average client waiting time (ASA) and percentage of calls answered within a designated time, the service level (SL).

When clients hang up before talking to an agent, they are said to abandon the queue, an example of queueing's concept of reneging. An important measure of interest is the client abandonment rate (CAR). A client who abandons the queue is presumably dissatisfied, an undesirable event, but this reduces the waiting time for subsequent clients in the queue, which enhances the center's waiting time measures (Mandelbaum and Zeltyn 2007).

Many call centers are able to track whether an issue is successfully resolved by the first phone call or requires one or more follow-up calls. The metric associated with this data is known as the first call resolution (FCR) rate.

Creating Agent Schedules

Managers schedule agents into time blocks that are typically 15 minutes to one hour in length. A 24-hour center with 15-minute time blocks has 96 time blocks each day. They try to keep staff costs low while having enough agents on duty to meet quality targets. The first OR/MS application of this tradeoff was for toll booth staffing, Edie (1954).

The process for scheduling agents is typically performed in five steps, as follows:

Step 1: Forecast call arrivals. Centers use standard statistical and forecasting techniques such as regression, exponential smoothing and its variants, and the time series models of ARIMA. Difficulties in making accurate forecasts are caused by noisy data due to small time blocks. In centers that have

complex routings with multiple queues, each queue requires a forecast. Further, call patterns can be complex in that those that are blocked or abandoned can affect future arrivals, as can call backs caused by inadequate problem resolution. Gans et al. (2003) discusses opportunities to improve center forecasts.

Step 2: Develop an estimate of operational performance measures. To plan effectively, managers must be able to estimate the impact of their decisions on operational performance measures to trade-off cost and client experience. Cost measures typically include total labor cost and average agent utilization. Client experience measures typically include ASA, SL and CAR. These measures can be estimated using analytic models or discrete-event simulation. The most common method is to apply the Erlang C formula (for determining the waiting probability in a queue) to produce estimates for ASA and SL. Arrival rates are assumed to be homogeneous and come from the Step 1 forecast.

Advanced call centers are characterized by complex routing arrangements that shunt clients among multiple queues. Skill-based routing sends calls initially to a queue that processes the most basic client inquiries and routes more challenging calls to better trained and more highly paid agents in a different queue. The task of estimating operational performance measures is thus complicated because arrivals in later queues depend upon performance, including other agent pools. Discrete-event simulation is the tool of choice in these circumstances; see Mehrotra and Fama (2003).

Step 3: Determine the number the number of agents to assign. The manager must set (or staff) the number of agents to be on duty in each time block. This is an aggregate decision, and does not consider the identities or work schedules of individual agents, which are addressed in Steps 4 and 5. Typically, the manager assigns agents to a time block to minimize total agents, while meeting a target performance measure, usually ASA or SL.

Step 4: Develop multi-time block shifts. The manager must take the number of agents assigned to each time block in Step 3 and back out a set of individual, multi-time block shifts that, in aggregate, sum up to the number of assigned agents in each time block, while honoring work rules, contract requirements, and labor laws. This can produce an infeasible or a difficult-to-apply solution, and approximations with high cost are often required.

Step 5: Assign individual agents to each shift. The manager makes final shift schedules, that is, rosters of named agents. This creates challenges of managing total hours worked per day and week to conform to labor laws, as well as managing personal preferences for work schedules and days off.

Integration of the Five Steps. The agent scheduling process is often executed step-by-step. There are obvious interactions across steps with opportunities to integrate the steps (Aksin et al., 2007a, b). Avramidis et al. (2009) show that integrating the staffing and scheduling steps in a center with skill-based routing can lead to better results. Cezik and L'Ecuyer (2008) used linear programming combined with simulation for a center with skills-based routing.

Outbound Systems

In outbound systems, a computer automatically calls designated parties from a given list. The computer recognizes and processes busy signals, no-answers, and telephone company messages. Answered calls are routed to call center agents. Typically, the computer predicts when agents will become free and dials in anticipation of agent availability, thereby reducing the time agents wait between connections.

A key analytical challenge is to determine the pacing, that is, when to dial the next call. If the pacing is too slow, agent time is wasted. If the pacing is too fast, a called party answers when no agent is available, creating a nuisance for the called party (who usually hangs up), a wasted expense for the system. Research in this area is mostly proprietary; there is scant research literature. The solution resulted in the first U.S. patent based on queueing theory (Samuelson 1989). This method estimates service durations, times from dialing to answer, and proportions of dial attempts that result in answers, and uses these statistics, updated frequently, to synchronize dialing attempts to finish shortly after predicted agent service completions (Samuelson 1999). Other patents, such as David (1997), expand and extend this method.

Other proprietary procedures establish call centers based on cloud computing. This approach drastically reduces facility costs, as agents can work from home. It also presents a new version of the predictive dialing problem: the situation is more complex and more subject to quick changes, but the huge computational resources readily available can be employed to do massive parallel simulations in real time to compute the required predictive parameters (Kaiser-Nyman et al. 2011).

Blended Systems

Blended call centers allow agents to be switched in real time between inbound and outbound calls. Bhulai and Koole (2003) discuss a queueing model which yields a threshold policy for assigning agents to outbound calls. Deslauriers et al. (2007) provide a set of Markov chain models for a call center where outbound agents can be diverted to serve inbound calls. Call center managers believe that frequent switching between inbound and outbound calls degrades agent performance for both types of calls, and common practice is to make reassignments for blocks of time rather than call by call. This added constraint makes the performance modeling and scheduling problem substantially more difficult.

Operational Trends and Research Opportunities

Forecasting and Workload Requirements

Some traditional call center assumptions have been questioned by OR/MS researchers, e.g. (Aksin et al. 2007a). One area concerns replacing the standard point-forecast of arrival rates for a short-time block with a stochastic forecast. It is possible to relax the assumption of independent time block call arrivals and model correlation of arrivals across time blocks. More general assumptions on arrival rates can affect the scheduling and rostering problems, see Steckley et al. (2009), Robbins and Harrison (2010), and Gans et al. (2009). Bassamboo et al. (2009) proposed a methodology for capacity planning and dynamic system control in the presence of random arrival rates and multiple inbound call types.

Also, significant research attention has been paid to developing and applying advanced statistical techniques to call center arrival forecasts, with many of these approaches being used to generate not only a point estimate, but also distributional forecasts. Channouf et al. (2007) tested different forecasting models for an emergency medical system. Weinberg et al. (2007) provides a model for forecasting for the short-time blocks commonly found in practice. Avramidis et al. (2004) examined how call volumes correlate across time blocks within a day, and suggested that call arrival data from early in the day can be used to update forecasts for later in the day. Shen and Huang (2008) developed a singular value decomposition model for updating same-day forecasts based on early data, and show it to be superior to benchmarks commonly used in practice. Soyer and Tarimcilar (2008) applied a Bayesian approach for modeling and analyzing call center arrival data.. Aldor-Noiman et al. (2009) developed a Gaussian mixed-model framework that allows for exogenous variables to model the contribution of specific events to forecasted call volumes.

Scheduling

Call center workforce scheduling is more complex than shift scheduling for many other service delivery organizations, such as hospitals (nurses) or transportation (bus drivers), because of the possible need to shift workload quickly to match skills required by the incoming customer to skills of the available agents. That is, call center workforce scheduling decisions are dynamic. As updates on call arrivals and agent availability become available, short-term forecasts and agent schedules can be adjusted. Mehrotra et al. (2010) developed a methodology for intra-day forecast and schedule updating, while Gans et al. (2009) suggest a stochastic-programming model with recourse to account for both random arrival rates and intra-day schedule updates.

Resource Acquisition

Call center resource acquisition is an important and continuing area of interest. Additional research is needed on long-term forecasting, personnel planning for general multi-skill call centers in the presence of both learning and attrition (Ryder et al. 2008), and for complex networks of service providers (Aksin et al. 2007a, Section 2.2). Companies routinely outsource

call center operations to third party service providers. Ren and Zhou (2008), and Milner and Olson (2008) explore issues associated with establishing and managing these relationships.

Use of Real-Time Data

Call center models generally have had to assume that agents' service times are identically distributed for a given class of client. This was due to limited computational resources that necessitated updating parameters at intervals (ten minutes is common) much longer than the typical call duration. Outbound models have similar limitations with respect to the proportion of called parties who answer. Actual call center data, however, indicate persistent differences among agent service times, even for probabilistically identical clients, and runs of high or low proportions of good contacts and of live answers. Therefore, using real-time data to adjust call center operations could produce improvement in performance, although call center managers are quick to point out that efficiency must be balanced against robustness. Kaiser-Nyman et al. (2011) report significant performance improvement from methods that do use the real-time capabilities of cloud computing.

Performance of Outbound Systems

Despite the amount of time that has passed since the solution of the basic predictive dialing problem, many interesting unsolved problems remain, as research has largely concentrated on inbound systems. For systems running multiple simultaneous outbound campaigns and applying multiple predictive dialing systems in parallel, a common tactic is to switch agents from one campaign to another as answer rates change This impairs productivity if agents are switched too abruptly or too often, hence pacing that takes human factors into account would be valuable.

Balancing the utilization of agents in blended systems, where agents could serve both inbound and outbound parties, is generally done with heuristics that tend to under-optimize productivity to ensure that high-priority, high-value inbound calls always get handled quickly. Again, the available heuristics under-optimize and overlook significant human factors.

In some outbound systems, the protocol is to have the first conversation introduce playing a recorded message to a called party who agrees to listen to it, then to have an agent (not necessarily the same one who had the first conversation) conduct a second live conversation. If agents can be switched between first and second conversations quickly, there is an opportunity for greater productivity with a predictive dialing method, but the synchronization problem is quite complex. Also, again, human factors considerations may add additional constraints.

Research Data

Available operational data tend to be aggregated into time-based averages, which is problematic from a queueing science perspective. Fortunately, the Web-based DataMOCCA Project provides a clean source of high-granularity, call-based client call data from several sources that can be used to test proposed center advances in a research environment.

Call Routing

Skill-based routing, in which different agents are capable of handling different subsets of calls in an environment with multiple call types, is a major trend in the call center industry (L'Ecuyer 2006). These systems route clients to different agents depending on their needs and support the creation of a hierarchy of agents with highly skilled personnel handling only the most challenging calls. There is an opportunity for research regarding design and appropriate performance measures in such systems, and in the dependency and interaction among staffing, scheduling, and routing. When there are multiple types of calls and multiple types of agents, performance modeling, staffing, scheduling, and rostering problems all become significantly more complex, which leads to many interesting and important research problems, see Fukunaga et al. (2002) and Avramidis et al. (2009, 2010).

Concluding Remarks

Call and contact center managers do not view models and algorithms as intrinsically appealing. Successful OR solutions need to integrate tightly with a center's existing software systems for data collection, analysis, decision support, and schedule creation. The OR value proposition can extend beyond just cost savings. Managers value OR professionals who can reduce future call volumes using process management

techniques, such as call content analysis, that collects structured data on caller issues and performs Pareto analysis to direct the organization to improve product quality, user manuals, and agent training (Mehrotra and Grossman 2009). Managers also value reduced service time, reduced labor headaches from improved scheduling, and, in some centers, sales made or clients retained. In addition to the technical aspects of the subject, there is room for more study of how to assess and address the business needs.

See

- ▶ Communications Networks
- ▶ Forecasting
- ▶ Manpower Planning
- ▶ Networks of Queues
- ▶ Queueing Theory
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Aksin, Z., Armony, M., & Mehrotra, V. (2007). The modern call center: A multi-disciplinary perspective on operations management research. *Prodution and Operations Management, 16*(6), 665–688.

Aksin, Z., Karaesmen, F., & Ormeci, E. (2007). A review of workforce cross-training in call centers from an operations management perspective. In D. Nembhard (Ed.), *Workforce cross training handbook*. Boca Raton, FL: CRC Press.

Aldor-Noiman, S., Feigin, P., & Mandelbaum, A. (2009). Workload forecasting for a call center: Methodology and a case study. *The Annals of Applied Statistics, 3*, 1403–1447.

Avramidis, A., Chan, W., Gendreau, M., L'Ecuyer, P., & Pisacane, O. (2010). Optimizing daily agent scheduling in a multiskill call center. *European Journal of Operational Research, 200*, 822–832.

Avramidis, A., Chan, W., & L'Ecuyer, P. (2009). Staffing multi-skill call centers via search methods and a performance approximation. *IIE Transactions, 41*, 483–497.

Avramidis, A., Deslauriers, A., & L'Ecuyer, P. (2004). Modeling daily arrivals to a telephone call center. *Management Science, 50*(7), 896–908.

Bassamboo, A., Harrison, J. M., & Zeevi, A. (2009). Pointwise stationary fluid models for stochastic processing networks. *Manufacturing and Service Operations Management, 11*(1), 70–89.

Bhulai, S., & Koole, G. (2003). A queueing model for call blending in call centers. *IEEE Transactions on Automatic Control, 48*(8), 1434–1438.

Brockmeyer, E., Halstrom, H., & Jensen, A. (Eds.). (1948). *The life and works of A. K. Erlang*. Copenhagen: The Copenhagen Telephone Company.

Cezik, M., & L'Ecuyer, P. (2008). Staffing multiskill call centers via linear programming and simulation. *Management Science, 54*(2), 310–323.

Channouf, N., L'Ecuyer, P., Ingolfsson, A., et al. (2007). The application of forecasting techniques to modeling emergency medical system calls in Calgary, Alberta. *Health Care Management Science, 10*(1), 25–45.

Cleveland, B. (2006). *Call center management on fast forward: Succeeding in today's dynamic inbound environment*. Colorado Springs (CO): ICMI Press.

David, J. (1997). Outbound call pacing method which statistically matches the number of calls dialed to the number of available operators. *U.S. Patent 5,640,445*. Washington, DC: U.S. Patent Office.

Deslauriers, A., L'Ecuyer, P., Pichitlamken, J., et al. (2007). Markov chain models of a telephone call center with call blending. *Computers and Operations Research, 34*(6), 1616–1645.

Edie, L. (1954). Traffic delays at toll booths. *Journal of the Operations Research Society of America, 2*(2), 107–138.

Fukunaga, A., Hamilton, E., Fama, J., Andre, D., Matan, O., & Nourbakhsh, I. (2002). Staff scheduling for inbound call centers and customer contact centers. In R. Dechter., M. Kearns, & R. Sutton (Eds.). 18th National Conference on Artificial Intelligence; July 28 – August 1, 2002, Edmonton, Alberta. Menlo Park (CA): American Association for Artificial Intelligence, 822–829.

Gans, N., Koole, G., & Mandelbaum, A. (2003). Telephone call centers: Tutorial, review and research prospects. *Manufacturing and Service Operations Management, 5*(2), 79–141.

Gans, N., Shen, H., Zhou, Y.-P., et al. (2009). *Parametric stochastic programming for call-center workforce scheduling*. Philadelphia: Wharton School, University of Pennsylvania.

Holman, D. (2005). Call centers. In D. Holman, T. D. Wall, C. W. Clegg, et al. (Eds.), *The essential of the new workplace: A guide to the human impact of modern work practices*. New York: John Wiley & Sons.

Ingolfsson, A., Akhmetshina, E., Budge, S., et al. (2007). A survey and experimental comparison of service level approximation methods for non-stationary M/M/s queueing systems. *INFORMS Journal on Computing, 19*, 201–214.

Kaiser-Nyman, M., Samuelson, D. A., & Swieskowski, B. (2011). Predictive dialing system. *U.S. Provisional Patent No. 61/564,756*. Washington, DC: U.S. Patent Office.

Koole, G., & Mandelbaum, A. (2002). Queueing models of call centers: An introduction. *Annals of Operations Research, 113*(1–4), 41.

Koole, G., & Pot, A. (2006). *An overview of routing and staffing algorithms in multi-skill customer contact centers*. Working paper, Amsterdam: Department of Mathematics, Vrije Universiteit Amsterdam.

L'Ecuyer, P. (2006). Modeling and optimization problems in contact centers. *Proceedings of the 3rd International Conference on the Quantitative Evaluation of Systems (QEST 2006)*. September 11–14, 2006. University of California, Riverside. Washington, DC: IEEE Computing Society, 145–154.

Mandelbaum, A., & Zeltyn, S. (2007). Service engineering in action: The Palm/Erlang-A queue, with applications to call centers. In D. Spath & K.-P. Fähnrich (Eds.), *Advances in services innovations* (pp. 17–48). Berlin-Heidelberg: Springer.
Mehrotra, V., & Fama, J. (2003). Call center simulation modeling: Methods, challenges, and opportunities. In S. Chick., P. J. Sánchez., D. Ferrin, & D. J. Morrice (Eds.), *Proceedings of the 2003 Winter Simulation Conference*, pp. 135–143.
Mehrotra, V., & Grossman, T. A. (2009). OR process skills transform an out of control call center into a strategic asset. *Interfaces, 39*(4), 346–352.
Mehrotra, V., Ozluk, O., & Saltzman, R. (2010). Intelligent procedures for intra-day updating of call center agent schedules. *Production and Operations Management, 19*(3), 353–367.
Milner, J., & Olson, T. (2008). Service-level agreements in call centers: Perils and prescriptions. *Management Science, 54*(2), 238–252.
Ren, Z., & Zhou, Y. (2008). Call center outsourcing: Coordinating staffing level and service quality. *Management Science, 254*(2), 369–383.
Robbins, T., & Harrison, T. (2010). A stochastic programming model for scheduling call centers with global service level agreements. *European Journal of Operational Research, 207*(3), 1608–1619.
Ryder, G., Ross, K., & Musacchio, J. (2008). Optimal service policies under learning effects. *International Journal of Services and Operations Management, 4*(6), 631–651.
Samuelson, D. (1989). System for regulating arrivals of customers to servers. *U.S. Patent 4,858,120*. Washington, DC: U.S. Patent Office.
Samuelson, D. (1999). Call attempt pacing for outbound telephone dialing systems. *Interfaces, 29*(5), 66–81.
Shen, H., & Huang, J. (2008). Interday forecasting and intraday updating of call center arrivals. *Manufacturing and Service Operations Management, 10*(3), 391–410.
Soyer, R., & Tarimcilar, M. (2008). Modeling and analysis of call center arrival data: A Bayesian approach. *Management Science, 54*, 266–278.
Steckley, S., Henderson, S., & Mehrotra, V. (2009). Forecast errors in service systems. *Probability in the Engineering and Informational Sciences, 23*(2), 305–332.
Taylor, J. (2008). A comparison of univariate time series methods for forecasting intraday arrivals at a call center. *Management Science, 54*(2), 253–265.
Weinberg, J., Brown, L., & Stroud, J. (2007). Bayesian forecasting of an inhomogeneous Poisson process with applications to call center data. *Journal of the American Statistical Association, 102*, 1185–1199.

Call Priorities

A strategy for handling calls with varying degrees of urgency. Many emergency services have instituted formal procedures for responding differently (e.g., with and without flashing lights and sirens) to calls depending upon priority level.

See

► Emergency Services

Candidate Rules

A group of rules that the inference engine has determined to be of immediate relevance at the present juncture in a reasoning process. These rules will be considered according to a particular selection order and subject to a prescribed degree of rigor.

See

► Artificial Intelligence

Capacitated Transportation Problem

A version of the transportation problem in which upper bounds are imposed on some or all of the flows between origins and destinations.

See

► Transportation Problem

Capital Budgeting

Reuven R. Levary
Saint Louis University, St. Louis, MO, USA

Introduction

The desired end result of the capital budgeting process is the selection of an optimal portfolio of investments

from a set of alternative investment proposals. An optimal portfolio of investments is defined as the set of investments that makes the greatest possible contribution to the achievement of the organization's goals, given the organization's constraints. The constraints faced by a corporation in the capital budgeting process can include limited supplies of capital or other resources as well as dependencies between investment proposals. A dependency occurs if two projects are mutually exclusive, acceptance of one requires rejection of the other, or if one project can be accepted only if another is accepted. Assuming that the organizational goals and constraints can be formulated as linear functions, the optimal set of capital investments can be found using linear programming (LP).

Capital Budgeting Under Capital Rationing

Capital rationing is a constrained capital budgeting problem in which the amount of capital available for investment is limited.

Pure Capital Rationing, with No Lending or Borrowing Allowed — Consider a firm that has an opportunity to invest in several independent projects. It is assumed that both the future cash flows associated with each project and the firm's future cost of capital can be forecast. These forecasts enable calculation of the net present value for each project, assuming that the firm expects to be affiliated with the projects for a period of N years. It is also assumed that the firm has a given fixed budget for funding the projects for each of the N years, with both the budget and the cost of capital in future periods being unaffected by investments made in previous periods. Finally, it is assumed that any portion of the budget not used in one year cannot be carried over to future years.

The basic model for capital budgeting under pure capital rationing is as follows:

$$\text{maximize} \quad \sum_{i=1}^{M} P_i x_i \tag{1}$$

$$\text{subject to} - \sum_{i=1}^{M} f_{it} x_i \leq b_i \quad \text{for } t = 1, 2, \ldots, N \tag{2}$$

$$0 \leq x_i \leq 1 \quad \text{for i} = 1, 2, \ldots, M \tag{3}$$

where P_i is the net present value for the ith project (calculated based on forecasts of future cash flows), f_{it} is the expected cash flow for project i during year t (cash flow is defined to be positive if it is inflow and negative if it is outflow), b_t is the available budget for year t, M is the number of alternative projects and x_i is the fraction of project i to be funded.

The objective function (1) represents the total expected net present value of the investment proposals that should be funded. Constraints (2) represent restrictions on the available yearly budget. Constraints (3) ensure that no more than one project of a given type will be included in the optimal portfolio. By adding the constraint that x_i be integer for $i = 1, 2, \ldots, M$, the problem becomes an integer program. In this case, no fractional projects will be allowed; a project is either accepted or rejected. Constraints on scarce resources, mutually exclusive projects, and contingent projects can easily be added to the above model when necessary.

Capital Budgeting Where Borrowing and Lending are Allowed — In this model, the amount available for lending in a given year is the "left-over" money for that year. This amount can be carried over to the next year at a given rate of interest r. Consider the case when the interest rate for borrowing, or cost of funds, depends on the amounts borrowed. The cost of borrowing is assumed to have the shape of a step function; that is, the larger the amount borrowed, with limits, the higher the interest rate. Let r_j be the interest rate that applies to borrowing an amount greater than C_{j-1} and less than or equal to C_j. A firm will borrow at interest rate r_j if it exhausts the limits placed on its borrowing at lower interest rates.

If the firm expects to be affiliated with the proposed projects for N years, then the objective is to maximize the total related cash flows at the end of the Nth year, that is, the horizon. Let α_t and β_t be, respectively, the amount lent and the amount borrowed (at interest rate r_j) in year t. Also, let f_{it} be the cash flow in year t resulting from approval of project i. All flows in this model are current values, that is, not present values. Revenues and expenditures are defined, respectively, to be positive and negative cash flows. A given project can generate cash flows after the Nth year as well. Let $\hat{f}_i$ be the present value of total cash flows at the horizon (i.e., year N) that are expected to be generated by project i at years following year N. These flows are

discounted to year N, assuming an interest rate equivalent to the firm's weighted average cost of capital. The model is formulated as follows:

$$\text{maximize} \quad \sum_{i=l}^{M} \hat{f}_i x_i + \alpha N - \sum_{j=l}^{m} \beta_{jN} \tag{4}$$

$$\text{subject to} \quad -\sum_{i=l}^{M} f_{il} x_i + \alpha l - \sum_{j=l}^{m} \beta_{jl} \le b_l \tag{5}$$

$$-\sum_{i=1}^{M} f_{il} x_i - (1+r)\alpha_{l-1} + \alpha_l + \sum_{j=1}^{m} (1+r_j)\beta_{jl-1} - \sum_{j=1}^{m} \beta_{jl} \le b_l \quad \forall \; t = 2,\ 3,\ \ldots,\ N \tag{6}$$

$$\beta_{jt} \le C_{jt} \quad \forall \; t = 1,\ 2,\ \ldots,\ N;\ j = 1,\ 2,\ \ldots,\ m \tag{7}$$

$$0 \le x_i \le 1 \quad \forall \, i = 1,\ 2,\ \ldots,\ M \tag{8}$$

$$\alpha_t,\ \beta_{jt} \ge 0 \quad \forall \; t = 1,\ 2,\ \ldots,\ N;\ j = 1,\ 2,\ \ldots,\ m \tag{9}$$

where m represents the number of different interest rates in the supply of funds schedule. The limit on borrowing during year t, at interest rate r_j, is denoted by C_{jt}. Objective function (4) represents the total flows resulting from the proposed projects at the end of the Nth year. The first component $\sum_{i=1}^{M} \hat{f}_i x_i$ of the objective function represents the present value at the horizon of the cash flows expected to be generated by the projects in years following the horizon year N. The second component $\sum_{j=1}^{M} \beta_{jN}$ is the amount lent minus the amount borrowed during the horizon year N. Inequality (5) and inequalities (6) represent the constraints on the available budget for a given year. The limits on borrowing are represented by constraints (7). This model can be extended by adding constraints on scarce resources and by incorporating mutually exclusive and contingent projects when applicable.

Fractional Projects

All LP models can result in an optimal portfolio of projects composed of fractional projects. Weingartner (1967) showed that the number of fractional projects in the optimal solution set of the basic LP model [described by relations (1)–(3)] cannot exceed the number of time periods for which constraints are imposed. Additional constraints such as mutual exclusion, contingency, and scarce resources can increase the maximum number of fractional projects. Each additional constraint increases the maximum number of fractional projects by one. Weingartner (1967) also showed that the number of fractional projects in the optimal solution of the model where borrowing and lending are allowed is no larger than the number of time periods during which the firm does not lend or borrow money.

Because solutions to LP models can include fractional projects, these models are only an approximation of the exact solution. The exact solution can be obtained by applying integer programming solution procedures. The fractions of mutually exclusive projects, which can be the solution of an LP model, may have a useful interpretation. Fractional projects may suggest the possibility of a partnership. For example, one might interpret the decision to fund the expenses of building a fraction of a shopping center to mean that it would be beneficial for the company to engage in a partnership arrangement.

Dual Linear Programming and Capital Budgeting

Consider the basic model for capital budgeting under pure capital rationing formulated by relations (1)–(3). To evaluate the profitability of various projects, a discount factor must be incorporated into the capital budgeting analysis. Define d_t as the discount factor for period t: $d_t = (1 + r_t)^{-1}$ where r_t is the interest rate at period t. The net present value for project i is

$$P_i = \sum_{t=l}^{N} f_{it}\, d_t. \tag{10}$$

Substitution of Equation (10) into (1) results in the following formulation, called Problem **P**:

$$\text{maximize} \quad Z = \sum_{i=l}^{M} \sum_{t=l}^{N} f_{it}\, d_t\, X_i$$

$$\text{subject to (2) and (3).} \quad (\mathbf{P})$$

Let y_t be the dual variable associated with the budget constraint for year t. The value of y_t at the optimal solution, y_t^*, represents the increase in the total combined net present value of the projects that results from an addition of $1 to the budget for year t.

Assume that V dollars are added to the budget in period t. This results in an increase of the net present value (the objective function) by $v \times y_t^*$. The net present value of v is $v \times d_t$. This implies that the discount factor d_t should be equal to the dual variable y_t^* at the optimal solution (Baumol and Quandt 1965). Problem **P** is called consistent if its optimal solution has the property $d_t = y_t^* \quad \forall t$. A solution to a capital budgeting problem under pure capital rationing where dual variables do not equal the discount factor is not optimal. Therefore, such a problem is inconsistent.

Ananalysis of consistent solutions helps clarify the relationship between Capital budgeting discount factors in discount factors and dual variables, as well as the choice of an objective function. Several properties of consistent solutions were summarized by Freeland and Rosenblatt (1978) and are:

1. The value of the objective function of Problem **P** equals zero if there are no upper bounds on the decision variables (i.e., in the case when the X_i are not restricted to be less than one).
2. When the value of the objective function is zero, the only way to obtain a consistent solution is by having all discount factors equal zero. This is a meaningless situation.
3. To ensure a meaningful consistent solution, the decision variables must have upper bounds. Furthermore, some projects must be fully accepted.
4. For a consistent solution to be meaningful, the optimal value of the objective function must be positive and the budget vector must include both positive and negative components.
5. If unused funds cannot be carried forward, the discount factor in period t may exceed the discount factor in period $t + 1$.

Finding the "Right" Discount Factors

Because different optimal solutions to Problem **P** are obtained for various values of the discount factor, it is necessary to find the right discount factor for the pure capital rationing case before Problem **P** is solved. Freeland and Rosenblatt (1978) reported that most of the proposed iterative procedures for finding the right discount factors described in the literature do not work properly. Problems involved in finding the right discount factors are avoided by using horizon models, such as (4)–(9). Center for Naval Analyses (CNA) origin of The horizon value of the model where borrowing and lending are allowed is $\alpha_N - \sum_{j=1}^{M} \beta_{jN}$ [see relation (4)] when there are no cash flows beyond the horizon. In this case, no discount rate is used in maximizing the horizon value and therefore the problem of finding the right discount factor is irrelevant. In the case where there are cash flows beyond the horizon, management must estimate the respective discount rates using financial and economic forecasting. The calculation of these estimates is external to the LP models used in capital budgeting decisions, and therefore is not linked to the solution procedure of the LP model.

Alternative Capital Budgeting Models

Some capital budgeting problems have multiple objectives. Such problems can be formulated as goal programming problems. In many cases, the values of variables affecting the cash flows of the projects are not known with certainty. Such variables include future interest rates, length of useful economic lives, and salvage values. Computer simulation can be used to handle the uncertainty surrounding capital budgeting decisions (Levary and Seitz 1990). Simulation can also be used to analyze the risk consequences of various capital budgeting alternatives. Decision tree analysis is a widely used method for analyzing risk associated with a single investment alternative (Levary and Seitz 1990). Expected return on investments can be adjusted for risk using the capital asset pricing model (CAPM). CAPM was generalized by Richard (1979) to include environmental uncertainty.

Applications of chance-constrained programming to capital budgeting problems have been reported in the literature. Byrne et al. (1967, 1969) incorporated payback and liquidity constraints into chance-constrained programming models for capital budgeting. The payback is represented in these models in the form of chance-constraints that filter acceptable from unacceptable risks. The liquidity constraints handle risks related to situations such as unplanned demand for cash and unplanned technological breakthroughs. Hillier (1969) formulated the net cash flows in each time period of a capital budgeting model as probabilistic constraints.

The objective function in the model is to maximize the expected utility of the shareholders at the horizon period. Näslund (1966) extended the horizon model [relations (4)–(9)], by including risks. Näslund assumed that the yearly cash flows were independent, normally distributed random variables having known means and standard deviations. He also assumed that no other random variables existed in his model. The adjusted model is a chance-constrained programming model. Näslund developed a deterministic equivalent to his chance-constrained programming model.

Relationships among investments contribute to portfolio risk and can be measured by covariances. Quadratic programming models for capital budgeting can be used in situations where the covariances between returns of various projects can be estimated. Various characteristics of a specific capital budgeting problem, like tax consequences, can be modeled using mathematical programming.

See

- ▶ Chance-Constrained Programming
- ▶ Goal Programming
- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Portfolio Theory: Mean-Variance Model
- ▶ Quadratic Programming

References

Baumol, W. J., & Quandt, R. E. (1965). Investment and discount rates under capital rationing — A programming approach. *Economic Journal, 75*(298), 317–329.

Byrne, R., Charnes, A., Cooper, W. W., & Kortanek, K. (1967). A chance-constrained approach to capital budgeting with portfolio type payback and liquidity constraints and horizon posture controls. *Journal of Financial and Quantitative Analysis, 2*(4), 339–364.

Byrne, R. F., Charnes, A., Cooper, W. W., & Kortanek, K. O. (1969). A discrete probability chance-constrained capital budgeting model-I. *Opsearch, 6*(3), 171–198.

Freeland, J. R., & Rosenblatt, M. J. (1978). An analysis of linear programming formulations for the capital rationing problems. *The Engineering Economist, 23*(Fall), 49–61.

Hillier, F. S. (1969). *The evaluation of risky interrelated investments*. Amsterdam: North Holland.

Levary, R. R., & Seitz, N. E. (1990). *Quantitative methods for capital budgeting*. Cincinnati, OH: South-Western Publishing.

Näslund, B. (1966). A model of capital budgeting under risk. *Journal of Business, 39*(2), 257–271.

Richard, S. F. (1979). *A generalized capital asset pricing model* (Studies in the management sciences, Vol. 11, pp. 215–232). Amsterdam: North Holland.

Seitz, N., & Ellison, M. (2005). *Capital budgeting and long-term financing decisions* (4th ed.). Hillsdale, IL: Dryden Press.

Weingartner, H. M. (1967). *Mathematical programming and the analysis of capital budgeting problems*. Chicago: Markham Publishing.

CASE

Computer-aided software-systems engineering.

See

- ▶ Systems Analysis

CDF

Cumulative distribution function.

Center for Naval Analyses

Carl M. Harris
George Mason University, Fairfax, VA, USA

Introduction

In the pre-World War II year of 1940, many scientists believed that organizing the nation's scientific research would strengthen national defense. As a result, the National Defense Research Committee (NDRC) was established by Presidential Executive Order.

The NDRC was placed under the direction of the newly created Office of Scientific Research and Development (OSRD), which reported directly to the president. NDRC's contact with British researchers indicated that studying actual operations was an essential part of any assessment process. Because the need for operations research was particularly pressing

in the area of antisubmarine warfare (ASW), the Navy created the Antisubmarine Warfare Operations Research Group (ASWORG). In 1942, comprising at first fewer than a dozen scientists, it was the first civilian group engaged in military operations research in the United States. The Center for Naval Analyses (CNA) traces its origins to ASWORG.

Today, CNA analysts provide the Navy and Marine Corps with objective studies of a wide variety of operations, systems and programs. Such studies range from the support of training and testing activities to the evaluation of new technologies and alternative force structures for top-level decision-makers. The following short history of CNA recounts the high-lights of its evolution and contribution to national security.

World War II

During the 1940s, the United States was preoccupied first with the war in Europe and then with the war in the Pacific. As soon as the United States entered the war, German submarines began to patrol the U.S. East Coast and Atlantic shipping lanes in earnest. The Navy's immediate focus was on the U-boat threat and the Battle of the Atlantic.

In Britain, Professor P.M.S. Blackett had demonstrated the value of operations research in solving military problems. Captain Wilder Baker, leader of the newly formed U.S. Navy Antisubmarine Warfare Unit in Boston, was inspired by Blackett's paper, "Scientists at the Operation Level" (see *Blackett's later work*, 1962). Baker believed that a cadre of civilian scientists could also help the U.S. Navy. He asked Professor Philip M. Morse of MIT to head such a group. ASWORG was formed in April 1942 with a mission to help defeat the German U-boats. The contract for ASWORG was administered by Columbia University, which already had an existing contract with the NDRC that focused on anti-submarine warfare.

ASWORG set a major precedent when it required its analysts to gather field data firsthand. Sending civilian experts to military commands was a delicate matter. In June 1942, the field program began when an ASWORG analyst assisted the Gulf Sea Frontier Headquarters in Miami. Shortly afterward, several analysts were assigned to the Eastern Sea Frontier in New York. The field analysts quickly became accepted; most of ASWORG's noteworthy work was achieved in the field.

In June 1942, ASWORG was assigned to the Head-quarters of Commander in Chief, U.S. Fleet (CominCh). Admiral Ernest J. King was both CominCh and the Chief of Naval Operations (CNO). The Tenth Fleet was formed in 1943 to consolidate U.S. ASW operations. In July 1943, ASWORG became part of the Tenth Fleet.

In October 1944, because of the decrease in enemy submarine activity and the increase in operations research requirements on subjects other than ASW, ASWORG was transferred from the Tenth Fleet to the Readiness Division of the Headquarters of CominCh. It was also renamed the Operations Research Group (ORG) as its analysis efforts had become more diversified.

By the end of the war, ORG had about 80 scientists whose scope of study was all forms of naval warfare. During most of World War II, about 40% of the group was assigned to various operating commands. These field analysts developed immediate, practical answers to tactical and force allocation questions important to their commands. Concurrently, they fed back practical experiences and understanding to the central Washington group, a practice still continued a half century later.

Among its many World War II contributions, ORG devised more effective escort screening plans; determined the optimum size of convoys; developed ASW tactics, such as optimum patterns and altitudes for flying AWS patrol aircraft; developed counter measures to German acoustic torpedoes and snorkeling U-boats; and contributed to the use of airborne radar.

Post-War Period

In August 1945, Admiral King, in a letter to Secretary of the Navy James V. Forrestal, recommended and requested that ORG be allowed to continue into peacetime at about 25% of its wartime size. Secretary Forrestal gave his approval shortly thereafter.

Both Admiral King and Secretary Forrestal concluded that much of ORG's unique value was due to its ability to provide an independent, scientific viewpoint to a broad range of Navy problems.

Consequently, in extending the service of ORG into peacetime, it was decided that its character could best be preserved by perpetuating the wartime arrangement through a contract with an academic institution. Such a contract was entered into with MIT in November 1945. At that time, ORG was renamed the Operations Evaluation Group (OEG), with Dr. Jacinto Steinhardt as its first director. OEG was to assist the Navy and its research laboratories in analyzing and evaluating new equipment, tactical doctrine and strategic warfare. OEG established a policy that all of its (male) analysts must spend time assigned to fleet operations, a practice that is partially maintained to this day by CNA.

After the war, OEG published several comprehensive reports on important naval operations, which included many new methodologies. Although some were originally classified secret, they later appeared in Morse and Kimball's *Methods of Operations Research*, Bernard Koopman's *Search and Screening*, and Charles Sternhell and Alan Thorndike's *Anti-submarine Warfare in World War II*. Taken together, these reports provided a record of vital lessons learned in World War II, as well as important operations research methods. With the Korean War and the intensification of the Cold War, the role of analysis in defense planning expanded in the 1950s. Once the Soviets had detonated their first thermonuclear device, the United States had to revise its thinking on many critical defense issues. As the consequences of nuclear war loomed and the cost of military preparedness escalated, the government, more than ever, needed reliable scientific information on which to base its strategic decision-making.

Before the Korean War, OEG began a slow but steady buildup. By 1950, the research staff had grown to about 40. As the war began, OEG received requests for analysts from combat commands. These analysts collected data, solved tactical problems and recommended improvements in procedures, improvements that were sometimes used immediately. OEG expended its major efforts on such specific tactical problems as: selection of weapons for naval air attack on tactical targets; scheduling of close air support; analysis of air-to-air combat; naval gunfire in shore bombardment; blockade tactics; and interdiction of land transportation. By the end of the war, OEG had 60 research staff members.

After the war, OEG continued to grow, albeit slowly. Analysts participated with naval forces in all post-Korean crises. The most important changes in the nature of the group's post-Korean activity were the results of major technological advances, particularly in the field of atomic energy and guided missiles. Issues were broadened to include the possible enemy use of nuclear weapons and the effect of U.S. policies and weapon system choices on the nature of wars the United States would have to be prepared to fight. During this period, the Navy also established the Long-Range Studies Project of MIT; it was later renamed the Institute for Naval Studies (INS).

Defense Management

By the 1960s, advances in weapons technology were causing defense costs to rise dramatically, and the increasing tempo of the Vietnam War later in the decade would cause the defense budget to balloon still further. The swearing in of Secretary of Defense Robert S. McNamara in 1961 marked the beginning of a new philosophy of defense management. Emphasis began to be placed on cost as well as effectiveness. McNamara believed that integrated systems analysis throughout the defense establishment was required to achieve a balanced, affordable military structure.

In 1961, MIT established an Economics Division within OEG because the cost of weapon systems was becoming a dominant factor in military decision-making. Until 1961, the Marine Corps had only one OEG analyst. By the early 1960s, however, Marine Corps requirements for operations research had increased substantially. The Marine Corps Section of OEG was established in December 1961.

By 1962, the Secretary of the Navy wanted to consolidate the study efforts of OEG and INS and began to look for a contractor. MIT, which had managed OEG since 1945, declined an invitation to manage this proposed new enterprise. The Navy then selected the Franklin Institute to administer the contract for the new organization. In August 1962, OEG and INS were brought under the common management of a new entity, the Center for Naval Analyses (CNA).

Center for Naval Analyses

Shortly after CNA was formed, OEG (now as a division) again became involved in an actual naval operation. In October 1962, it helped the Office of the Chief of Naval Operations (OPNAV) develop plans for the naval quarantine of Cuba and assessed the effectiveness of surveillance operations.

As combat escalated in Southeast Asia, so did the number of CNA field representatives providing direct support to the naval operating forces. CNA participated in the study of many operations, such as interdiction campaigns in North Vietnam and infiltration rates in South Vietnam. Also, a large data base on war-related activities was being developed and maintained in CNA's Washington office. In August 1967, management of the CNA contract transferred from the Franklin Institute to the University of Rochester.

Because the war in Vietnam was escalating, the Navy needed more combat analysis. As a result, the Southeast Asia Combat Analysis Group (SEACAG) was established within OPNAV. Shortly thereafter, the Southeast Asia Combat Analysis Division (SEA-CAD) was established within OEG. SEACAD's role was to support SEACAG and to increase the amount of war-related analysis that CNA was performing. CNA analyzed various operations of the Southeast Asian conflict, including combat aircraft losses, interdiction, strike warfare and carrier defense, surveillance and naval gunfire support.

In the 1970s, as the war in Vietnam wound down, military budgets, forces and equipment began to deteriorate. To maintain effectiveness in the face of reduced budgets, the Navy increased its emphasis on analysis. As new systems became available, the Navy needed to determine how best to exploit their capabilities. With old systems that were already deployed, the Navy needed to develop tactics that overcame technical shortcoming.

Military Buildup

The 1980s witnessed a major buildup of U.S. forces in response to the growth of Soviet military power during the 1970s. For the Navy, this meant not only more ships and aircraft but also more emphasis on a maritime strategy and on specific concepts of operations for employing the Fleet in a global war. These efforts matured by 1987, just as Gorbachev unleashed the forces that would lead to the razing of the Berlin Wall and, ultimately, the demise of the Soviet Union.

In 1982, CNA began a major study of concepts of operations for employing the Atlantic Fleet in a global war. This work involved issues ranging from Soviet objectives and intentions in a war to actions the Navy could take to counter Soviet strategy, as well as theater-level tactics that would be executable in the face of a concerted Soviet threat. The results of this work were put into practice in 1984 by Commander, Second Fleet, who also added important tactical innovations. The resulting interaction and cooperation of Washington and the Fleet (and of CNA-Washington and the field analysts) set the tone for similar efforts at other fleet commands.

By December 1982, differences concerning the management of CNA had arisen between the Department of the Navy and the University of Rochester. The Secretary of the Navy decided to open the CNA contract to competition, and several universities and nonprofit research organizations responded. In August 1983, the Navy announced that the Hudson Institute had been awarded the contract for the management of CNA, effective October 1983.

New World Order

The 1990s ushered in an entirely new security environment. In light of the collapse of the Soviet Union and the new emphasis on Third World threats, the Navy and Marine Corps are reevaluating their structure. Unlike the threat of the Cold War era, these new threats are smaller and more diffuse. They require smaller units that can operate jointly in distant areas where the United States often has a limited number of forces and restricted access to bases. Developing these types of forces and operations is a continuing theme for defense planning in the 1990s.

During the 1980s, some significant events had solidified CNA's stature in the analytical field. Demands for CNA's analytical assistance had grown, particularly from senior Navy and Marine Corps

leaders. CNA had become more involved in critical issues and issues of concern to top-level decision-makers, and CNA's staff had increased in size and quality to meet those growing demands.

Organizationally, CNA had changed often over the years to meet the demands of a changing world and a changing military environment. In the spring of 1990, CNA's management, the Board of Overseers, the Navy, and the Hudson Institute all agreed that CNA could function as an independent organization. On October 1, 1990, CNA became independent and began operating under a direct contract with the Department of the Navy, ready to help the Navy and Marine Corps cope with the impending changes in national security policy, defense strategy, defense budgets and defense management practices.

After Iraq annexed Kuwait in August 1990, the CNO asked CNA to track and document the events in the Middle East, to analyze activities, and to develop a lessons-learned data base. CNA had up to 20 field representatives providing support to various naval commands in the Middle East, including Commander, U.S. Naval Central Command.

After the Persian Gulf War, CNA was designated the Navy's lead agency for Desert Shield/Storm data collection and analysis. The Navy believed that future force composition, systems design and budget decisions would be shaped by events of the war and the subsequent analysis. CNA led the reconstruction of Desert Shield/Storm and provided the Navy with a 14-volume report. In addition, CNA is continuing its analysis of the war and is archiving all the fleet data for the National Archives.

During Desert Storm, the value of concepts that CNA had analyzed for the Navy and Marine Corps — the Tomahawk cruise missile, the air-cushioned landing craft (LCAC), the maritime prepositioning — became evident. The Tomahawk land-attack missile was one of the high-tech "stars" of the war; the LCAC played an important role in creating fear of an amphibious assault; and maritime prepositioning allowed two brigades of Marines to deploy to the Gulf in record time.

In the 1990s, CNA's most important task was to help the Navy and Marine Corps make the transition to a post-Cold War security environment. To do this, CNA's research program plan emphasized areas of immediate importance to this transition: the new security environment, littoral operations, communications, warfare area adjustments, training and education, investment alternatives, force structure, and economies and efficiencies.

See

- ▶ Field Analysis
- ▶ Military Operations Research
- ▶ Operations Research Office and Research Analysis Corporation
- ▶ RAND Corporation

References

Blackett, P. M. S. (1962). *Studies of war*. London: Oliver and Boyd.

Center for Naval Analyses. (1993). Victory at sea: A brief history of the center for naval analyses. *OR/MS Today, 20*(2), 46–51.

Kreiner, H. W. (1992). *Fields of operations research*. Baltimore: Operations Research Society of America.

Morse, P. M., & Kimball, G. E. (1946). *Methods of operations research, OEG Report 54, Operations Evaluation Group (CNA)*. Washington, DC: U.S. Department of the Navy.

Tidman, K. (1984). *The operations research group*. Annapolis, MD: Naval Institute Press.

Certainty Equivalence

Jeffery L. Guyse
University of California, Irvine, USA

The certainty equivalent of a gamble or lottery is the sum of money for which, in a choice between the money and the gamble, the decision maker is indifferent between the two. Certainty equivalents are used to determine decision makers' attitudes toward risk, which can then be reflected in the shape of their utility functions. Certainty equivalents can also be used to order a set of alternatives. Classic examples of operationalizations of certainty equivalents used in the literature are minimum selling price, maximum buying price, and cash equivalent. Buying and selling prices may be theoretically different though, due to income effects.

By definition, the utility of the certainty equivalent must be equal to the expected utility of the gamble. With this in mind, the relationship between the certainty equivalent (CE) and the expected value (EV) of a gamble can reveal the decision maker's attitude toward risk. If CE < EV, then the individual is said to exhibit a risk-averse attitude. In this case, the difference between the expected value and the certainty equivalent (EV − CE) is known as the "risk premium" that the decision maker is willing to pay in order to avoid the risk associated with the gamble. If CE > EV, a risk-prone (or risk-seeking) attitude is displayed. Finally, if the two values are equal (CE = EV), then the decision maker is risk-neutral. By assessing CEs, decision analysts can calibrate the utility function of the decision maker to reflect risk attitude in the decision process. For a formal discussion, see Keeney and Raiffa (1976).

Certainty equivalents are also used to elicit a preference order on a set of alternatives. It is assumed that the order induced by assigning certainty equivalents reveals the true preference order of the individual. If one alternative has a higher certainty equivalent than a second alternative, one would expect the individual to choose the former over the latter when asked to make a choice between the two. The method by which the certainty equivalents are elicited has been an area of ongoing research. It was once believed that subjects could simply state their certainty equivalent to a gamble, in which case their response is known as a judged certainty equivalent.

Recent empirical studies have provided evidence that the judged certainty equivalent may not necessarily equal the true certainty equivalent elicited through a choice mechanism. Such a violation of procedure invariance is examined in the stream of research on preference reversals. Subjects provide both judged certainty equivalents and then make choices between pairs of gambles. By carefully selecting the gambles, researchers have been able to elicit judged certainty equivalents that produce an ordering on the set, while the same subject's choices results in the reverse ordering (Grether and Plott 1979; Lichtenstein and Slovic 1971, 1973; Lindman 1971; Slovic and Lichtenstein 1983). Such a pair of gambles is:

A:	0.99 Win $4.00 0.01 Win $0	B:	0.25 Win $16.00 0.75 Win $0

The expected values of the two gambles are $3.96 and $4 respectively. A large proportion of people will indicate a preference for gamble A when asked to choose between the two, yet place a higher dollar value on B (Grether and Plott 1979, p. 623).

Work by Tversky, Slovic, and Kahneman (1990) as well as Bostic, Herrnstein, and Luce (1990) has shown that these preference reversals virtually disappear when the certainty equivalents are elicited through a choice mechanism, such as the Parameter Estimation by Sequential Testing (PEST) procedure. For a review of preference reversals, see Tversky and Thaler (1990).

See

- Decision Analysis
- Lottery
- Risk
- Utility Theory

References

Bostic, R., Herrnstein, R. J., & Luce, R. D. (1990). The effect on the preference-reversal phenomenon of using choice indifference. *Journal of Economic Behavior & Organization, 13*, 193–212.

Grether, D., & Plott, C. (1979). Economic theory of choice and the preference reversal phenomenon. *The American Economic Review, 69*, 623–638.

Keeney, R. L., & Raiffa, H. (Eds.). (1976). *Decisions with multiple objectives: Preference and value trade-offs*. New York: Wiley and Sons.

Lichtenstein, S., & Slovic, P. (1971). Reversals of preference between bids and choices in gambling decisions. *Journal of Experimental Psychology, 89*, 46–55.

Lichtenstein, S., & Slovic, P. (1973). Response-induced reversals of preference in gambling: An extended replication in Las Vegas. *Journal of Experimental Psychology, 101*, 16–20.

Lindman, H. R. (1971). Inconsistent preferences among gambles. *Journal of Experimental Psychology, 89*, 390–397.

Slovic, P., & Lichtenstein, S. (1983). Preference reversals: A broader perspective. *The American Economic Review, 73*, 596–605.

Tversky, A., Slovic, P., & Kahneman, D. (1990). The causes of preference reversals. *The American Economic Review, 80*, 204–217.

Tversky, A., & Thaler, R. (1990). Anomalies: Preference reversals. *The Journal of Economic Perspectives, 4*, 201–211.

Certainty Factor

A numeric measure of the degree of certainty about the goodness, correctness, or likelihood of a variable value, an expression (e.g., premise) value, or conclusion.

See

► Expert Systems

Chain

A chain in a network is a sequence of arcs connecting a designated initial node to a designated terminal node such that the direction (orientation) of flow in the arcs is from the initial node to the terminal node.

See

► Cycle
► Markov Chains
► Path

Chance Constraint

A constraint that restricts the probability of a certain event to a prespecified range of values. Under certain conditions, chance constraints can be incorporated into mathematical-programming problems.

See

► Chance-Constrained Programming
► Linear Programming
► Stochastic Programming

Chance-Constrained Programming

A mathematical-programming problem in which the parameters of the problem are random variables and for which a solution must satisfy the constraints of the problem in a probabilistic sense. Here the usual linear-programming constraints are given as probability statements of the form $\Pr\{\sum_{j=1}^{n} a_{ij} x_j \leq b_i\} \geq \alpha_i$ for $i = 1, \ldots, m$, where the $\{\alpha_i\}$ are given constants between zero and one. Some forms of the chance-constrained programming problem can be transformed to an equivalent linear-programming problem.

See

► Linear Programming
► Stochastic Programming

References

Charnes, A., & Cooper, W. (1959). Chance-constrained programming. *Management Science, 6*, 73–79.
Prékopa, A. (1995). *Stochastic programming*. Dordrecht: Kluwer.

Chaos

A mathematical term describing a situation in which arbitrarily small variations in independent variable values can produce large variations in the dependent variable. The term is most typically used to characterize the behavior of deterministic, nonlinear, differentiable dynamic systems. The term is sometimes used to describe situations in which true mathematical chaos is not present, but where the results are similarly disturbing. The disturbing effect in battle modeling, for example, is the apparent loss of deterministic behavior.

Chapman-Kolmogorov Equations

In a parameter-homogeneous Markov chain $\{X(t)\}$ with state space S, define $p_{ij}(t)$ as the probability that $X(t+s) = j$, given that $X(s) = i$ for $s, t \geq 0$. Then, for all states i, j and index parameters $s, t \geq 0$,

$$p_{ij}(t+s) = \sum_{k \in S} p_{ik}(t) p_{kj}(s)$$

are the Chapman-Kolmogorov equations. There is a comparable definition when the state space is instead continuous.

See

- ▶ Markov Chains
- ▶ Markov Processes

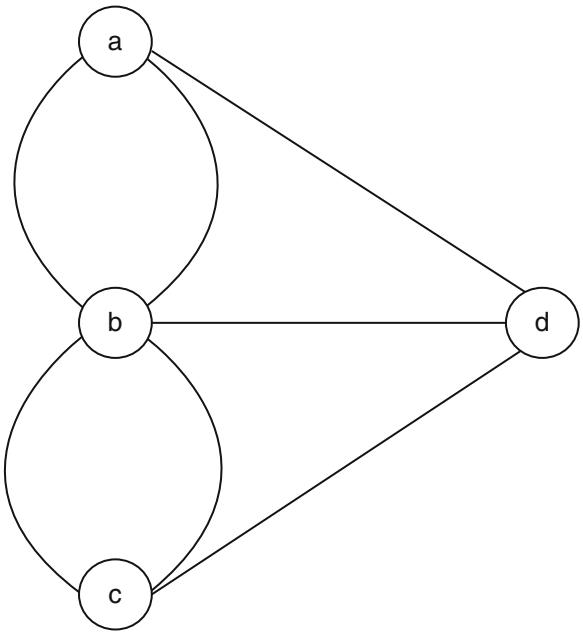
Characteristic Function

For a random variable X, the characteristic function is given by $\phi_X(t) = E[e^{itX}]$, where i denotes the imaginary number $\sqrt{-1}$.

Chinese Postman Problem

William R. Stewart Jr.
College of William and Mary, Williamsburg, VA, USA

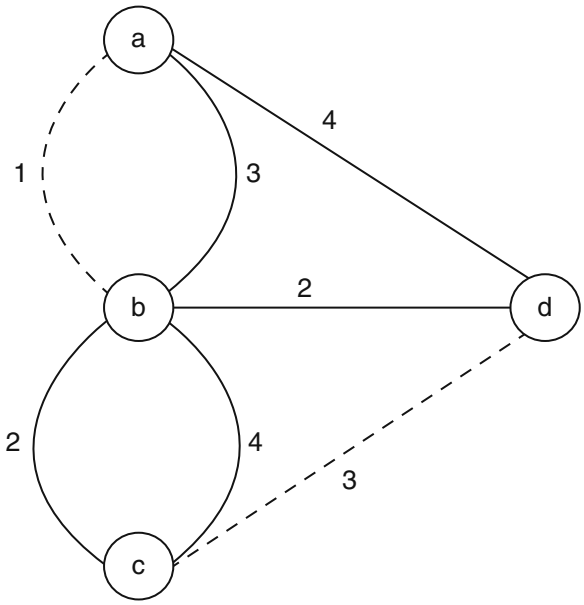
The Chinese Postman Problem acquired its name from the context in which it was first popularly presented. The Chinese mathematician Mei-Ko Kwan (1962) addressed the question of how, given a postal zone with a number of streets that must be served by a postal carrier (postman), does one develop a tour or route that covers every street in the zone and brings the postman back to his point of origin having traveled the minimum possible distance. Researchers who have followed on Kwan's initial work have since referred to this problem as the Chinese Postman Problem or CPP. In general, any problem that requires that all of the edges of a graph (streets, etc.) be traversed (served) at least once while traveling the shortest total distance overall is a CPP. Like its cousin, the traveling salesman problem, that seeks a route of minimum cost that visits every vertex of a graph exactly once before returning to the vertex of origin, the CPP has many real world manifestations, not the least of which is the scheduling of letter carriers. Such problems as street sweeping, snow plowing, garbage collection, meter reading and the inspection of pipes or cables can and have all been treated as CPPs.



Chinese Postman Problem, Fig. 1 A graph of Euler's Konigsberg bridge problem

In the following discussion, the terms tour and cycle will be used interchangeably to refer to a route on a graph that begins and ends at the same vertex and that traverses all of the edges of that graph at least once. Unless otherwise noted, the edges are assumed to be undirected (i.e., they may be traversed in either direction).

The CPP and its many variants have their roots in the origins of mathematical graph theory. The problem of finding a cycle (tour/route) on a graph which traverses all of the edges of that graph and returns to its starting point dates back to the mathematician Leonid Euler and his analysis in 1736 of a popular puzzle of that time, the Königsberg Bridge problem. Euler's problem of traversing all of the bridges of Königsberg and returning to his starting point without retracing his steps is equivalent to asking if there is a tour of the graph shown in Fig. 1 that traverses all of the edges exactly once. Euler showed that such a cycle exists in a graph if and only if each vertex in the graph has an even number of edges connecting to it or, in mathematical terms, each vertex is of even cardinality. This follows logically from the observation that, in a tour that traverses all of the edges exactly once, each vertex must be exited the same number of times it is entered. Tours that traverse each edge of a graph exactly



Chinese Postman Problem, Fig. 2 The Konigsberg bridge problem with edge costs

once are termed Euler cycles or tours, and graphs that contain an Euler cycle are appropriately called Eulerian. When costs are assigned to each of the edges, the problem of finding a minimum cost tour is a CPP.

When a graph is Eulerian, the cost of a tour is just the sum of the costs of all of the edges in the graph, and the solution to the CPP is any Eulerian tour, of which there are usually many. In general, an Eulerian tour can easily be found when one exists. When a graph has more than one odd cardinality vertex (exactly one such vertex is impossible), the CPP is the problem of finding which of the edges must be traversed more than once in order to produce a minimum cost tour. The graph shown in Fig. 1 has four vertices with odd cardinality, and a tour of this graph requires that one or more of the edges be crossed more than once. Figure 2 shows hypothetical costs on each edge, and the dashed lines indicate the edges that must be traversed twice in order to achieve a minimal cost tour. This tour will have a total cost of 23, the cost of crossing each edge once plus the cost of crossing edges (a, b) and (c, d) a second time each.

In mathematical terms, the CPP can be stated as follows: given a graph $G = \{V, E\}$, where V is a set of n vertices, E is a set of edges connecting these vertices, and each edge (i, j) connecting vertices i and j has a nonnegative cost, c_{ij}, find x_{ij}, the number of times that edge (i, j) is to be traversed from i to j so that the total cost of traversing all of the edges in E at least once is a minimum. The sum of x_{ij} and x_{ji} is the total times that the edge between vertices i and j must be traversed in an optimal tour.

$$\text{Minimize} \sum_i \sum_j c_{ij} x_{ij} \tag{1}$$

$$\text{Subject to} \sum_i x_{ik} - \sum_j x_{kj} = 0, \quad \text{for } k = 1, \ldots, n, \tag{2}$$

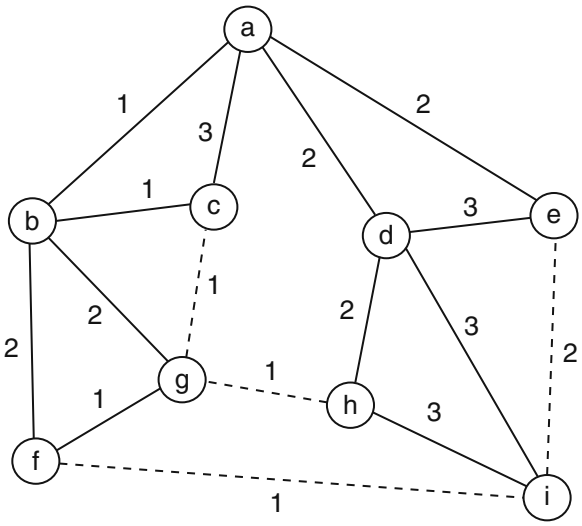
$$x_{ij} + x_{ji} \geq 1, \quad \text{for all } (i,j) \text{ and } (j,i) \in E, \tag{3}$$

$$x_{ij} \geq 0, \quad \text{and integer, for all } (i,j) \in E. \tag{4}$$

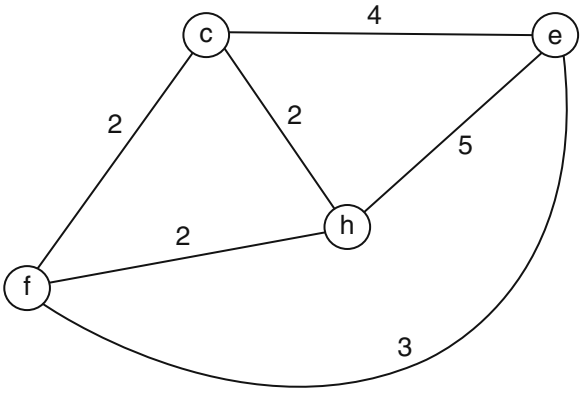
For ease of exposition, this formulation assumes that there is a maximum of one edge between any two vertices. As can be seen in the illustration in Figs. 1 and 2, this may not always be the case. However, cases where there are multiple edges between the same pair of vertices do not complicate the treatment, since those cases can easily be transformed into the form shown in (1)–(4).

As pointed out by Edmonds and Johnson (1973) and Christofides (1973), when there are odd cardinality vertices in the graph, the CPP reduces to the problem of finding a minimum cost matching among the odd cardinality vertices. A minimum cost matching on a graph is a pairing of the vertices on that graph such that each vertex is paired with exactly one other vertex and the total cost of the edges connecting the pairs is a minimum. When no edge exists between a pair of vertices, the cost of pairing them is the cost of the shortest path running between the pair. Replicating the edges that connect each pair of odd cardinality vertices in the minimum matching produces an Eulerian graph (i.e., all vertices now have even cardinality) where the total cost of all the edges, the edges in the original graph plus the edges that have been replicated as a result of the matching, is the cost of the optimal tour of the original graph.

To illustrate the general solution process, Figure 3 presents a graph with four odd cardinality vertices (c, e, f, h). None of the four vertices is directly connected to another of the four. To find the required minimum cost matching requires the construction of the graph G', shown in Fig. 4, which consists of the



Chinese Postman Problem, Fig. 3 The Graph G



Chinese Postman Problem, Fig. 4 The Graph G'

four odd cardinality vertices connected by edges whose costs are the cost of the shortest path between each pair on the original graph. The problem is then to find a minimum cost matching on the graph G'. This matching will determine which edges must be traversed twice to achieve a minimum cost tour on G. A quick inspection of G' shows that the edges (c, h) and (e, f) constitute a minimal matching on G'. The paths $(c\text{-}g\text{-}h)$ and $(e\text{-}i\text{-}f)$ on graph G in Fig. 3 correspond to this matching, and the edges along these paths will be traversed twice each in an optimal tour and are shown as dashed lines in Fig. 3.

Solving the CPP requires two operations, both of which can be performed in polynomial time. A matching of the odd cardinality vertices must be found and the corresponding edges replicated that results in an Eulerian graph. An Eulerian tour of this expanded graph must then be found. The complexity of the CPP is dominated by the complexity of solving the minimum cost matching problem, which can be solved in at most $O(n^3)$ time. Variations of the basic CPP, briefly described below, are generally not as tractable.

Variations of The Chinese Postman Problem

The CPP has many variations that can and do occur on a regular basis. In the CPP, the edges are undirected and they may be traversed in either direction. The most obvious variation of the CPP is the directed postman problem where each of the edges has a direction associated with it. This is often encountered when an edge represents a one way street in a routing problem, or an edge must be traversed twice, once in each direction, as might occur in routing a street sweeper. In this latter case, each street would be represented in the graph by two edges, one in each direction. Like the CPP, the directed postman problem can be solved in polynomial time. In a sense, it is even easier than the CPP since it requires a network flow algorithm rather than a matching algorithm.

When the graph contains a mixture of both directed and undirected edges, the problem of finding a minimum cost tour is called the mixed postman problem. The mixed postman problem has been shown to be NP-hard. The rural postman problem is a variation of the CPP where a subset of the edges in the graph must be traversed. The rural postman problem has been shown to be equivalent to a traveling salesman problem and, as such, it is also an NP-hard problem (see Lawler et al. 1985). Finally, the capacitated Chinese Postman Problem recognizes that each edge may have a nonzero demand for service and that the server (postman) may have a finite capacity for supplying service. In the general case, multiple servers must be assigned to routes such that the demands on all of the edges are met and no server is assigned a route that exceeds his capacity. This then is the problem of partitioning the edges of the graph into subsets and assigning a server (postman) to each subset in such a way that all capacity constraints are met and the total distance covered by all of the servers is a minimum. As with the directed and rural postman problems, the capacitated postman problem has been shown to be NP-hard.

See

- Combinatorics
- Computational Complexity
- Graph Theory
- Integer and Combinatorial Optimization
- Matching
- Network
- Traveling Salesman Problem
- Vehicle Routing

References

Christofides, N. (1973). The optimal traversal of a graph. *Omega, 1*, 719–732.

Edmonds, J., & Johnson, E. (1973). Matching, Euler tours, and the Chinese postman problem. *Mathematical Programming, 5*, 88–124.

Kwan, M. K. (1962). Graphic programming using odd or even points. *Chinese Mathematics, 1*, 273–277.

Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G., & Shmoys, D. B. (Eds.). (1985). *The traveling salesman problem: A guided tour of combinatorial optimization*. Chichester, UK: Wiley.

Choice Strategies

The different approaches people use to combine deterministic information in their mind; sometimes referred to as combination rules.

See

- Choice Theory
- Decision Analysis
- Decision Making and Decision Analysis

Choice Theory

Leonard Adelman
George Mason University, Fairfax, VA, USA

Introduction

There is no one descriptive theory of human choice. Instead, there are different theoretically and empirically-based approaches for describing choice behavior. This article briefly overviews five approaches: bounded rationality, prospect theory, choice strategies, recognition-primed decision making, and image theory. These approaches are descriptive in the sense that they describe certain aspects of how people actually make choices. They contrast with prescriptive approaches, such as decision analysis or other economic-based theories (or models) of choice behavior, which prescribe how one should make decisions, but do not necessarily describe choice behavior.

Bounded Rationality

The concept of bounded rationality is attributed to Nobel laureate Herbert Simon (Simon 1955, 1979; Hogarth 1987), who argued that humans lack both the knowledge and computational skill required to make choices in a manner compatible with economic notions of rational behavior. The rational model's requirements are illustrated by the concept of a payoff matrix, an example of which is presented in Table 1.

The rows of the matrix represent all the different alternatives available to the decision maker for solving a choice problem. The columns represent all of the different states of the world, as defined by future events, that could affect the attractiveness of the alternatives. The $p_1,\ldots, p_k$ values represent the probabilities for each state of the world. The cell entries in the matrix indicate the value or utility of the outcome or payoff for each combination of alternatives and states of the world. Each outcome represents a cumulative payoff comprised of perceived advantages and disadvantages on multiple criteria of varying importance to the decision maker. Finally, the rational decision maker is required to select the alternative that maximizes expected utility, which is calculated for each alternative by multiplying the values for the outcomes by the probabilities for the future states, and then summing the products.

Numerous studies have shown that, unaided, people do not employ the above decision matrix due to the complex, dynamic nature of the environment and to basic human information acquisition and processing limitations. Therefore, how does unaided human choice remain purposeful and reasonable? Simon

Choice Theory, Table 1 The rational economic model's decision making requirements as represented in a payoff matrix

	States of the world			
Alternatives	S_1 (p_1)	S_2 (p_2)	. . .	S_k (p_k)
A	a_1	a_2	. . .	a_k
B	b_1	b_2	. . .	b_k
.	.	.	. . .	.
N	n_1	n_2	. . .	n_k

suggested that people employ three simplification strategies, which result in a bounded rationality. First, people simplify the problem by only considering a small number of alternatives and states of the world at a time. Second, people simplify the problem by setting aspiration (or acceptability) levels on the outcomes. And, third, people choose the first alternative that satisfies the aspiration levels. In other words, people do not optimize (i.e., choose the best of all possible alternatives), but satisfice (i.e., choose the first satisfactory alternative). In this way, people can reduce information acquisition and processing demands and act in a purposeful, reasonable manner.

Prospect Theory

Like Simon's bounded rationality, prospect theory is juxtaposed against expected utility theory. For example, this prospect (or choice) is taken from Kahneman and Tversky (1979):

$$\text{Choice } A: (\$4000 \text{ with } p = .8;\ \$0 \text{ with } p = .2), \text{ or}$$
$$\text{Choice } B: (\$3000 \text{ for sure; that is, } p = 1.0)$$

The majority of participants will select Choice B. Yet, Choice A has the greater expected value; that is, $\$4000 \times .8 = 3200$. Now, consider the following prospect:

$$\text{Choice } C: (-\$4000 \text{ with } p = .8;\ \$0 \text{ with } p = .2), \text{ or}$$
$$\text{Choice } D: (-\$3000 \text{ for sure; that is, } p = 1.0).$$

The only change in the second prospect is that the sign has been reversed so that one is now considering losses, not gains. In this case, however, the majority of the subjects picked Choice C. That is, they would now be willing to take a gamble of losing $4000 with a probability of.8, which has an expected value of losing $3200, instead of taking a sure loss of $3000. Again, they selected the choice with the lower expected value. In addition, they switched from the sure thing to preferring the gamble.

What Kahneman and Tversky (Tversky and Kahneman 1981) have shown is that the way the choice problem is presented (or framed) significantly affects how people evaluate it, such that information that should result in the same choice from the perspective of expected utility theory actually results in different choices. In particular, people perceive outcomes as gains or losses from a reference point rather than from final states (e.g., of wealth), as assumed by economic-based models of choice. The current position is usually considered as the reference point. However, the location of the reference point and, in turn, the coding of outcomes as either gains or losses, can be affected by how the choices are framed.

This framing is particularly important for choice because, as the example presented above indicates, people tend to be risk adverse when considering gains and risk seeking when considering losses, particularly if one of the prospects is certain. Moreover, the value function is steeper for losses than for gains, consistent with the observation that losses loom much larger than gains. For these reasons, many people are willing to gamble to avoid a sure loss, but unwilling to gamble when they have a sure gain, even when both choices have a lower expected value than another choice.

Choice Strategies

Substantial research has focused on describing the different strategies people use to combine information when facing a choice. In contrast to bounded rationality and prospect theory, these strategies are used when people (a) have information on a number of different dimensions (or attributes) describing the alternatives, and (b) do not consider probabilities, either in terms of different states of nature or the reliability (or accuracy) of the information. A representative type of problem is making a purchase decision, such as choosing a car.

The literature (Beach 1990; Hogarth 1987) makes a distinction between two classes of choice strategies: compensatory and noncompensatory. Compensatory

strategies are used when one trades-off (e.g., via relative importance weights) a low value on one attribute for a high value on another. For example, when choosing among cars, one may trade-off gas mileage for comfort. Non-compensatory strategies do not employ trade-offs but, rather, employ thresholds (or cut-offs) that need to be achieved for choice of an alternative. For example, one eliminates all cars that do not get at least 25 miles per gallon, regardless of comfort. Some of the strategies identified in the literature are defined below.

The literature cites three different types of compensatory models:

1. *Linear, additive strategy* — the value of an alternative is equal to the sum of the products, over all the dimensions, of the relative weight times the scale value for the dimension.
2. *Additive difference strategy* — the decision maker evaluates the differences between the alternatives on a dimension by dimension basis, and then sums the weighted differences in order to identify the alternative with the highest value overall.
3. *Ideal point strategy* — is similar to the additive difference model, except the decision maker compares the alternatives against an ideal alternative instead of each other.

The literature cites four different types of noncompensatory strategies:

4. *Dominance strategy* — select the alternative that is at least as attractive as the other alternatives on all the dimensions, but is better than them on at least one dimension. Although the dominance strategy is easier for an unaided decision maker to use, all three compensatory strategies will also identify the dominant alternative. Moreover, the compensatory strategies can be used if there is no dominant alternative; the dominance strategy cannot.
5. *Conjunctive strategy* — select the alternative that best passes some critical threshold on all dimensions. This is the satisficing strategy when one selects the first option that passes a threshold on all dimensions. The conjunctive strategy is often used to reduce the set of alternatives by eliminating all alternatives that fail to pass a threshold on all dimensions.
6. *Lexicographic strategy* — select the alternative that is best on the most important dimension. If two or more alternatives are tied, select among them by choosing the alternative that is best on the second most important dimension, and so on.
7. *Elimination by aspects* —sequentially identify different dimensions, either according to their importance or some more probabilistic scheme. Eliminate all alternatives that fail to pass the threshold or aspect for each dimension until only one alternative is left.

Research (Payne et al. 1993) has shown that people often use multiple strategies when considering choice alternatives. Typically, they use noncompensatory strategies to reduce the number of alternatives and dimensions under consideration. To use a job selection example, a person might first eliminate all alternatives that fail to pass a specific threshold on security, which may no longer be as important when considering the reduced set of alternatives. Then, after the set of alternatives and dimensions have been reduced to a smaller, more manageable set, people often employ a compensatory strategy where they weigh the strengths and weaknesses of the remaining alternatives in order to select the one which best satisfies their values.

Recognition-Primed Decision Making (RPD) and Image Theory

Some descriptive theories have been developed to explain the choice behavior of experts working in naturalistic settings (Zsambok and Klein 1997). These descriptive theories of choice behavior are farther removed from the basic rational economic man model than the three presented thus far. Two are presented here, RPD and image theory, for illustrative purposes.

The RPD model (Klein 1993) emphasizes four critical cognitive processes:

- *Situation recognition* — experienced decision makers know what cues (or indicators) to focus on and, often, simply recognize (or perceive) the situation they are facing through an automatic, feature (or pattern)-matching process, much like perceptual objects in our environment are recognized. People also are quite capable of using explanation-based reasoning to understand a situation when there uncertainties and anomalies in it. In fact, these stories are often, but not always, the causal explanations for the feature-matching process that appears to operate so automatically.

- *Decision option generation* — Once a situation is recognized, decision makers typically generate only one option for consideration, not multiple options.
- *Evaluation through mental simulation* — The initial option tends to be quite good for dealing with the (recognized) situation. Decision makers, however, may evaluate it by mentally simulating the consequences of implementing the option. Although the mental simulation will use intuitive and analytical thought processes, depending on the consequences being evaluated during the simulation, the option will seldom, if ever, be evaluated by a formal analysis on a set of attributes (e.g., by a decision matrix).
- *Use of a decision rule* (emphasizing acceptability, not optimality) — The mental simulation may result in modifications to the proposed option to address problems uncovered during the mental simulation, or even a new option, but the option will be accepted once it is deemed satisfactory; it does not have to be optimal. Thus, RPD explicitly incorporates Simon's (1955) satisficing concept.

Beach (1990, 1993) developed the concept of images to convey the notion that decision makers bring certain knowledge structures to bear on a problem that constrain (or frame) how they evaluate it. In particular, Beach discussed three images: value, trajectory, and strategic, as follows:

- *Value Image* — this is composed of the overriding principles that guide one's behavior or that of one's organization. They "serve as rigid criteria for the rightness or wrongness of any particular decision about a goal or plan" (Beach 1993, p. 151).
- *Trajectory Image* — this consists of previously adopted goals, the timetable for achieving them, and the ideal future once they are achieved.
- *Strategic Image* — this is composed of the plans for achieving the goals in the trajectory image. The plans consist of specific tactics (or actions) for implementing the more abstract plan, and forecasts of what will happen if specific tactics are implemented. These forecasts change in light of new information. "By monitoring these forecasts (or expectations) in relation to the goals on the trajectory image, the decision maker can evaluate his or her progress toward realization of the ideal agendum on the trajectory image" (Beach 1993, p. 152).

In many ways, Beach's image theory is another way for describing the cognitive processes emphasized in Klein's RPD model. Image theory also emphasizes: (a) monitoring behavior; (b) expectations and goals; (c) situation recognition through feature matching and explanation-based reasoning; (d) automatic generation of a decision option to deal with the recognized situation; (e) mental simulation to evaluate it; (f) satisficing; and (g) processes for monitoring and managing the decision process. It is different, however, in its emphasis of three things.

The first difference in emphasis is that the images strongly frame the interpretation of how well the situation is going or even what the problem is, as emphasized in Prospect Theory. (Keeney (1992) also emphasized framing in his approach, called value-focused thinking, but from more of a prescriptive than descriptive perspective.) Second, routine progress decisions are made to compare the current situation and future forecasts with the ideal future. And, third, adoption decisions are routinely made to modify plans, tactics, and expectancies (i.e., the elements of Strategic Image) — and, less frequently, goals, timetables, and ideal future (i.e., the elements of Trajectory Image) — in response to progress decisions. Depending on the situation and person, these adoption decisions are made using one or more of the choice strategies described above. Although frames, progress decisions, and adoption decisions may be concepts that are inherent in Klein's RPD model, they are strongly and explicitly emphasized in Beach's image theory. In addition, Beach's image theory explicitly incorporates the many descriptive choice strategies found in the literature. Thus, image theory integrates many of the concepts in the choice theory literature to describe how people choose to react to changing situations.

Concluding Remarks

In closing, there is a need to emphasize again that there is no one descriptive theory of human choice. Instead, there are different theoretically and empirically-based approaches for describing choice behavior. This article provided brief overviews of five of them: bounded rationality, prospect theory, choice strategies, recognition-primed decision making, and image

theory. These approaches were contrasted with prescriptive approaches, such as decision analysis or other economic-based theories of choice, which prescribe how one should make decisions, but do not necessarily describe choice behavior.

See

- ▶ Decision Analysis
- ▶ Decision Making and Decision Analysis
- ▶ Preference Theory
- ▶ Utility Theory

References

Beach, L. R. (1990). *Image theory: Decision making in personal and organizational contexts*. New York: Wiley.
Beach, L. R. (1993). Image theory: Personal and organizational decisions. In G. Klein, J. Arisen, R. Calderwood, & C. E. Zsambok (Eds.), *Decision making in action: Models and methods*. Norwood, NJ: Ablex.
Hogarth, R. M. (1987). *Judgment and choice*. New York: Wiley.
Kahneman, D., & Tversky, A. (1979). Prospect theory: An analysis of decision making under risk. *Econometrica, 47*, 263–289.
Keeney, R. L. (1992). *Value-focused thinking*. Cambridge, MA: Harvard University Press.
Klein, G. (1993). A recognition-primed decision (RPD) model of rapid decision making. In G. Klein, J. Arisen, R. Calderwood, & C. E. Zsambok (Eds.), *Decision making in action: Models and methods*. Norwood, NJ: Ablex.
Payne, J. W., Bettman, J. R., & Johnson, E. J. (1993). *The adaptive decision maker*. New York: Cambridge University Press.
Rubinstein, A. (1998). *Modeling bounded rationality*. Cambridge, MA: MIT Press.
Simon, H. A. (1955). A behavioral model of rational choice. *Quarterly Journal of Economics, 69*, 99–118.
Simon, H. A. (1979). Rational decision making in business organizations. *American Economic Review, 69*, 493–513.
Tversky, A., & Kahneman, D. (1981). The framing of decisions and the psychology of choice. *Science, 211*, 453–458.
Wakker, P. P. (2010). *Prospect theory: For risk and ambiguity*. Cambridge, UK: Cambridge University Press.
Zsambok, C. E., & Klein, G. (1997). *Naturalistic decision making*. Hillsdale, NJ: Erlbaum.

Chromatic Number

In a graph, the minimum of colors needed to ensure that adjacent nodes receive different colors.

See

- ▶ Graph Theory

Chromosome

In genetic algorithms, a chromosome represents a potential solution to the problem at hand.

See

- ▶ Evolutionary Algorithms

CIM

Computer integrated manufacturing.

See

- ▶ Automation in Manufacturing and Services

Circling

- ▶ Cycling

Classical Optimization

- ▶ Unconstrained Optimization

Closed Network

A queueing network in which there is neither entrance nor exit but only a fixed number of customers endlessly circulating.

See

- ▶ Networks of Queues

Closed-Loop Supply Chains

Gilvan C. Souza
Indiana University Bloomington, Bloomington, IN, USA

Introduction

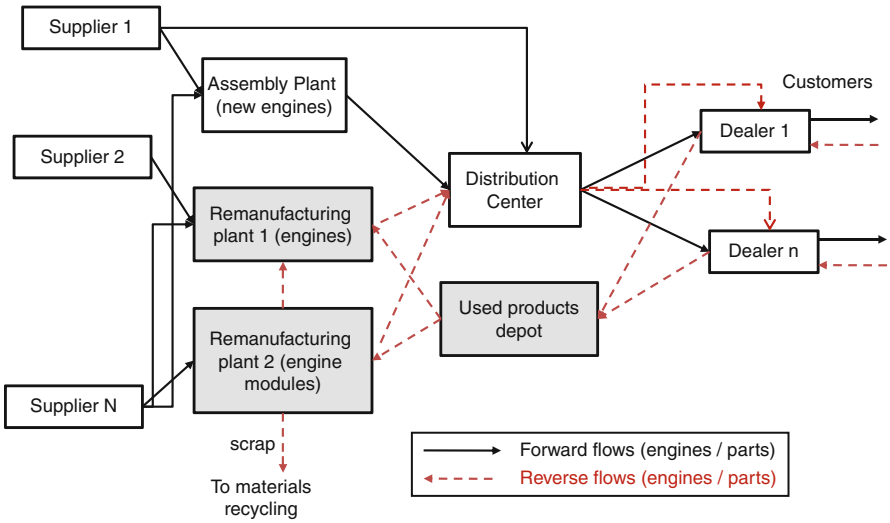
In a (regular) supply chain, there are physical flows of products, components or subassemblies from suppliers to subassembly manufacturers, from subassembly manufacturers to Original Equipment Manufacturers (OEMs), and from OEMs to customers through a distribution system. The distribution system could be comprised of a combination of distribution centers, central, and regional warehouses, and resellers. In the traditional supply chain literature, these physical flows are assumed to be unidirectional, from suppliers to customers. There are, however, bi-directional financial and information flows, for example, orders and payments placed from one tier in the supply chain (e.g., distributors) to its immediate upper tier (OEMs).

Closed-Loop Supply Chains (CLSC)

In a closed-loop supply chain (CLSC), there are, in addition to the forward physical flows described above, reverse physical flows of (used) products or components from customers to manufacturers (and possibly suppliers). As an example of closed-loop supply chain, consider the supply chain for diesel engines and parts for a major North American OEM (Fig. 1). Figure 1 depicts the main physical flows in this supply chain in a simplified manner; the flows are differentiated between forward and reverse flows. Forward flows consist of new parts and/or engines, and reverse flows consist of used parts and/or engines, and remanufactured parts or engines. Remanufacturing is the process of restoring a used product (post consumer use) to a common operating and aesthetic standard, sometimes with upgrades to the original product's functionality. For a diesel engine or module, remanufacturing consists of six different steps: (i) full disassembly to the part level, (ii) thorough cleaning of each part (often through multiple sequential techniques), (iii) making a disposition decision for each part (keep for remanufacturing or dispose the part for materials recycling), (iv) salvaging if necessary (value added work that restores functionality to that of a new part), (v) re-assembly, and (vi) testing. Other terms commonly used for remanufacturing include refurbishing, rebuilding, and overhauling, depending on the industry (no such distinctions are made in this article).

New engines are produced and assembled from new parts, some of which are manufactured and shipped by the OEM's many suppliers. Those suppliers also supply the firm's distribution center with spare parts. New engines are shipped to a (central) distribution center; they are then shipped from this (central) distribution center to several regional distribution centers (not depicted in Fig. 1), and from there to over 3,000 dealers in North America. Customers, say a trucking company, buy new (or remanufactured) diesel engines or engine modules, say a water pump or a turbocharger, from dealers due to replacement needs. They receive a dollar credit from returning the old engine or module upon purchasing a new (or remanufactured) engine or module; the dollar credit can be as high as 30% off the purchase price. Remanufactured engines or modules sell at a 35% discount relative to the corresponding new engine or module. Used modules or engines are shipped from dealers to one of 30 or so different consolidation points in North America (not depicted in Fig. 1), and from there to the OEM's main used products depot. At the depot, shipments are unpacked, customers are given the proper credit for returning the used module or engine; engines and modules are then shipped to one of two plants (or put into inventory for later shipment when needed): engine remanufacturing (plant 1), or part (or module) remanufacturing (plant 2). Remanufactured engines are shipped from plant 1 to the main distribution center, joining new engines or parts for distribution to the dealers. Remanufactured parts or modules are shipped from plant 2 to either the distribution center, or to the engine remanufacturing plant 1, depending on forecasts and current needs. Used parts not suited for remanufacturing are sold to recyclers. The flows depicted in Fig. 1 are simplified, but they convey the major flows in this CLSC. Used products are typically referred to as returns or cores.

The supply chain in Fig. 1 illustrates two major disposition decisions for cores: remanufacturing and recycling. Recycling means materials recovery, i.e., the



Closed-Loop Supply Chains, Fig. 1 CLSC for diesel engines and parts (simplified)

geometry of the used part or product is not preserved. Recycling occurs when remanufacturing is not possible or not economical, for example, the core is highly damaged (e.g., an engine block with a hole on it), there is significant wear and tear of the part (piston rings), or the part is technologically obsolete (this is common in consumer electronics and computers). In addition to these two disposition options, the firm can also disassemble a core, and use some of the resulting parts as spare parts for, say, fulfilling warranty claims or servicing products under service contracts; in these cases there may be no cleaning or salvage needed. Dismantling for spare parts is common in electronic goods industries, such as computers, and IT networking equipment (servers, routers, switches). Dismantling for spare parts can be an attractive alternative when it produces significant savings compared to procuring a new part from a supplier, when demand for remanufactured products is weak, or when the part supplier is no longer active. Other disposition decisions include incineration (which can recover energy, but there can be pollution concerns), and dumping in landfills (which is illegal for some materials known to contaminate water and soil). This article focuses on remanufacturing, as it presents significant operational challenges due to the natural mismatch between supply of cores, which for most firms is not certain, and demand for remanufactured products, which is also uncertain; as a result remanufacturing is a natural candidate for application of OR models.

In addition, Fig. 1 illustrates a CLSC where the main source of cores are end-of-use returns, where the product has undergone a full cycle of use with a customer, but the product still has significant value left for recovery. In addition to end-of-use returns, there are end-of-life returns, which are products that have reached the end of their useful life, mostly due to obsolescence, and whose main disposition decision is recycling; examples include very old computers, monitors, VCRs, and very old cars. Finally, there are consumer returns, which are products that have undergone little or no use by consumers—they are returned by consumers to retailers as a result of liberal returns policies by powerful retailers primarily in North America; most consumer returns are not defective. For example, about 80% of deskjet printers returned to retailers by consumers in the U.S. are not defective; reasons for return include remorse, and lack of product fit with consumer needs (Ferguson et al. 2006).

For the design and operation of a CLSC, the decision making is classified into three buckets, as is the case for regular supply chains: strategic, tactical, and operational. Examples of these three types of decisions are shown in Table 1. As Table 1 suggests, there are many different decisions in CLSC management that are amenable to the use of OR tools and techniques, including mathematical programming, Markov decision processes, and simulation.

Next, a brief description is given on how OR techniques are applied to two decisions: the decision to remanufacture or not by an OEM, and network design. A more complete review of the basic models and extensions for the other decisions is given in Souza (2008) and Ferguson and Souza (2010).

Closed-Loop Supply Chains, Table 1 Strategic, Tactical and Operational Decisions in CLSCs

Decision Type	Examples
Strategic	• **Remanufacturing or not:** Should an OEM remanufacture? • **Network design:** What is the location of remanufacturing plants, recycling plants, collection points, and consolidation points? Should used products be collected through retailers, or directly from consumers? Should forward and reverse flows be combined, or should the forward and reverse supply chain be separate? • **Leasing:** Should the firm lease or sell to customers? • **Strategic alliances:** Should the firm enter into partnerships with third-parties for remanufacturing or collection of its products? • **Design for recovery:** How should a firm design a product if there is remanufacturing at the end of use?
Tactical	• **Product acquisition:** How many used cores should the firm acquire, when, in which quality, and at what price? • **Remanufacturing planning and disposition:** Given a supply of cores, demand forecasts, relevant costs and revenues, what should a firm do with a core (remanufacture, recycle, dismantle for parts), and when?
Operational	• **Disassembly planning:** What is the sequence and depth of disassembly for a core? • **Shop floor scheduling and control**: What is the routing and scheduling priority for remanufacturing orders in the job shop?

Should an OEM Remanufacture?

On the surface, the decision to remanufacture or not by an OEM appears to be simple: if the price the firm can sell a remanufactured product far exceeds the variable cost to remanufacture a core, which includes collection and transportation of the core, disassembly, cleaning, salvaging, re-assembly, testing, and remarketing, then the firm should remanufacture, after properly accounting for any upfront fixed remanufacturing costs (e.g., building a facility and acquiring equipment). This simple revenue vs. cost accounting may not capture all of the facets of the problem, however. There are other factors that favor remanufacturing, such as: (i) extending the OEM's product line and offering a product, priced most likely lower than the corresponding new product, and therefore reaching a customer segment that would not otherwise be reached; (ii) allowing brand protection, given that many third party firms offer remanufactured products—if the OEM offers a certified remanufactured product, then it communicates to consumers that only its version of remanufactured product has the appropriate quality level; (iii) using as a deterrent to market entry of third-party remanufacturers; and (iv) value recovery for used products returned after leases, trade-in programs, or consumer returns. On the other hand, there are factors that do not favor remanufacturing; chiefly among them are: (a) the fear of cannibalization of sales of a (typically more expensive and more profitable) new product by a (typically less expensive and less profitable) remanufactured product; and (b) the ability to reliably collect an appropriate pipeline of cores to sustain a remanufacturing operation. Discussions with manufacturing managers indicate that factor (a) is of significant concern to firms in the IT equipment industry, while factor (b) is of significant concern to firms in automotive parts remanufacturing.

The following simple analytic model provides some insights into the answer to the critical strategic question, "Should an OEM remanufacture?" Consider an OEM selling a new product (say, a diesel engine model X), and considering the decision to offer its remanufactured counterpart (a remanufactured diesel engine model X). The firm has to decide whether to offer the remanufactured product, and if so, how to set the prices of remanufactured and new products, denoted by p_r and p_n respectively. First, consider a monopolist under a single period model (extensions are discussed below.) The model is based on some assumptions about consumer behavior. Specifically, assume that the consumer base is heterogeneous, so that consumers differ in their intrinsic valuation for the new product. A consumer such as a third-party logistics company, for example, with an extensive fleet of large trucks, has a high valuation for the new diesel engine; whereas an operator of small gasoline-powered delivery trucks has a low intrinsic valuation for a new diesel engine. This intrinsic valuation for the new product, which differs across the heterogeneous consumer base, is referred to as willingness-to-pay (the maximum amount a consumer is willing to pay), or w.t.p., for a new

product. Each consumer has an intrinsic w.t.p., a random variable that is denoted by θ. Thus, a consumer has a unique association with its w.t.p. θ and can be referred to simply as consumer θ. Further, it is assumed θ is uniformly distributed between a lower and an upper bound, where the bounds are normalized to be zero dollars and one dollar, respectively. Mathematically, $\theta \sim U[0,1]$. Thus, all consumers (potential customers) are distributed uniformly in the real line between \$0 and \$1. This assumption is common in the marketing and operations literature, because it results in linear demand curves, as shown below (it also allows analytical tractability). A consumer θ's w.t.p. for a remanufactured product is, however, $\delta\theta$, where $0 \leq \delta \leq 1$. If $\delta = 0$, then consumers do not consider the remanufactured product as a potential substitute for the new; this is a limiting case. An example that approaches this limit is retreaded passenger car tires in the U.S., where many consumers perceive them as unsafe, and there are many cheap imports that are priced quite low but are new. If $\delta = 1$, then consumers perceive remanufactured and new products as perfect substitutes. One example of this is retreaded truck tires used in commercial fleets in the U.S., where fleet owners have service contracts with certain dealers and pay them by each mile of service a tire provides to the fleet owner. These firms are insensitive as to whether the dealers use retreaded or new tires to keep the truck running. Most products fall in between, i.e., $0 < \delta < 1$; [see Hauser and Lund (2003), and Souza (2008) for a complete discussion]. For diesel engines, for example, $\delta \cong 0.65$; for power tools $\delta \cong 0.85$.

Suppose the firm only offers the new product at a price $p_n \leq 1$ (the firm would never offer a new product priced higher than \$1 because the maximum w.t.p. in the consumer base is \$1). Then, only those consumers with w.t.p θ higher than p_n buy the product, because they are the only ones with a non-negative net utility $(\theta - p_n)$ for the product. Because consumers' w.t.p. are distributed uniformly between 0 and 1, then the number of consumers that buy the new product (q_n) is $q_n = M \cdot \Pr\{\theta - p_n \geq 0\} = M \cdot \Pr\{\theta \geq p_n\}$, where M is the overall size of the consumer base (number of potential customers). Normalizing $M = 1$, and because $\theta \sim U[0,1]$, then $\Pr\{\theta \geq p_n\} = (1 - p_n)/1 = 1 - p_n$; as a result the firm sells $q_n = 1 - p_n$ new products. Now, suppose the firm offers both new and remanufactured products at prices p_n and p_r, respectively. A consumer θ's net utility for a new product is $\theta - p_n$, and for a remanufactured product is $\delta\theta - p_r$. Consumers whose net utilities are higher for a new than for a remanufactured product, i.e., $\theta - p_n > \delta\theta - p_r$, buy a new product; solving for θ yields $\theta > (p_n - p_r)/(1 - \delta)$. If $\theta < (p_n - p_r)/(1 - \delta)$, then consumers have a higher net utility for a remanufactured than a new product; they will buy remanufactured if their net utility is positive, that is, $\delta\theta - p_r > 0$, or $\theta > p_r/\delta$. Consumers with w.t.p. θ lower than p_r/δ will not buy anything. Given the uniform distribution for θ, the quantities of new and remanufactured products sold, given their prices, are $q_n = 1 - (p_n - p_r)/(1 - \delta)$, and $q_r = (p_n - p_r)/(1 - \delta) - p_r/\delta$, respectively. These two expressions constitute the demand curves for new and remanufactured products given respective prices; the demand curves are linear, assuming a uniform w.t.p. distribution (a different distribution results in a different demand curve shape). For a period with R cores available for remanufacturing, denote the remanufacturing yield—the percentage of cores that are found fit for remanufacturing—by μ. Further, assume that the remanufacturing cost per unit is constant at c_r, and the manufacturing cost per unit (new) is c_n. Then, the OEM's decision problem can be formulated as:

$$\max_{p_n, p_r} \Pi = q_n(p_n - c_n) + q_r(p_r - c_r), \tag{1}$$

$$\text{s.t. } q_r \leq \mu R, \tag{2}$$

$$q_n, q_r \geq 0, \tag{3}$$

Equation (1) is the OEM's per period profit; equation (2) is a constraint that limits the availability of cores for remanufacturing, and equation (3) is a logical constraint. Note that the decision variables are the prices p_n and p_r, thus, one needs to substitute the corresponding expressions for the quantities $q_n = 1 - (p_n - p_r)/(1 - \delta)$, and $q_r = (p_n - p_r)/(1 - \delta) - p_r/\delta$ in (1–3). This is a non-linear optimization problem, which can be solved analytically. The solution comprises several regions, depending on which constraints are binding or not. It can be shown that if $c_r < c\delta$ (and $R > 0$), then

the firm remanufactures, i.e., $q_r > 0$. Thus, the decision to remanufacture in this simple model is dependent upon the unit remanufacturing cost relative to new, the consumer's perception of remanufactured products relative to new (δ), and the availability of cores.

The model described above is very stylized, and does not include the following factors that are (typically) present in real life:

1. Competition: This transforms the decision problem into a game. There is a second decision maker with an objective function similar to the second term in (1). The demand functions now become significantly more complicated. See Atasu et al. (2008) for a way to incorporate competition.
2. Non-linear recovery costs: the model above assumes a constant marginal remanufacturing cost c_r. In practice, there is a cost of collection that is convex increasing in the quantity of cores collected, because it is increasingly more difficult to improve collection rates. Remanufacturing cost per se (i.e., disassembly, cleaning, salvage, testing) may also be convex increasing in the quantity remanufactured, because as remanufacturing quantity increases, the firm needs to dig deeper into the pile of cores, and remanufacture cores in worse quality condition, which demands more labor and materials. The combination of convex collection and remanufacturing costs implies that c_r in (1) is substituted with αq_r^2, where α is a positive constant. See Ferguson and Toktay (2006) for an analysis of this problem.
3. Availability of cores is dependent on sales in previous periods: simply put, the number of cores available for recovery are a function of sales in previous periods. To capture this dimension, one needs a multi-period model, so that all decision variables are defined for each period t: $p_{n,t}$, $p_{r,t}$, $q_{n,t}$, and $q_{r,t}$. If a product can only be remanufactured once (for example, if the product becomes technologically obsolete after the third generation is introduced), and L is the lag, in periods, between the sale of a new product, and its collection as a core post-consumer use, then (2) should be rewritten as $q_{r,t} \leq \mu q_{n,t-L}$, for each $t > L$. (See Ferrer and Swaminathan (2006), and Debo et al. (2005) for examples of models incorporating this dynamic aspect).

CLSC Network Design

As Table 1 indicates, designing a CLSC network requires deciding upon the locations of manufacturing and remanufacturing plants, warehouses (or distribution centers), points of sale, and consolidation centers for shipping cores from points of sale to remanufacturing plants, among other facilities. To help design such a network, a mixed-integer linear program (MILP) is described next based on the modeling framework by Fleischmann et al. (2001). For a review of CLSC network design, see Ammons et al. (2001) and Pochampally et al. (2008).

Assume an OEM that manufactures and remanufactures products, similar to the diesel engine CLSC shown in Fig. 1. The supply chain comprises four levels: (i) manufacturing and remanufacturing plants (a facility can do one or both), (ii) warehouses for distribution of manufactured and remanufactured products, (iii) consolidation centers for consolidating shipments of used products originating from resellers for shipment to plants, and (iv) resellers, who are independent entities that sell manufactured and/or remanufactured products to customers, in addition to collecting used products from customers.

Indexes

i Potential plant locations, $i \in I, I_0 = I \cup \{0\}$, where $i = 0$ is the disposal option.
j Potential warehouse locations, $j \in J$
k Fixed reseller locations, $k \in K$
l Potential consolidation center locations, $l \in L$

Variables

X_{ijk}^f Fraction of reseller k's demand served from plant i through warehouse j
X_{kli}^r Fraction of reseller k's returns returned to plant i through consolidation center l
U_k Unsatisfied fraction of reseller k's demand
W_k Uncollected fraction of reseller k's returns
Y_i^p Indicator variable for opening plant i ($= 1$ if plant is open; 0 otherwise); Y_j^w and Y_l^r are similarly defined

Costs

c_{ijk}^f Unit cost (transportation, production, handling) of serving k from i via j

c^r_{kli} Unit cost of returns (transportation, handling) from k to i via l

c^r_{kl0} Unit disposal cost (including collection, transportation, handling) for k via l

c^u_k Unit penalty cost for not serving reseller k's demand

c^w_k Unit penalty cost for not collecting reseller k's returns

f^p_i Fixed cost for opening plant i (f^w_j and f^r_l similarly defined)

Parameters

D_k Demand for reseller k

R_k Returns from reseller k

γ Minimal disposal fraction

Note that in this formulation, the continuous decision variables (e.g., X^f_{ijk} and X^r_{kli}) are defined in terms of *fractions* of demand and returns at each reseller, and as a result they are all bounded below by zero and above by one. An alternative formulation would have the continuous decision variables defined simply as quantities shipped. The firm's network design problem can be formulated mathematically as a MILP as follows:

$$TC = \min \sum_{i\in I} f^p_i Y^p_i + \sum_{j\in J} f^w_j Y^w_j + \sum_{l\in L} f^r_l Y^r_l + \sum_{i\in I}\sum_{j\in J}\sum_{k\in K} c^f_{ijk} D_k X^f_{ijk} + \sum_{k\in K}\sum_{l\in L}\sum_{i\in I_0} c^r_{kli} R_k X^r_{kli} + \sum_{k\in K}\left(c^u_k D_k U_k + c^w_k R_k W_k\right) \tag{4}$$

$$\text{s.t.} \sum_{i\in I}\sum_{j\in J} X^f_{ijk} + U_k = 1, \forall k \tag{5}$$

$$\sum_{l\in L}\sum_{i\in I_0} X^r_{kli} + W_k = 1, \forall k \tag{6}$$

$$\sum_{k\in K}\sum_{l\in L} R_k X^r_{kli} \le \sum_{j\in J}\sum_{k\in K} D_k X^f_{ijk}, \forall i \tag{7}$$

$$\gamma \sum_{i\in I_0} X^r_{kli} \le X^r_{kl0}, \forall k, \forall l \tag{8}$$

$$\sum_{j\in J} X^f_{ijk} \le Y^p_i, \forall i, \forall k \tag{9}$$

$$\sum_{i\in I} X^f_{ijk} \le Y^w_j, \forall j, \forall k \tag{10}$$

$$\sum_{i\in I_0} X^r_{kli} \le Y^r_l, \forall k, \forall l \tag{11}$$

$$Y^p_i, Y^w_j, Y^r_l \in \{0,1\}\ \forall i, \forall j, \forall l \tag{12}$$

$$0 \le X^f_{ijk}, X^r_{kli}, U_k, W_k \le 1, \forall i, \forall j, \forall k. \tag{13}$$

The objective function (4) minimizes total cost, comprised of fixed costs of opening and operating the facilities, and variable distribution costs. Constraints (5–6) represent basic flow constraints, which indicate that, for each reseller k, shipments plus unsatisfied demand are equal to total demand; similarly for reseller k's returns. Constraint (7) represents flow balancing at each plant, where the difference between incoming returns and outgoing shipments represent manufacturing of new products. Constraint (8) indicates that the number of disposed products should be larger than a given fraction of all returned products; in this model disposal is meant to represent material recycling or dismantling for spare parts. Thus, constraint (8) indicates the extent remanufacturing should take place (for example if $\gamma = 1$, then there is no remanufacturing, and all cores are recycled, or dismantled for spare parts). Constraints (9–11) are logical constraints—there is no shipment to/from a facility if that facility is not open. Constraint (12) states the binary decision variables for the problem, and constraint (13) represents the non-negativity constraints, and the fact that the continuous variables in this problem are defined as fractions of total demand or total returns at each reseller.

As described in Fleischmann et al. (2001), this formulation is very general and can accommodate many different scenarios. For example, if the firm has two separate networks for forward and reverse flows, then it can set R_k and D_k equal to zero, respectively. The values of c^f_{ijk} relative to c^r_{kli} and c^u_k can model different production scenarios for each plant, such as whether a plant only produces new products or it only produces remanufactured products, or both. In this model, demand can be met through remanufactured or new products. If there are separate demand streams for these products, then one can add another index, say t, to the decision variables and parameters of the problem to indicate the product type. For example, D_{kt} would be demand at customer k for product type t, where $t \in$ {remanufactured, new}. Finally, this

problem has been aggregated in that there is only one "aggregate" product being sold at the resellers. In the case of the CLSC for diesel engines, modules and parts remanufacturing are treated differently than engine remanufacturing; to accommodate this scenario one can again simply add another index to indicate product type (say, entire engines, or modules).

The discussion above was centered on remanufacturing as the key recovery activity taking place in the network. The same formulation can be used to design a network for recycling; an example is paper recycling, studied by Bloemhof-Ruwaard et al. (1996).

CLSC OR Applications

Closed-loop supply chain management provides numerous opportunities for application of OR methodology. The two applications discussed above are at the strategic level: an OEM's decision to engage in remanufacturing, and the design of a network of remanufacturing and manufacturing plants, distribution centers, and consolidation centers. The first model analyzes an OEM's decision to remanufacture—it is based on a model of consumer behavior that, in essence, implies linear demand curves for remanufactured and new products, and where they are partial substitutes for each other, so that each customer values a remanufactured product less than a corresponding new product. Based on these demand curves, and relevant costs, the firm chooses prices for remanufactured and new products that maximize its profit by solving a non-linear program. The second model is more of a decision support-type model—given relevant fixed costs of opening new facilities, relevant distribution costs, and demand and return points, the firm designs its CLSC network to minimize its fulfillment costs.

One significant area of application of OR models not covered in this article concerns the match between supply of cores and demand for remanufactured products and parts. The problem of product acquisition—acquiring the right amount of cores at the right price at the right quality at the right time, and its corresponding disposition decision—deciding what to do with a core: disassemble, remanufacture, recycle, or put it in inventory for future use, given relevant demand forecasts, and underlying costs is of significant importance to firms, at the tactical level. See Souza (2008) and Ferguson and Souza (2010) for more information on these models.

For the range of CLSC applications, especially from a policy maker's perspective, the design of environmental legislation is of significant concern. Specifically, the decision maker (say, the government) is interested in designing environmental legislation that sets appropriate collection and recycling levels to maximize society's welfare. This is comprised of total profits across all manufacturers impacted by legislation, consumer surplus, and environmental benefits of the legislation (e.g., lower pollution levels); notice that environmental benefits must be measured in dollar terms. Again, OR models can be used to help design such legislation (Ferguson and Souza 2010).

See

- ▶ Industrial Applications
- ▶ Integer and Combinatorial Optimization
- ▶ Supply Chain Management

References

Ammons, J. C., Realff, M. J., & Newton, D. J. (2001). Decision models for reverse production system design. In C. N. Madu (Ed.), *Handbook of environmentally conscious manufacturing*. Boston: Kluwer Academic Publishers.

Atasu, A., Sarvary, M., & Van Wassenhove, L. N. (2008). Remanufacturing as a marketing strategy. *Management Science, 54*, 1731–1746.

Bloemhof-Ruwaard, J., Van Wassenhove, L. N., Gabel, H., & Weaver, P. (1996). An environmental life cycle optimization model for the European pulp and paper industry. *Omega, 24*, 615–629.

Debo, L. G., Toktay, L. B., & Van Wassenhove, L. N. (2005). Market segmentation and production technology selection for remanufactured products. *Management Science, 51*, 1193–1205.

Ferguson, M. E., Guide, V. D. R., Jr., & Souza, G. C. (2006). Supply chain coordination for false failure returns. *Manufacturing & Service Operations Management, 8*, 376–393.

Ferguson, M. E., & Souza, G. C. (Eds.). (2010). *Closed-loop supply chains: New developments to improve the sustainability of business practices*. Boca Raton: CRC Press.

Ferguson, M. E., & Toktay, B. (2006). The effect of competition on recovery strategies. *Production and Operations Management, 15*, 351–368.

Ferrer, G., & Swaminathan, J. (2006). Managing new and remanufactured products. *Management Science, 52*, 15–26.

Fleischmann, M., Beullens, P., Bloemhof-Ruwaard, J., & Van Wassenhove, L. N. (2001). The impact of product recovery on logistics network design. *Production and Operations Management, 10*, 156–173.
Hauser, W., & Lund, R. (2003). *The remanufacturing industry: Anatomy of a giant*. Boston: Boston University, Department of Manufacturing Engineering Report.
Pochampally, K. K., Nukala, S., & Gupta, S. (2008). *Strategic planning models for reverse and closed-loop supply chains*. Boca Raton, FL: CRC Press.
Souza, G. (2008). Closed-loop supply chains with remanufacturing. In Z. L. Chen & R. Raghavan (Eds.), *Tutorials in operations research*. Hanover, MD: INFORMS.

Cluster Analysis

Jay E. Aronson[1] and Lakshmi S. Iyer[2]
[1]The University of Georgia, Athens, USA
[2]The University of North Carolina at Greensboro, Greensboro, NC, USA

Introduction

Cluster analysis is a generic term for various procedures that are used objectively to group entities based on their similarities and differences. In applying these procedures, the objective is to group the entities (elements, items, objects, etc.) into mutually exclusive clusters so that elements within each cluster are relatively homogeneous in nature while the clusters themselves are distinct. The key purposes of cluster analysis are reduction of data, data exploration, determination of natural groups, prediction based on groups, classification, model fitting, generation and testing of hypotheses (Everitt 1993; Aldenderfer and Blashfield 1984; Lorr 1983).

Due to the importance of clustering in different disciplines such as psychology, zoology, botany, sociology, artificial intelligence and information retrieval, a variety of other names have been used to refer to such techniques: Q-analysis, typology, grouping, clumping, classification, numerical taxonomy, and unsupervised pattern recognition (Everitt 1993). In fact, as Jain and Dubes (1988) noted: "I.J. Good (1977) has suggested the new name botryology for the discipline of cluster analysis, from the Greek word for a cluster of grapes."

Though clustering techniques have existed for many years, profuse work in this area has been accomplished only in the past two decades. The primary stimuli for this were the founding of the Classification Society in 1970 and the publication of the *Principles of Numerical Taxonomy* by Sneath and Sokal (1973; also see, Lorr 1983). Other reasons for the rapid growth in cluster analysis literature are the basic importance of classification as a scientific procedure, prolific developments in high-speed computers, and the need to solve large, real-world problems efficiently. The complexity of clustering methods are known to increase tremendously with increase in problem sizes. With the availability of sophisticated computing power, the handling of large practical problems is of less concern now.

Applications of Cluster Analysis

Clustering methods are applied in a variety of fields including psychology, biology, medicine, economics, marketing research, pattern recognition, weather prediction, environmental science, linguistics, information systems design, electronic brainstorming and flexible manufacturing systems. Some interesting cluster analyses include analyzing large engineering records collections (Homayoun 1984), measuring welfare and quality of life across countries (Hirschberg et al. 1991), management of cutting tools in flexible manufacturing systems (DeSouza and Bell 1991), clustering as a quality management tool (Spisak 1992), identifying the structure and content of human decision making (Allison et al. 1992), mapping consumers' cognitive structures (Hodgkinson et al. 1991), information systems design (Aronson and Klein 1989; Karimi 1986; Klein and Aronson 1991), vehicle routing, production scheduling and sampling (Romesburg 1984), income tax bracket determination (Mulvey and Crowder 1979), management team construction, and idea grouping to handle information overload in electronic brainstorming. The maximum diversity problem forms clusters based on maximizing the differences (distances) among the items rather than the similarities. Applications include forming a single, diverse group from a larger set in which the objective function is imposed only on those items in the group (Kuo et al. 1993), and multiple diverse groups consisting of all items (Weitz and Lakshminarayanan 1997, 1998). Punj and Stewart (1983) provide a good description of the applications of cluster analysis,

including some details on the various clustering packages and programs that are available.

Clustering Techniques

Authors such as Everitt (1993), Cormack (1971), Aldenderfer and Blashfield (1984), Hartigan (1975), and Anderberg (1973) have provided good reviews on existing clustering methods. Nevertheless, there has been no unique classification of the various clustering methods. In fact, this is one of the pitfalls of cluster analysis. Due to work by Cormack (1971), Punj and Stewart (1983), and Everitt (1993), the following five categories have been accepted as a basis:

1. Hierarchical methods,
2. Optimization techniques,
3. Density search techniques,
4. Clumping methods, and
5. Other techniques.

Hierarchical methods: Hierarchical procedures are tree-like structures in which elements are first separated into broad classes. These classes are further subdivided into smaller classes and so on until the terminal classes are not further subdivisible. These methods are most frequently used in the biological sciences. The hierarchical methods are basically of two types — agglomerative and divisive.

The agglomerative methods begin by making each item its own cluster. In subsequent iterations two or more closest clusters are combined to form a new, aggregate cluster. Eventually, all items are grouped into one large cluster. Hence, these methods are some times referred to as build-up methods (Hair et al. 1987).

In contrast to agglomerative methods are the divisive methods that begin with one large cluster. Groups of items that are most dissimilar are removed and placed into smaller clusters. The process continues until each item becomes a one-element cluster. Cormack (1971), Everitt (1993), Aldenderfer and Blashfield (1984) and Hair et al. (1987) have provided comprehensive descriptions of the various agglomerative and divisive procedures.

Optimization techniques: These methods allow relocation of items during the clustering process, improving from an initial solution to optimality. The number of clusters must be decided a priori, although some methods allow for changes (manually or automatically) while solving. There are differences in optimization techniques due to the different methods used for obtaining an initial solution and different objective criteria (Everitt 1993).

Since most of the objective criteria of the optimization techniques are based on those of the well-established statistical concepts, very few mathematical programming approaches have been developed to solve these problems. The statistical methods have proved adequate for many situations, because (1) the solutions found are believed to be reasonably close to the optimum; (2) the solutions typically involve human analyst intervention to determine when an appropriate number of clusters have been identified; and (3) the combinatorial nature of clustering makes it difficult to solve a large problem to a guaranteed optimum. Mulvey and Crowder (1979) developed a subgradient method coupled with a simple search procedure for solving the clustering problem. However, their method did not yield an exact optimum. Though heuristics generally seem efficient, the need to obtain optima to problems such as effective information systems design (Klein et al. 1988) make heuristics less attractive. Klein and Aronson (1991) developed a mixed-integer programming model and method to obtain an optimal solution to clustering problems, where the objective function is the sum of pairwise interactions among all items in each cluster. No metric space nor median are used. Their method is based on the implicit enumeration method of Balas (1965). Extensions including precedence and group size limits are discussed by Aronson and Klein (1989). Earlier, Gower and Ross (1969) and Rohlf (1974) showed that there is a direct relationship between some common cluster formulations and certain types of well-known graph theoretic problems, primarily that of the minimum spanning tree. A further expansion on the use of graph theoretic techniques in cluster analysis may be found in Matula (1977).

Density search techniques: This concept, proposed by Gengerelli (1963), depicts the items as points in a metric space. Parts of the space where the distribution of points is very dense but separated by parts of low density suggest natural clusters. Everitt (1993) describes the different types of density search techniques.

Clumping techniques: These techniques are most popular in language studies where words that tend to have several meanings, when classified based on their

meaning, belong to several groups. Thus, in general, clumping techniques allows for overlapping clusters. This terminology was introduced by Jones, Needham, and co-workers at the Cambridge Language Research Unit (Everitt 1993). This method attempts to partition entities into two groups based on the similarity matrix from the original data. The Needham (1967) criterion is to minimize the cohesion function between the two groups. Other clumping procedures are also discussed in Rohlf (1974) and Everitt (1993).

Other techniques: This comprises all clustering techniques that do not fall in to the above four categories. For example, there is inverse "Q" factor analysis that is commonly used in behavioral sciences (Cattell 1952). The "R" factor analysis is a type of "Q" factor analysis that utilizes the correlations between variables. Gower (1966) provided a good review of the properties of various "Q" and "R" factor analysis techniques. Everitt (1993) and Aldenderfer and Blashfield (1984) included a good summary of various other clustering methods.

Issues of Concern

Though initially the concepts of cluster analysis seem to be intuitive, one can encounter a host of problems while performing an actual analysis. Some of the problems include selection of data units and variables, knowing exactly what to cluster, distance or similarity measures, transformation of measures, clustering criterion, the clustering method to use, the number of clusters and interpretation of the results (Anderberg 1973). Authors such as Aldenderfer and Blashfield (1984), Everitt (1993), Hair et al. (1987) and Anderberg (1973) have addressed some of the issues in great detail. A few of the more critical issues are discused next.

Measurement of distance or similarity matrix: The relationship between elements are represented by using either a similarity or distance measure. While similarity measures (indicating cohesion) take values between 0 and 1, distance measures can be any positive value. The output of any clustering method depends on the type of input measure used. One of the most commonly used measures is the Euclidean distance. This concept can be easily generalized for additional variables (Hair et al. 1987).

Another measure which allows for correlations between variables was originally proposed by Mahalanobis in 1936 (Everitt 1993). This is similar to Euclidean distance measure using standardized variables when the correlations are zero. The Mahalanobis distance measure has been used by McRae (1971). Everitt (1993), Hair Jr et al. (1987) and Hartigan (1975) have provided some discussions on other types of distance measurements. The clustering model for computer-assisted organization presented by Klein and Aronson (1991) accounts for total pairwise interactions independent of a metric. The need to consider all interactions among items in each cluster led to the formulation of a mixed-integer model for optimal clustering based on scaled, pairwise distance (Klein and Aronson 1991).

Which clustering method to use: The problem of choosing an appropriate clustering method generally arises after one has determined the variables, distance measure and criterion for clustering. A number of software packages and programs are available for clustering. Punj and Stewart (1983) and Anderberg (1973) identified some early programs for clustering; now, all major statistical packages contain one or more routines to do such analyses effectively. For selecting the best clustering method one should be aware of the performance characteristics of the various methods (Hair et al. 1987).

Appropriate number of clusters: One of the practical issues of concern in clustering is choosing the number of clusters. Some algorithms find the best fitting structure for a given number of clusters while others, like the hierarchical methods, provide configurations from the number of entities to one large cluster, that is, the entire data set as one cluster. However, if the number of clusters cannot be predetermined, a range of clusters can be selected, solving the problem for each of those cluster sizes, and then selecting the best alternative (Hair et al. 1987).

See

- ▶ Data Mining
- ▶ Decision Making and Decision Analysis
- ▶ Graph Theory
- ▶ Information Systems and Database Design in OR/MS
- ▶ Integer and Combinatorial Optimization
- ▶ Minimum Spanning Tree Problem
- ▶ Vehicle Routing

References

Abonyi, J., & Feil, B. (2007). *Cluster analysis for data mining and system identification*. Basel: Birkhäuser.

Aldenderfer, M. S., & Blashfield, R. K. (1984). *Cluster analysis*. California: Sage Publications.

Allison, S. T., Jordan, A. M. R., & Yeatts, C. E. (1992). A cluster-analytic approach toward identifying the structure and content of human decision making. *Human Relations, 45*, 49–73.

Anderberg, M. R. (1973). *Cluster analysis for applications*. New York: Academic.

Aronson, J. E., & Klein, G. (1989). A clustering algorithm for computer-assisted process organization. *Decision Sciences, 20*, 730–745.

Balas, E. (1965). An additive algorithm for solving linear programs with zero–one variables. *Operations Research, 13*, 517–546.

Cattell, R. B. (1952). *Factor analysis: An introduction and manual for the psychologist and social scientist*. New York: Harper.

Cormack, R. M. (1971). A review of classification. *Journal of the Royal Statistical Society (Series A), 134*, 321–367.

DeSouza, R. B. R., & Bell, R. (1991). A tool cluster based strategy for the management of cutting tools in flexible manufacturing systems. *Journal of Operations Management, 10*, 73–91.

Everitt, B. (1993). *Cluster analysis* (3rd ed.). New York: Halsted Press.

Gengerelli, J. A. (1963). A method for detecting subgroups in a population and specifying their membership. *Journal of Psychology, 5*, 456–468.

Good, I. J. (1977). The botryology of botryology. In J. Van Ryzin (Ed.), *Classification and clustering*. New York: Academic.

Gower, J. C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika, 53*, 325–338.

Gower, J. C., & Ross, G. J. S. (1969). Minimum spanning trees and single linkage cluster analysis. *Journal of the Royal Statistical Society. Series C (Applied Statistics), 18*, 54–64.

Hair, J. F., Jr., Anderson, R. E., & Tatham, R. L. (1987). *Multivariate data analysis* (2nd ed.). New York: Macmillan.

Hartigan, J. A. (1975). *Clustering algorithms*. New York: Wiley.

Hirschberg, J. G., Maasoumi, E., & Slottje, D. J. (1991). Cluster analysis for measuring welfare and quality of life across countries. *Journal of Econometrics, 50*, 131–150.

Hodgkinson, G. P., Padmore, J., & Tomes, A. E. (1991). Mapping consumer's cognitive structures: A comparison of similarity trees with multidimensional scaling and cluster analysis. *European Journal of Marketing, 25*, 41–60.

Homayoun, A. S. (1984). The use of cluster analysis in analyzing large engineering records collection. *Records Management Quarterly, October*, 22–25.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Englewood Cliffs: Prentice Hall.

Karimi, J. (1986). An automated software design methodology using CAPO. *Journal of Management Information Systems, 3*, 71–100.

Klein, G., & Aronson, J. E. (1991). Optimal clustering: A model and method. *Naval Research Logistics, 38*, 447–461.

Klein, G., Beck, P. O., & Konsynski, B. R. (1988). Computer aided process structuring via mixed integer programming. *Decision Sciences, 19*, 750–761.

Kuo, C.-C., Glover, F., & Dhir, K. S. (1993). Analyzing and modeling the maximum diversity problem by zero–one programming. *Decision Sciences, 24*, 1171–1185.

Lorr, M. (1983). *Cluster analysis for social scientists*. California: Jossey-Bass Publishers.

Matula, D. W. (1977). Graph theoretic techniques for cluster analysis algorithms. In J. Van Ryzin (Ed.), *Classification and clustering*. New York: Academic Press.

McRae, D. J. (1971). MICKA, A FORTRAN IV iterative K-means cluster analysis program. *Behavioural Science, 16*, 423–424.

Mulvey, J., & Crowder, H. (1979). Cluster analysis: An application of lagrangian relaxation. *Management Science, 25*, 329–340.

Needham, R. M. (1967). Automatic classification in linguistics. *The Statistician, 17*, 45–54.

Punj, G., & Stewart, D. W. (1983). Cluster analysis in marketing research: Review and suggestions for application. *Journal of Marketing Research, 20*, 134–148.

Rohlf, F. J. (1974). Graphs implied by the Jardine-Sibson overlapping clustering methods. *Journal of the American Statistical Association, 69*, 705–710.

Romesburg, H. (1984). *Cluster analysis for researchers*. Belmont, CA: Lifetime Learning Publications.

Sneath, P. H. A., & Sokal, R. R. (1973). *Principles of numerical taxonomy*. San Francisco: W.H. Freeman.

Spisak, A. W. (1992). Cluster analysis as a quality management tool. *Quality Progress, 25*, 33–38.

Weitz, R. R., & Lakshminarayanan, S. (1997). An empirical comparison of heuristic and graph theoretic methods for creating maximally diverse groups, VLSI design, and exam scheduling. *Omega, 25*, 473–482.

Weitz, R. R., & Lakshminarayanan, S. (1998). An empirical comparison of heuristic methods for creating maximally diverse groups. *Journal of the Operational Research Society, 49*, 635–646.

Cobb-Douglas Production Function

► Economics and Operations Research

COEA

Cost and operational effectiveness analysis.

See

► Cost Analysis

Coefficient of Variation

The ratio of the standard deviation to the mean of a random variable.

Cognitive Mapping

A graphical notation for capturing concepts in use by decision makers for understanding a problematic situation. Concepts are fixed by reference to polar opposites, and directed arcs indicate perceived causal relationships.

See

► Problem Structuring Methods

Coherent System

► System Reliability

COIN-OR Computational Infrastructure for Operations Research

► Open-Source Software and the Computational Infrastructure for Operations Research (COIN-OR)

Column Generation

A technique that permits solution of very large linear-programming problems by generating the columns of the constraint matrix only when they are needed. It is typically employed when the constraint matrix is too large to be stored, or when it is only known implicitly. Column generation, as imbedded in the revised simplex method, has been used to solve the trim problem and other such problems in which the columns are formed from combinatorial considerations.

See

► Integer and Combinatorial Optimization
► Lagrangian Relaxation
► Trim Problem

Column Vector

One column of a matrix or a matrix consisting of a single column.

See

► Matrices and Matrix Algebra

Combat Model

A model whose object is military combat or some aspect of some combat. Three associated terms are often used as synonyms, but are frequently used to differentiate three common aspects of combat modeling: combat model, combat simulation, and war game. When combat model is used as a discriminator it often is used to mean that the model in question is an analytic combat model.

See

► Analytic Combat Model
► Battle Modeling

Combat Simulation

A type of model whose object is military combat or some aspect of combat. Combat simulation is used as a discriminator to emphasize the time or process aspect of the model in question.

See

► Battle Modeling

Combinatorial Auctions

Karla L. Hoffman
George Mason University, Fairfax, VA, USA

Introduction

The advent of the Internet has led to the creation of global marketplaces in which sales of everything from low-cost used merchandise to billion dollar government procurements are conducted through auctions. This article concentrates on designs where many items are auctioned simultaneously and where bidders have the flexibility to combine the goods into packages. The discussion (1) highlights alternative combinatorial auction designs and provides the reader with multiple references to resources that describe more fully the underlying theory of these designs., and (2) describes the mechanisms used to evaluate the efficacy of such approaches in terms of their efficiency, equity, and cognitive complexity, and presents some examples of the use of combinatorial auctions for high-value government lease rights, as well as the use of such auctions for supply-chain procurement. These auctions require knowledge of both game theory and combinatorial optimization.

General Concepts

Governments throughout the world use auctions to lease the right to explore and extract minerals, fuel, and lumber on government properties, to use the airwaves for mobile or broadcast communications, or to control emissions through cap and trade regulations. In addition, the use of business-to-business auctions (often called supply chain auctions) has become a billion-dollar industry. In each of these cases, the need to be able to bundle buys and sells has resulted in new auction theory and designs that enable the simultaneous selling or buying of items using mechanisms that allow participants to indicate their value for the entire package which may have a greater value than the sum of the items within that package. In addition, such auction designs allow users to specify quantity discounts, to indicate budget constraints on the total procurement, and to define other goals of the auction, e.g. social welfare goals in a government auction. These auction designs are computationally more complex for all participants and require languages that allow bidders to express their willingness to participate at a given price for a collection of objects. Such auctions have been termed combinatorial auctions. There are many books that describe the history of auctions, auction theory and its relationship to game theory, and others that are focused exclusively on combinatorial auction designs. For further reading on the subject, see: McMillan (2002) on the history of markets, Krishna (2002) on auction theory, Steiglitz (2007) on the success and pitfalls of EBAY auctions, Klemperer (2004) on auction theory and practice, and Milgrom (2004) and Cramton et al. (2005) on combinatorial auctions. In this review, only the major topics of the field are described, but multiple references are provided for further reading.

In what follows, one-sided auctions are considered and are restricted to the case where there is a single seller and multiple buyers (two-sided auctions are often referred to as exchanges, see Milgrom (2007), Parkes et al. (2001), and Hoffman and Menon (2010) on exchange designs). Since the multiple-sellers/single-buyer case and the multiple-buyers/single-seller case are symmetric, the discussion emphasizes the latter, but all results follow for either case. The concentration is on auction designs where that there are multiple items being sold. For at least some of the buyers, a collection of items must be procured to have a viable business plan; consideration is given only to auction designs that allow the packaging of collections of items. Such designs can provide greater efficiency, as well as greater revenue to the seller than the sequential selling of items individually. These designs are sufficiently general to allow bidders to express a value on a package where the collection of items may have a value greater than the individual items (i.e. the goods are complements), as well as on a package where a buyer can express a quantity discount for buying more of the good (i.e. the goods are substitutes).

Why are auctions such a popular mechanism for buying and selling valuable objects? With the advent of the Internet, auctions are capable of reaching many more possible participants. Here, the potential buyers wish to determine the minimum price that they must pay given that they must compete with others for the ownership of a good or collection of goods. From the

seller's perspective, submitting goods to an auction may increase the number of buyers, thereby increasing the potential for competitive bidding and higher selling prices. Thus, an auction is a mechanism to determine the market-based price, since the bidders set the price through the competition among the bids. This mechanism is dynamic and reacts to changes in market conditions. The determination of selling prices by an auction is perceived as fairer than if the price were set by bilateral negotiations because all buyers must adhere to the same set of rules. Most importantly, if the rules are well designed, the result will have the goods allocated to the entity that values them the most.

The two basic classes of auctions are described next: (1) sealed bid auctions whereby there is only a single opportunity to provide bids to the auction, and (2) multi-round auctions where bids are taken over a period of time and any high bid can be overtaken whenever a new bid is received that increases the overall revenue to the seller.

Sealed Bid Auctions

One common auction mechanism is the first-price (sealed bid) auction**.** In this design, all bidders submit their bids by a specified date. The bids are examined simultaneously and the auctioneer determines the set of bidders that maximizes the revenue to the seller. The optimization problem that determines a collection of package bids that do not often overlap and produce the maximum revenue is known as the Winner Determination Problem (WDP). Mathematically, the problem can be stated as follows:

$$WDP_{OR} : Max \sum_{b=1}^{\#Bids} BidAmount_b x_b \quad (1)$$

$$subject\ to:$$

$$Ax \leq 1$$

$$x \in \{0, 1\} \quad (2)$$

where x_b is a zero–one variable which indicates whether bid b loses or wins, respectively. A is an n x m matrix with m rows, one for each item being auctioned. Each of the n columns represents a bid where there is a one in a given row if the item is included in the bid and zero otherwise. Constraint set (1) specifies that each item can be assigned at most once. Set (1) constraints are equations when the seller chooses to put a minimum price on each item and is unwilling to sell any item below that price. In this case, there is a set of m bids each with only a single item in the package and a bid price at a price slightly below the minimum opening bid price. In this way, the seller will keep the item rather than allow it to be won by a bidder at less than the opening bid price.

In this formulation of the WDP, the bidder can win any combination of bids, as long as each item is awarded only once; this is referred to as the "OR" language. The problem with this language is that it creates a type of exposure problem, that of winning more than the bidder can afford. When multiple bids of a single bidder can be winning, it is incumbent on the software to highlight the maximum exposure to the bidder. This calculation requires that a combinatorial optimization problem be solved for each bidder that calculates the dollar exposure, creating new computational issues for the auctioneer and may result in packages that are not best for the bidder.

The most natural alternative to this "OR" language is the "XOR" language. In this case, the user supplies every possible combination of bids of interest along with a maximum bid price that she is willing to pay for that package. This language removes the dollar exposure problem, since the maximum number of bids that a bidder can possibly pay is the highest bid amount of any of its bids. The problem with the XOR language is that it places a new burden on the bidder: the bidder is forced to enumerate all possible combinations of packages of interest and their associated values. Clearly, as the number of items in an auction increase, the number of possible bids goes up exponentially. When the XOR bidding language is used the Winner Determination Problem (WDP_{XOR}) becomes:

$$WDP_{xor} : Max \sum_{b=1}^{\#Bids} BidAmount_b x_b \quad (3)$$

$$subject\ to:$$

$$x = 1$$

$$\sum_{b \in S_B} x_b \leq 1 \text{ for each bidder B} \quad (4)$$

$$x_b \in \{0, 1\} \quad (5)$$

Where S_B is the set of bids of bidder B, and constraint set (4) specifies that at most one of these bids can be in the winning set.

Fujishima et al. (1999) proposed a generalization of the OR language that does not require the enumeration of all possible combinations. They label this language OR*. Here, each bidder is supplied dummy items (these items have no intrinsic value to any of the participants). When a bidder places the same dummy item into multiple packages, it tells the auctioneer that the bidder wishes to win at most one of these collections of packages. This language is fully expressive, as long as bidders are supplied sufficient dummy items. This language is also relatively simple for bidders to understand and use, as was shown in a Sears Corporation supply-chain transportation auction. In that auction, all bids were treated as "OR" bids by the system. Some bidders cleverly chose a relatively cheap item to place in multiple bids thereby making these bids mutually exclusive, Ledyard et al. (2002). There have been a number of alternative bidding languages that have been proposed; see Fujishima et al. (1999), Nisan (2000), Boutilier and Hoos (2001), and Boutilier et al. (2001) for descriptions of alternative languages.

One serious flaw in a first-price sealed-bid design is that the bidder can experience what is referred to as the winner's curse, i.e., the winning bidder may pay more than was necessary to win since the second highest bid price was far less than the winning bid amount. For this reason, sealed-bid first price auctions encourage bidders to shave some amount off of the bid price. From a game-theoretic perspective, one wants an auction design that encourages straight-forward honest bidding.

An alternative that overcomes this problem is the second price (sealed bid) auction whereby the bidder that has submitted the highest bid is awarded the object (package), but the bidder pays only slightly more (or the same amount) as that bid by the second-highest bidder. In second price auctions with statistically independent private valuations, each bidder has a dominant strategy to bid exactly his valuation. The second price auction also is often called a Vickrey auction (1961).

In a second-price auction, one solves the same winner determination problem as one does for the first-price sealed-bid case, but the winners do not necessarily pay what they bid. Instead, one determines the marginal value to the seller of having this bidder participate in the auction. To do this, for each winning bidder, one calculates the revenue that the seller would receive when that bidder participates in the auction and when that bidder does not, i.e. when none of the bids of this bidder are in the winner determination problem. The difference in the two objective function values is known as the Vickrey-Clarke-Groves discount, named after the three authors, Vickrey (1961), Clarke (1971), and Groves (1973). Each of these authors wrote separate papers producing certain attributes that this auction design has as it relates to incentivizing bidders to reveal their truth value of the goods demanded, and the bidder pays the bid price minus the discount. When winners pay this amount, the auction is known as the Vickrey-Clarke-Groves (VCG) Mechanism.

Although it can be shown that the VCG mechanism encourages truthful bidding, it is almost never used in practice. For a complete list of reasons for it being impractical, see Ausubel and Milgrom (2006) and Rothkopf (2007). In essence, the prices provided by this mechanism may be very low. Worse yet, when items have complementary values, i.e. the package is worth more to the bidder than the sum of the values of the individual items, the outcome may price the items so low that there is a coalition of bidders that would prefer to renege on the auction and negotiate privately with the seller, and the seller may respond by reneging on the sale since both the seller and the coalition of buyers will be better off. Ausubel and Milgrom (2002) argue that prices should be set high enough so that no such coalitions exist. In game theoretic terms, the prices are set such that the outcome is in the core of a coalitional game. These authors introduced an auction design known as the ascending proxy auction in which the bidders provide all bids as if in a sealed-bid auction. Each bidder is provided with a proxy that bids for the bidder in a straightforward manner during an ascending auction. The proxy only announces bids to the auctioneer that maximize the bidder's profit, (i.e. bid price minus announced price) in any given round. The auction continues as an ascending package-bidding auction until, in some round, there are no new bids. Thus, the auction simulates, through proxy bidders, an ascending auction where the increment in each round is infinitesimally small and each bidder, through the use of its proxy, bids in a straight-forward manner.

This auction design is very similar to the iBundle design of Parkes and Ungar (2000).

Hoffman et al. (2005) provide a computational approach toward speeding up the calculations associated with this proxy auction design, and Day and Raghavan (2007) provide an elegant mechanism to obtain minimal core prices directly. The direct mechanism of Day and Raghavan sequentially solves winner determination problems to determine losing coalitions that could supply more revenue to the seller at the current prices. When the solution to this optimization problem yields revenue greater than what the VCG mechanism would provide, the prices of the winning bid set are raised so that the total price paid by winning bidders is equal to this new revenue. To determine these new prices, one must be sure that any winning bidder that forms part of this blocking coalition does not have its price raised from its prior price since it would not be willing to join a coalition if it were to lose revenue relative to its prior offer by the seller. The algorithm is an iterative cutting plane algorithm that forces the prices higher at each iteration until one can find no coalition that can increase revenue to the seller. Therefore, the algorithm finds prices for each winning bidder that are in the core. Since there may be many such minimum core prices, Day and Milgrom (2008) suggest that, in order to encourage sincere bidding, one choose the minimum core prices that are closest in Euclidean distance from the VCG prices. Alternatively, Erdil et al. (2009) argue for a different set of minimum core prices that are based "on a class of 'reference rules' in which bidders' payments are, roughly speaking, determined independently of their own bids as far as possible."

These core-selecting second-price sealed-bid mechanisms have the following properties: They are in the core, they eliminate the exposure problem, and they encourage bidders to bid sincerely. As with all sealed-bid auctions, they make collusion and punishment for not adhering to tacit agreements extremely difficult.

There are, however, negatives associated with this auction, as well as for all sealed-bid auction designs, in that it puts a significant burden on the bidders. Each bidder needs to assess, for every possible combination of items, whether it is a package of interest and then, for all such packages, determine the maximum it is willing to pay. In addition, such mechanisms do not provide any information about how the packages submitted might fit with packages submitted by other bidders. To overcome these problems, a number of authors have suggested simultaneous ascending combinatorial auction designs that allow users price information during the auction.

Multi-round Auctions

Often the value of the good or package of goods being auctioned is not completely known and/or private. Instead, there is a common component to the bid value, that is, the value of the item is not independent of the other bidders, but rather there is a common underlying value as well. In such situations, each agent has partial information about the value. Many high-stakes auctions, such as government auctions for spectrum, oil exploration, and land use, fall into this class. In the case of package-bidding auctions, when there is a common component and bidders want to assess how much others are willing to pay for that item or package of items, the auction is usually an ascending auction with multiple rounds. A round consists of a given time period where bidders have the opportunity to submit new bids. When the round ends, all bids are collected and the winner determination problem is solved. This optimization problem determines the packages that provide the seller with the maximum revenue. The bids that are in the winning set are labeled "provisionally winning," i.e. they would be winning if the auction ended in this round. Thus, in an ascending combinatorial auction, all items are sold simultaneously and a bidder can bid on any collection of items in a given round. To overcome the current set of provisionally winning bids, a bidder must submit a bid that increases the total revenue to the seller.

There are a number of design question that must be answered to have a complete combinatorial auction design:

1. How does the auction end?
2. Must bidders participate in every round?
3. Are bids from previous rounds part of the bids considered by the winner determination problem?
4. How are the prices set in each round?
5. What do bidders know about the bids of other bidders?
6. What other rules might be necessary to ensure that collusion is avoided, to make reneging costly, and to encourage bidders to act truthfully?

Of importance is how to assure that the auction ends in a reasonable period of time and that price discovery (the main reason for a multi-round auction) is accomplished. Most package-bidding auctions have discrete time periods, called rounds, and in each round, the auctioneer provides a price to the user that is the minimum price that the bidder must supply in order to place a new bid. One can choose either a fixed stopping rule or a stopping rule that is determined dynamically. A fixed time stopping rule specifies that the auction will end at a given time. With a fixed stopping time, bidders are encouraged to not provide any bids until the very last seconds of the auction, called sniping. The purpose of sniping is to give other bidders no chance of responding to an offer. In this way, a bidder can acquire price information from other bidders but does not reciprocate, since throughout most of the auction, the bidder is silent. If all bidders chose to snipe and provide no bids until the end of the auction, the auction essentially becomes a first-price sealed-bid auction. To overcome the problem of sniping and to encourage price discovery, most package bidding auctions use an alternative stopping criteria whereby the auction ends when no new bids are presented within a round.

Often, for high-stakes multi-round auctions, there are also activity rules that require a bidder to bid in a consistent way throughout the auction. Activity rules force bidders to maintain a minimum level of bidding activity to preserve their eligibility to bid in the future. Thus, a bidder desiring a large quantity at the end of the auction (when prices are high) must bid for a large quantity early in the auction (when prices are low). If the bidder cannot afford to bid on a sufficient number of items to maintain current eligibility, then eligibility will be reduced so that it is consistent with current bidding. Once eligibility is decreased, it can never be increased. As the auction progresses, the activity requirement increases, reducing a bidder's flexibility. The lower activity requirement early in the auction gives the bidder greater flexibility in shifting among packages early on when there is the most uncertainty about what will be obtainable. Precisely how the activity and eligibility rules are set matters and must be depend upon the type of auction – the value of the items being auctioned, the projected length of the auction, the number of participants, etc. In many high-stakes auctions, such as spectrum or electricity, these activity rules have proven highly successful, Klemperer (2002), McMillan (2002), and Milgrom (2004).

In an ascending multi-round auction design, the auctioneer must provide information about the current value of each package. This information is used for two related purposes: (1) to specify the minimum bid for each item or package in the next round and (2) to provide valuation information to bidders so that they can determine what might be required for a bid to be winning in a subsequent round. While pricing information is easy to ascertain in single item auctions or in simultaneous multi-round auctions without package bidding, (i.e. where bids can be placed on only single items), pricing information for combinatorial auctions is not well defined. Bidders provide only aggregate package prices without providing the information about how each of the individual components that made up the bundle contributes to the overall price. Attempting to disaggregate these bundles into single item prices unambiguously is not possible. Also, since there are many ways that some bundle might partner with other packages to create a winning set, determining the minimal cost partnering for a given package by a given bidder is a complex problem.

To further complicate the pricing issue, bidders may view certain items as substitutes and other items as complements. In the case where items are substitutes, bidders are likely to express sub-additive values for their packages. That is, the value of a package of items is less than or equal to the sum of the values of the items that make up the package. In the complementary case, bidders are likely to express super-additive values for packages. In this case, the value of a package of items is greater than or equal to the sum of the values of the items that make up the package. When items can be both substitutes and complements for bidders, providing unambiguous, complete and accurate price information is an unsolved problem. The non-convex nature of the problem means that the linear prices (i.e. the sum of a package is equal to the sum of the individual items that make up the package) that can be obtained from dual prices from the linear relaxation of the WDP problem will overestimate the true values of the items. In most auctions, one adjusts the dual prices so that the prices are modified so that when one sums the items in each of the winning packages, the prices on those packages exactly equal the prices bid by the provisionally winning bidders (i.e. the winners at the end of the current round). Rassenti et al. (1982) terms these prices pseudo-dual prices.

(For theoretical issues with duals associated with non-convex problems see Wolsey(1981), and for non-anonymous non-linear prices see deVries and Vohra (2003) and Bikhchandani and Ostroy (2002).

Although linear pricing cannot accommodate all aspects of the pricing associated with the non-linear, non-convex, winner determination problem, there are still good reasons for considering its use for determining future bid requirements. First, even perfect pricing is only correct when all other aspects of the problem remain fixed, i.e. when bid amounts remain the same on all other bids and when no new bids are submitted. Second, a dual price associated with a given constraint is only correct when one changes this single restriction (the right-hand-side of the associated constraint) by a very small amount. In the case of combinatorial auctions, the item is either won or it is not. Changes to a constraint would either remove the item entirely from consideration or create a second identical item. Thus, even non-linear, non-anonymous pricing has serious limitations in the context of the winner determination problem since the removal of a single item from the auction (e.g. the removal of the New York City market from consideration in a nationwide spectrum auction) may change the willingness of bidders to participate.

Finally, in an ascending bid auction, bidders need pricing information that is easy to use and understand, and is perceived to be fair. In this situation, easy to use means that bidders can quickly compute the price of any package, whether or not it had been previously bid. Often, bidders want to know what it would take for such a bid to be competitive, i.e. have some possibility of winning in the next round. Bidders may also perceive such prices to be fair since all bidders must act on the same information. Linear prices are likely to move the auction along and deter such gaming strategies as parking (parking is an approach whereby the bidder bids on packages that currently have very low prices knowing that these packages have a very low probability of winning). Bidding on such low-priced packages allows a bidder to maintain eligibility (by maintaining activity), while hiding interest in the packages that are really desired until later in the auction). Thus, virtually all ascending combinatorial auctions use pseudo-dual pricing. For more on alternative pricing within this general framework and the testing thereof, see (Dunford et al. (2003), Bichler et al. (2009) and Brunner et al. (2011).

In 1999, DeMartini et al. proposed an auction design labeled The Resource Allocation Design or RAD where the WDP is solved each round and all losing bidders can only bid on packages where the package price is the sum of the pseudo-dual prices plus some increment (as announced by the auctioneer). There is no activity rule for this auction design. In 2002, the Federal Communications Commission (FCC) announced a similar package bidding design but proposed refinements to the pseudo-price calculations that attempts to limit fluctuations (both positive and negative) in prices. A related design was proposed by Bichler et al. (2009) and is called the Approximate Linear Pricing Scheme (ALPS). It also uses similar rules but chooses the ask price to better balance prices across items. Note that all of these pricing procedures allow prices to both increase and decrease depending upon the packages that are in the winning set. In virtually all of these designs, any bid submitted in any round is considered active throughout the auction. This rule works well with the XOR language since only one bid of a bidder can be in an optimal set and bidders should be willing to win bids placed in early rounds of the auction, when prices were low. This rule forces bidders to provide sincere bids throughout the auction.

A very different ascending package bidding design was proposed by Porter et al. (2003). It is called the combinatorial clock auction. In this design, the auctioneer provides prices for each unique good (if there are multiple identical items, then the bidder indicates that number of units of that item they desire) based solely on whether there is more demand for the item than for supply; no WDP problem is solved. There is no concept of a provisionally winning bidder. Instead, prices increase whenever demand for a given item is greater than supply. Bidders indicate the single package bid that is best given the per-unit prices announced by the auctioneer. All bidders must rebid on any item that they wish to procure in each round. The only information provided to bidders at the end of each round is the quantity demanded for each item and the price for the next round. As long as demand exceeds supply for at least one item, the price is increased for those items with excess demand. If there are no new bids in a round and supply equals demand, then the auction ends. However, it may happen that when there are no new bids, demand has been reduced to below supply. If this occurs, a WDP is solved using all bids from all rounds. If the computed

prices do not displace any bids from the last round, then the auction ends. Otherwise, the auction resumes with the prices determined by using the pseudo-prices calculated from the WDP. Thus, for most rounds, the computation has been drastically reduced to merely increasing prices by a given increment. Only, when demand has dropped below supply is the WDP solved.

Other approaches are the auction designs that simplify the problem by only allowing a few pre-defined packages (Harstad et al. 1998) for which the WDP is polynomially solvable. This idea of only allowing a certain pre-determined set of packages (called hierarchical packages, Goeree and Holt 2010) was used in the 2009 FCC auction for broadband spectrum that brought over $19B into the U.S. Treasury. In that design, all bids were additive (the OR language applied) and the WDP was solved in linear time. When it is possible, in advance, to understand the needs of the bidders and when the packages most desired can be represented in a hierarchical fashion, then one obtains an auction design that is both simpler and quite efficient. However, if the demand for packages does not take on this hierarchical structure, then imposing such structure on the problem for the sake of computability will likely lead to less efficient outcomes.

Hybrid Designs

Ausubel et al. (2005) have argued for a hybrid design that reduces the computational burden on both the bidder and the auctioneer. Here, one first uses a combinatorial clock design followed by a last round second-price sealed-bid approach. The combinatorial clock is similar to that proposed by Porter et al. (2003) with the further enhancement that bidders who find the increment too high are able to place a bid at a price between the old price and the new price that indicates the maximum amount the bidder is willing to pay for that combination of items. In this way, the efficiency loss due to increment size is lessened. This phase of the auction ends when demand is less than or equal to supply or when demand on most items has trailed off. When demand does not exactly equal supply on all items, a sealed-bid phase is initiated. Here, the ascending proxy auction of Ausubel and Milgrom (2002) is imposed. When these two auction designs are merged, one must be careful that the activity rules work well for both phases of the auction. One wants tight activity rules in the ascending phase of the auction to ensure that the bidders are forced to bid sincerely. However, these rules may need to be relaxed or altered during the final sealed-bid phase or a straightforward bidder may be precluded from providing all of the packages that bidder values during the sealed-bid round. Also, theory dictates that in order to guaranteed an efficient outcome, losing bidders (i.e. bidders who dropped out prior to the final phase) must also provide all of the bids that they value in the final phase. Thus, although this hybrid auction is promising in that it is likely to speed up combinatorial auctions, research is still necessary to better understand how the rules of these two disparate auctions should be set so that they mesh well. For more on testing of this design, see Bichler et al. (2011).

Complexity of Combinatorial Auctions

As the previous discussion illustrates, most combinatorial auction designs requires considerable computation and most of the computational burden falls to the auctioneer. This seems appropriate since the auctioneer wants an auction that allows much participation; bidders should not be required to understand combinatorial optimization in order to participate. In terms of these computations, commercial software, such as CPLEX, GUROBI, or XPRESS have shown their ability to solve such problems in reasonable times (less than 30 minutes). Thus, although there is much in the literature that argues against combinatorial auctions because of the computational burden, the optimization software has proven up to be capable of handling the problems that are currently being considered applicable for this type of auction. For more on the computational issues in computing winner determination problems, see Leyton-Brown et al. (2005) and Bichler et al. (2009).

Since multi-item auctions are complex and require bidders to consider multiple alternative bid options, it is important that the computer software used for communication between the bidder and the auctioneer be easy to use and understand. Good graphical user interfaces help bidders to feel comfortable that they understand the current state of the auction (they have been able to find the current

price information, the items they are winning, the amount of bidding necessary to remain eligible, their dollar exposure based on what they have bid, etc.). The system must also provide easy ways for bidders to input their next moves and confirm that they have provided the system with the correct information. As the use of auctions is spreading, computer interfaces for such processes continue to improve and to provide better ways of displaying information to the users through charts, graphs and pictures. There is likely to be continued improvement in this area.

These tools do not, however, help the bidder determine the optimal combination of items to bundle as a package and the optimal number of packages to supply to the system. Since bidders face the serious problem of determining which bids are most likely to win at prices that are within the their budgets, tools that assist bidders in understanding the state of the auction is important. In both supply-chain auctions and in high-stakes government auctions (such as spectrum auctions), bidder-aided tools are often developed to assist the bidder in determining the package or packages to submit in any given round. In the case of supply-chain auctions, the auctioneer often suggests packages to the suppliers that will fit well with other bidder's bids (e.g. by either adding or removing a single item from the package, or by considering a quantity discount for supplying more of an item). Such tools have been found to be very useful and also computationally tractable; see An et al. (2005), Dunford et al. (2003), and Boutilier et al. (2004). Day and Raghavan (2005) and Parkes (2005) provide alternative ways for bidders to express preferences that do not require that the bidder specify particular packages to the auctioneer.

Applications of Combinatorial Auctions

There are many examples of governments' using auctions for the allocation of valuable assets. In most of these auctions, the government is allocating a good and uses auctions to determine both the price and the allocation. Since 1994, governments throughout the world have been using simultaneous multi-round auctions for the allocation of spectrum. For spectrum, a government has the goal of allocating the good to the entities that value it the most with the hope that the bid cost will encourage the build-out of the services. To assure that there is sufficient competition in the telecommunications industry, the U.S. government has, in the past, set spectrum caps for each region. These auctions have been copied globally and are now the standard way that spectrum is allocated. Recently, a number of different package-bidding designs are being tried including the hierarchical ascending auction, the combinatorial clock auction, or the clock-proxy design. As of 2005, these auctions have resulted in revenues in excess of $200 billion dollars worldwide (Cramton 2005).

Within the power industry, there has also been an evolutionary movement toward auctions for the determination of who can supply power to the electricity grid and at what price. Most of the allocation is determined one day ahead of the demand. The auction reflects the unique characteristics (both physical and structural) of the industry. The allocation is determined by a complicated optimization that evaluates the demands at various nodes of the networks and prices power generation at each such node. The spot market corrects this allocation for any last minute changes due to weather, plant outages, etc. Long term contracts make this process work.

Similarly, auctions have been used to bring market-based forces to control air pollution. Here, a government entity (either nationally run or regionally administered) establishes a fixed number of tradable allowances each of which represents the legal right for its owners to emit a fixed quantity of pollution. A firm holding an allowance can emit the fixed quantity and surrender the allowance to the government, or if the firm can abate its emissions, it can profit by selling the allowance to another polluter than cannot so inexpensively abate emissions. The establishment of the fixed quantity is the cap. The exchange of allowances (credits) between polluters is the trade. See Ellerman et al. (2003) and Tietenberg (2006) for a general overview of cap and trade ideas.

The use of combinatorial auctions for the procurement of goods in services has also been growing. Some of these auctions are sealed-bid auctions, while most are moving toward multi-round auction designs. In such auctions, the providers of the goods and services are pre-screened and are then allowed to provide bids for collections of good and/or services as all or nothing packages. For a general

survey of supply-chain auctions, see Bichler et al. (2006). The three applications described next highlight a few examples to show how such auctions differ from government auctions.

1. The first use of a combinatorial auction within the transportation industry was an auction conducted by Sears. Here, suppliers of freight delivery were allowed to bundle multiple lanes together into a single bid thereby allowing carriers to coordinate multiple businesses and reduce empty or low value backhaul movements. It also provided a means to incorporate surge demand contingencies into the longer (3-year) contracts, thereby lessening the need to renegotiate contracts whenever demands changed; Ledyard et al. (2002).
2. Mars Incorporated used a combinatorial auction mechanism to procure the necessary goods from multiple suppliers allowing bidders to specify complex bid structures that indicated quantity discounts, minimum supply, and multiple goods collected within a single bid. No bidder was allowed to supply more than a certain percentage of the overall quantity needed and newer suppliers were limited more severely than their suppliers they had used over a number of years. The algorithm also assured that there were multiple suppliers in the solution for each critical entity. These auctions are not simple, but work to match the needs of the procurer, Mars, with the capabilities of the suppliers (often farmers). The allocation considers geographic, volume and quality factors. The suppliers liked the auction mechanism because of its transparency, shorter negotiation time and fairness; Honer et al. (2003).
3. Motorola Corporation used auctions for the procurement of the multitude of parts needed for cellular devices. Motorola needed to reduce both the time and the effort required to prepare for and conduct negotiations with its suppliers, simplify their coordination, and optimize contract awards across sectors, in order to save costs; Metty et al. (2005).

Governments are moving toward procuring their goods and services in a similar fashion. One such example is the use of auctions to determine the suppliers of lunches in a large school system. Chile spends around US$180 million a year to feed 1,300,000 students from low income families. To improve the quality of the goods and services being provided to the school system and to save money, the government chose to assign catering contracts in a single-round sealed-bid combinational auction. This auction resulted in a transparent and objective allocation approach, thereby generating competition among firms. It also allowed the companies to build flexible territorial bids to include their scale of economies, leading to more efficient resource allocation. This new methodology improved the price-quality ratio of the meals with yearly savings of around US$40 million, equivalent to the cost of feeding 300,000 children during one year; Epstein et al. (2002).

In supply-chain auctions, rules are designed to assure a certain diversification in suppliers and to assure the reliability of the supply chain. In each case, are goals other than revenue maximization or efficiency that drove the auction design. In addition, the auction design must consider the nature of the investment. For spectrum, where there was both uncertainty in the long-term use of the technologies and where the cost of build-out are high, long-term leases were chosen. For energy, auctions are used for a much shorter decision problem. The U. S. Treasury uses multiple auctions for short, medium and long-term debt allocation. Oil and gas exploration must have a relatively long-term horizon where payments for wildcatting are based on the bid price and a yearly rent, whereas payments for extraction are based on bid price and royalties.

Thus, one must consider carefully the application when designing the allocation mechanism and the payment scheme. Auction theory and its use is growing because of its proven value. It provides price discovery and signals where more capacity is needed. It is often a fairer and more transparent process for the allocation of goods and services.

Concluding Remarks

Combinatorial auctions are appropriate for problems where the bidders need to procure a collection of items that contribute to their having a viable business plan. When evaluating alternative designs, one is likely to want to satisfy the following goals:

1. The property rights are well-defined.
2. Bidders are able to, through their bids, announce the entire collection of objects that they need for a given business plan.

3. The auction results in maximum revenue to the seller.
4. The auction results in an efficient outcome i.e. all items are collectively allocated to the bidders that value these items the most.
5. The auction is perceived as fair to all bidders.
6. The auction ends in a reasonable amount of time.
7. The auction has limited transaction costs, i.e. the rules are not so difficult or the bidding so complicated that a straightforward bidder finds it difficult to participate.
8. The auction cannot be gamed, i.e. truthful bidding is an optimal strategy for all bidders.
9. The auction allows price discovery.
10. The auction is computationally feasible and scalable.

It is not possible to have all such attributes obtain simultaneously. For each applications, some of these goals will be more important than others. One should, however, keep all of these goals in mind when evaluating a mechanism.

In addition, the auction mechanism should consider any application-specific issues that might arise. For example, in government auctions one might want to consider how market power impacts the outcome, whether there will be sufficient participation, and whether the outcome will limit future competition in the industry. In certain situations, there may need to be a transition period that allows the market to adjust to a change in the way rights are allocated; One may have to consider the associated rights that a bidder would need to be able to use the right being sold or leased in the auction; The seller needs to determine if the rights are paid for over time or at the end of the auction; The money obtained may need to be designated for a specific use in order for the government to obtain the approval of all constituents. The auction design may also need to satisfy other social goals specific to the application (e.g. reducing emissions, increasing competition, incentivizing innovation, improving multi-modal transportation). Similarly, in supply chain auctions, a variety of goals need to be considered– quality of the goods, price, historical dependability of the supplier, among others.

See

► Auction and Bidding Models
► Integer and Combinatorial Optimization

References

An, N., Elmaghraby, W. J., & Keskinocak, P. (2005). Bidding strategies and their impact on auctioneer's revenue in combinatorial auctions. *Journal of Revenue and Pricing Management, 3*(4), 337–357.

Ausubel, L. M., Cramton, P., & Milgrom, P. (2005). The clock proxy auction. In P. Cramton, Y. Shoham, & R. Steinberg (Eds.), *Combinatorial auctions* (pp. 113–136). MIT Press.

Ausubel, L. M., & Milgrom, P. (2002). Ascending auctions with package bidding. *Frontiers of Theoretical Economics, 1*, 1–42.

Ausubel, L. M., & Milgrom, P. (2006). The lovely but lonely Vickrey auction. In P. Cramton, Y. Shoham, & R. Steinberg (Eds.), *Combinatorial auctions*. Cambridge, MA: MIT Press.

Bichler, M. (2011). Auctions: Complexity and algorithms. In *Wiley encyclopedia of operations research and management science*. John Wiley and Sons.

Bichler, M., Shabalin, P., & Wolf, J. (2011). *Efficiency, auction revenue, and bidding behavior in the combinatorial clock auction*. Technical Report available from M. Bichler.

Bichler, M., Davenport, A., Hohner, G., & Kalagnanam, J. (2006). Industrial procurement auctions. In P. Crampton, Y. Shoam, & R. Steinberg (Eds.), *Combinatorial auctions* (pp. 593–612). MIT Press.

Bichler, M., Shabalin, S., & Pikovsky, A. (2009). A computational analysis of linear price iterative combinatorial auction formats. *Information Systems Research, 20*(1), 33–59.

Bikhchandani, S., DeVries, S., Schummer, J., & Vohra, R. (2002). Linear programming and Vickrey auctions. In B. Dietrich & R. Vohra (Eds.), *Mathematics of the internet: E-auctions and markets* (pp. 75–115).

Bikhchandani, S., & Ostroy, J. M. (2002). The package assignment model. *Journal of Economic Theory, 107*, 337–406.

Boutilier, C., & Hoos, H. H. (2001). Bidding languages for combinatorial auctions. *Seventh International Joint Conference on Artificial Intelligence (IICAI-01)*, 1211–1217.

Boutilier, C., Sandholm, T., & Shields, R. (2004). Eliciting bid taker non-price preferences in "Combinatorial Auctions". In V. Khu-Smith & C. J. Mitchell (Eds.), *Proceedings of the national conference on artificial intelligence* (pp. 204–211). San Jose, CA.

Brunner, C., Goeree, J. K., Holt, C. H., & Ledyard, J. O. (2011). An experimental test of flexible combinatorial spectrum auction formats. *American Economic Journal: Microeconomics, 2*, 39–57.

Cason, T. N. (1993). Seller incentive properties of EPA's emission trading auction. *Journal of Environmental Economics and Management, 25*, 177–195.

Clarke, E. (1971). Multipart pricing of public goods. *Public Choice, 8*, 19–33.

Cramton, P. (2005). Simultaneous ascending auctions. In P. Cramton, Y. Shoham, & R. Steinberg (Eds.), *Combinatorial auctions* (pp. 99–114). MIT Press.

Cramton, P., Shoham, Y., & Steinberg, R. (Eds.). (2005). *Combinatorial auctions* (pp. 99–114). MIT Press.

Day, R., & Milgrom, P. (2008). Core-selecting package auctions. *International Journal of Game Theory, 36*(3), 393–407. Springer.

Day, R. W., & Raghavan, S. (2007). Fair payments for efficient allocations in public sector combinatorial auctions. *Management Science, 53*(9), 1389–1406.

Day, R., & Raghavan, S. (2005). *Assignment preferences and combinatorial auctions*. Working paper, Operations and information management school of business, University of Connecticut. http://users.business.uconn.edu/bday/index.htm

DeMartini, C., Kwasnica, A. M., Ledyard, J. O., & Porter, D. (1999). *A new and improved design for multi-object iterative auctions*, Social Working Paper. Pasadena, CA: Division of the Humanities and Social Sciences, California Institute of Technology.

DeVries, S., & Vohra, R. (2003). Combinatorial auctions: A survey. *INFORMS Journal on Computing, 15*(3), 284–309.

Dunford, M., Hoffman, K., Menon, D., Sultana, R., & Wilson, T. (2003). *Price estimates in ascending combinatorial auctions*, Technical Report. Fairfax, VA: George Mason University, Systems Engineering and Operations Research Department.

Ellerman, A. D., Joskow, P. L., Montero, J., Schmalensee, R., & Bailey, E. M. (2000). *Markets for clean air: The U.S. acid rain program*. Cambridge University Press.

Ellerman, A. D., David, H., & Paul L. J. (2003). *Emissions Trading: Experience, Lessons, and Considerations for Greenhouse Gases*. Washington, D.C.: Pew Center for Global Climate Change.

Epstein, R., Henriquez, L., Catalan, J., Weintraub, G., & Martinez, C. (2002). A combinatorial auction improves school meals in Chile. *Interfaces, 32*(6), 1–14.

Erdil, A., Klemperer, P., Cramton, P., Dijkstra, G., Goeree, J., Marszalec, D., Meyer, M., Milgrom, P., Pagnozzi, M., & Parkes, D. C. (2009). *A new payment rule for core-selecting package auctions*. Technical report available on Paul Klemperer's website.

Friedman, D., & Rust, J. (Eds.). (1993). *The double auction market: Institutions, theories and evidence* (Santa Fe Institute studies in the sciences of complexity, Vol. XIV). Addison Wesley.

Fujishima, Y., Leyton-Brown, K., & Shoham, Y. (1999). Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. *Proceedings of IJCAI 1999*, 548–553.

Goeree, J. K., & Holt, C. A. (2010). Hierarchical package bidding: A paper & pencil combinatorial auction. *Games and Economic Behavior, 70*(1), 146–169.

Groves, T. (1973). Incentives in teams. *Econometrica, 41*, 617–631.

Harstad, R., Pekec, A., & Rothkopf, M. H. (1998). Computationally manageable combinatorial auctions. *Management Science, 44*, 1131–1147.

Hoffman, K., Menon, D., van den Heever, S. A., & Wilson, T. (2005). Observations and near-direct implementations of the ascending proxy auction. In P. Cramton, Y. Shoham, & R. Steinberg (Eds.), *Combinatorial auctions* (pp. 415–450). MIT Press.

Hoffman, K., & Menon, D. (2010). A practical combinatorial clock exchange for spectrum licenses. *Decision Analysis, 7*(1), 58–77.

Hoffman, K., Menon, D., & van Den Heever, S. A. (2008). A package bidding tool for the FCC's spectrum auctions and its effect on auction outcomes. *Telecommunications Modeling Policy and Technology: Operations Research/ Computer Sciences Interfaces Series, 44*, 153–189.

Holt, C. A., Shobe, W., Burtraw, D., Palmer, K., & Goeree, J. (2007). *Auction design for selling CO_2 emission allowances under the regional greenhouse gas initiative. Regional greenhouse gas initiative*. Technical Report to RGGI.

Honer, G., Rich, J., Ng, E., Reid, G., Davenport, A., Kalagnanam, J., Lee, H. S., & An, C. (2003). Combinatorial and quantity discount procurement auctions benefit mars, incorporated and its suppliers. *Interfaces, 33*(1), 23–35.

Klemperer, P. (1999). Auction theory: A guide to the literature. *Journal of Economic Surveys, 13*(3), 227–286.

Klemperer, P. (2002). What really matters in auction design. *Journal of Economic Perspectives, 16*, 169–189.

Klemperer, P. (2004). *Auctions: Theory and practice* (The toulouse lectures in economics). Princeton, NJ: Princeton University Press.

Koboldt, C., Maldoom, D., & Marsden, R. (2003). *The first combinatorial spectrum auction*. Ofcom Technical Report describing the results of the 2003 Nigerian Spectrum Auction, available on of com website.

Krishna, V. J. (2002). *Auction theory*. Academic Press, 200pp.

Kwasnica, A. M., Ledyard, J. O., Porter, D., & DeMartini, C. (2005). A new and improved design for multi-object iterative auctions. *Management Science, 51*, 419–4234.

Ledyard, J. O., Olson, M., Porter, D., Swanson, J. A., & Torma, D. P. (2002) The first use of a combined value auction for transportation services. *Interfaces, 32*, 4–12.

Lehmann, D., Mueller, R., & Sandholm, T. (2005). The winner determination problem. In P. Cramton, Y. Shoham, & R. Steinberg (Eds.), *Combinatorial auctions*. Cambridge, MA: MIT Press.

Leyton-Brown, K., Nudelman, E., & Shoham, Y. (2005). Empirical hardness models. In P. Cramton, Y. Shoham, & R. Steinberg (Eds.), *Combinatorial auctions* (pp. 479–503). MIT Press.

McMillan, J. (2002). *Reinventing the bazaar: A natural history of markets*. Norton Press, 278pp.

Metty, T., Harlan, R., Samelson, Q., Moore, T., Morris, T., & Sorenson, R. (2005). Reinventing the supplier negotiation process at motorola. *Interfaces, 35*(1), 7–23.

Milgrom, P. (2004). *Putting auction theory to work*. Cambridge Press, 368pp.

Milgrom, P. (2007). Package auctions and exchanges. *Econometrica, 75*(4), 935–965.

Nisan, N. (2000). Bidding and allocation in combinatorial auctions. *Proceedings of the 2nd ACM Conference on Electronic Commerce*, 1–12.

O'Neill, R. P., Helman, U., Hobbs, B., Stewart, W. R., & Rothkopf, M. (2007). The joint energy and transmission rights auction: A general framework for RTO market designs. *Power Engineering Review, IEEE, 22*(10), 59–68.

Parkes, D. C. (2005). Auction design with costly preference elicitation. *Annals of Mathematics and AI, 44*, 269–302.

Parkes, D. C., Kalagnanam, J., & Eso, M., (2001). Achieving budget-balance with Vickrey-based payment schemes in combinatorial exchanges. *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI-01)*, 1161–1168.

Parkes, D. C., & Ungar, L. H. (2000). Iterative combinatorial auctions: Theory and practice. *Proceedings of the 17th National Conference on Artificial Intelligence* (AAAI-00), 74–81.
Pekec, A., & Rothkopf, M. H. (2006). Non-computational approaches to mitigating computational problems in combinatorial auctions. In P. Cramton, Y. Shoham, & R. Steinberg (Eds.), *Combinatorial auctions*. M.I.T. Press.
Porter, D., Rassenti, S., Roopnarine, A., & Smith, V. (2003). Combinatorial auction design. *Proceedings of the National Academy of Sciences, 100*(19), 11153–11157.
Porter, D., & Smith, V. (2006). FCC license experiment design: A 12-year experiment. *Journal of Law Economics and Policy, 3*, 63–80.
Rassenti, S., Smith, V., & Bulfin, R. I. (1982). A combinatorial mechanism for airport time slot allocation. *Bell Journal of Economics, 13*, 402–417.
Rothkopf, M. H. (2007). Thirteen reasons why the Vickrey-Clarke-Groves process is not practical. *Operations Research, 55*(2), 191–197.
Steiglitz, K. (2007). *Snipers, shills and sharks: eBay and human behavior*. Princeton University Press, 298pp.
Tietenberg, T. H. (2006). *Emissions trading: Principles and practice* (2nd ed.). Washington: RFF Press.
Vickrey, W. (1961). Counter-speculation, auctions and competitive sealed tenders. *Journal of Finance, 16*, 8–37.
Wolsey, L. A. (1981). Integer programming duality: Price functions and sensitivity analysis. *Mathematical Programming, 20*(1), 173–195.
Wurman, P. R., & Wellman, M. P. (1999). *Equilibrium prices in bundle auctions*, Sante Fe Institute Working Papers (Paper: 99-09-064).

Combinatorial Explosion

The phenomenon associated with optimization problems whose computational difficulty increases exponentially with the size of the problem. One common paradigm is the traveling salesman problem.

See

- ▶ Combinatorics
- ▶ Curse of Dimensionality
- ▶ Integer and Combinatorial Optimization
- ▶ Traveling Salesman Problem

Combinatorial Optimization

- ▶ Integer and Combinatorial Optimization

Combinatorics

Eugene L. Lawler

Combinatorics is the branch of mathematics that deals with arrangements of objects, usually finite in number. The term arrangement encompasses, among other possibilities, selection, grouping, combination, ordering or placement, subject to various constraints.

Elementary combinatorial theory concerns permutations and combinations. For example, the number of permutations or orderings of n objects is $n! = n(n-1) \ldots (2)(1)$, and the number of combinations of n objects taken k at a time is given by the binomial coefficient $\binom{n}{k} = n!/[k!(n-k)!]$. In order to compute the probability of throwing a seven with two dice, or of drawing an inside straight at poker, one must be able to count permutations and combinations, as well as other types of arrangements. Indeed, combinatorics is said to have originated with investigations of games of chance. Combinatorial counting theory is the foundation of discrete probability theory as is exists today.

Experimental design provides the motivation for another classic area of combinatorial theory. Suppose five products are to be tested by five experimental subjects over a period of 5 days, with each subject testing one product per day. Labeling the subjects A, B, C, D, E, the products 1,2,3,4,5, and the days M, Tu, W, Th, F, one way to schedule the tests is as follows:

	M	*Tu*	*W*	*Th*	*F*
1	A	B	C	D	E
2	B	C	D	E	A
3	C	D	E	A	B
4	D	E	A	B	C
5	E	A	B	C	D

A square array of symbols, with each symbol occurring in each row exactly once and in each column exactly once, is called a Latin square.

Now suppose each of the tests is to be performed by a subject in the presence of an observer. In order to reduce the effects of bias due to subject-observer interactions, the Latin square should represent the

schedule for the subjects to be combinatorially orthogonal to the Latin square for the observers. This means that when the two Latin squares are superimposed, each of the 25 possible subject-observer pairs appears exactly once in the resulting array, called a Graeco-Latin square. Labeling the observers a, b, c, d, e, a 5 × 5 Graeco-Latin for our experiment is as follows:

Aa	Bb	Cc	Dd	Ee
Bc	Cd	De	Ea	Ab
Ce	Da	Eb	Ac	Bd
Db	Ec	Ad	Be	Ca
Ed	Ae	Ba	Cb	Dc

Leonhard Euler observed that no 2 × 2 Graeco-Latin square exists and found he was able to construct examples of $n \times n$ Graeco-Latin squares for n up to five, but had trouble with six. In 1782 Euler conjectured the nonexistence of such an arrangement for any $n = 4k + 2$, where k is an integer. About 1900, Euler's conjecture was confirmed, by systematic examination of cases, for $n = 6$. However, his more general conjecture remained unsettled until 1959 when Bose, Shirkhande and Parker exhibited a 22 × 22 Graeco-Latin square. Shortly after, these same investigators (Euler's Spoilers) demolished what remained of Euler's conjecture by establishing that Graeco-Latin squares do exist for all n other than two and six. Their work made use of results of number theory, a branch of mathematics with which combinatorics exists in happy symbiosis.

Another investigation of Euler turned out to have considerable importance for combinatorial mathematics. In the old city of Königsberg in Eastern Prussia the River Pregel divided into two branches surrounding an island. The river was spanned by seven bridges. It is said that the people of Königsberg entertained themselves by trying to find a route around the city that would cross each of the bridges exactly once. In 1736, Euler provided a definitive answer to the Königsberg bridge problem, and any related instances: "If there are no more than two areas to which an odd number of bridges lead, then such a journey is not possible. If, however, the number of bridges is odd for exactly two areas, then the journey is possible if it starts in either of these areas. If, finally, there are no areas to which an odd number of bridges leads, then the required journey can be accomplished from any area." This result has been viewed as the oldest theorem of what is now known as graph theory.

With the advent of digital computers and operations research, the emphasis of combinatorics shifted from problems of counting and existence of arrangements to problems of optimization. Modern combinatorics may be said to have come of age with the development of network flow theory by Lester Ford and Ray Fulkerson in the 1950s. This remarkable theory enables a great variety of practical optimization problems to be solved by efficient algorithms. A number of elegant duality results follow directly from Ford and Fulkerson's Max-Flow Min-Cut Theorem. For example, consider the König-Egervary Theorem, which can be stated as follows: Let us call a subset of elements of a matrix independent if no two of the elements lie in the same row or the same column. Let all elements be 0 or 1. Then the maximum size of an independent set of 1 s is equal to the minimum number or rows and columns containing all the 1 s in the matrix.

In the 1960s Jack Edmonds generalized many of the results of Ford and Fulkerson by exploiting the concept of a matroid, a combinatorial structure abstracting the notion of linear independence. Edmonds also developed a general theory of matching in graphs, where a matching is a subset of edges, no two of which are incident to the same vertex. He also proved a generalization of the König-Egervary Theorem, which may be viewed as a duality theorem for matchings in the special case of bipartite graphs.

Edmonds (1965) further observed that the running time of his general matching algorithm was bounded by a polynomial in the size of the graph it is applied to, and made an eloquent argument for the goodness of polynomial-time bounded algorithms. The significance of polynomial time bounds came to be more fully appreciated with the development of NP-completeness theory by Stephen Cook, Richard Karp and Leonid Levin in 1973. The theory of NP-completeness has been an essential tool for researchers in combinatorial optimization ever since.

Algorithms arising from network flow theory, matroid optimization theory, matching theory, or similar theories, may all be viewed as special-purpose linear programming algorithms. Combinatorial duality results, including the Max-Flow Min-Cut Theorem and the König-Egervary Theorem, are most often special cases of linear programming duality. The term applied

to the general paradigm of formulating and solving combinatorial problems by linear programming techniques is polyhedral combinatorics.

More often than not, combinatorial optimization problems that arise in the real world are too idiosyncratic and complicated to be fully tamed by polyhedral techniques alone. For these problems, it is usually necessary to engage in some form of enumeration of cases if one seeks to find a provably optimal solution. The Traveling Salesman Problem (TSP) is prototypical of a difficult (NP-complete) problem with a real-world flavor. In this problem, one is asked to find a shortest closed tour of n cities (visiting each city exactly once, and ending at the starting point), given an $n \times n$ matrix of intercity distances. The number of possible tours is, of course, finite: $(n-1)!$. But for any interesting value of n, say 100 or 1,000, the number of tours is so astronomically large as to be effectively infinite. An exhaustive enumeration of even a tiny fraction of the tours is out of the question. Hence if the TSP is to be solved by enumeration, the enumeration must be very artfully limited.

The TSP has served as a testbed for algorithmic research. Indeed, the approaches that have been applied to the TSP are representative of the full range of techniques of combinatorial optimization. These include polyhedral and integer linear programming, Lagrangian relaxation, nondifferentiable optimization, heuristic and approximation algorithms, branch-and-bound, dynamic programming, neighborhood search, and simulated annealing. With much effort by many investigators, it is today possible to find optimal, or provably near-optimal, solutions to instances of the TSP with hundreds, even thousands of cities.

Combinatorial optimization has assumed great practical importance, in such diverse problem areas as machine scheduling and production planning, vehicle routing, plant location, network design, VLSI design, among many others. The practical and theoretical importance of this field can only be expected to grow in the future.

See

- ▶ Chinese Postman Problem
- ▶ Computational Complexity
- ▶ Graph Theory
- ▶ Integer and Combinatorial Optimization
- ▶ Traveling Salesman Problem

References

Biggs, N. L., Lloyd, E. K., & Wilson, R. J. (1976). *Graph theory* (pp. 1736–1936). London, UK: Oxford Univ Press.

Edmonds, J. (1965). Paths, trees, and flowers. *Canadian Journal of Mathematics, 17*, 449–467.

Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to NP-completeness*. San Francisco: W.H. Freeman.

Graham, R. L., Rothschild, B. L., & Spencer, J. H. (1980). *Ramsey theory*. New York: John Wiley.

Lawler, E. L. (1976). *Combinatorial optimization: Networks and matroids*. New York: Holt, Rinehart and Winston.

Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G., & Shmoys, D. B. (Eds.). (1985). *The traveling salesman problem: A guided tour of combinatorial optimization*. New York: John Wiley.

Lovasz, L. (1979). *Combinatorial problems and exercises*. Amsterdam: North Holland.

Lovasz, L., & Plummer, M. D. (1986). *Matching theory*. Amsterdam: North Holland.

Nemhauser, G. L., & Wolsey, L. A. (1988). *Integer programming and combinatorial optimization*. New York: John Wiley.

Roberts, F., & Tesman, B. (2009). *Applied combinatorics*. New York: CRC Press.

Schrijver, A. (1986). *Theory of linear and integer programming*. New York: John Wiley.

Tucker, A. (2006). *Applied combinatorics* (5th ed.). New York: John Wiley.

Wilson, R. J., & Watkins, J. J. (1990). *Graphs: An introductory approach*. New York: John Wiley.

Common Random Numbers

- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Variance Reduction Techniques in Monte Carlo Methods

Common Value Bidding Model

A bidding model in which the value of what is being auctioned, while unknown at the time of the auction, is known to be the same for all bidders. In such a model, bidders must correct for the selection bias, often called

the winner's curse, caused by the fact that winning bidder is likely to have been the one who most overestimated the value.

See

► Bidding Models

Communications Networks

Edward A. Sykes
Make Systems, Inc., Carey, NC, USA

Introduction

Communications networks are systems of electronic and optical devices that support information exchange among their subscribers. Examples of communications networks are abundant in everyday life: telephone networks, broadcast and cable television networks, and computer communications networks such as the Internet. The impacts of communications networking on the individual, society and the planet are staggering, rivaling that of the tall ship and the automobile. In just under two centuries, humanity has been transformed from myriad villages and towns isolated in obscure corners of the continents to one global information village. This transformation is no more evident than in the fact that the very boundaries between information transfer and information processing are increasingly hard to define. The integration of communications networks, computing technology, and end-user devices (e.g., the telephone, television, personal computer) is increasingly being referred to simply as the information infrastructure. This global transformation of the world community is no more evident than in the Internet, which some analysts predict will be the predominate mechanism for conducting business (both consumer and business-to-business) within 5 years.

OR and MS have been major players in the development, deployment and management of information technologies and infrastructure. Applications of OR/MS in modeling, analysis and design of communications networks are among the oldest of the fields, dating from the late nineteenth and early twentieth century. Among the most notable of all work in OR/MS history is queueing modeling of telephony by A.K. Erlang. Modeling, analysis and design of communications networks, moreover, is an area rich in applications of more generic OR/MS work. Communications networks are, fundamentally, networks and thus, almost all generic discussion of networks applies. Analogous remarks are appropriate: in communications network modeling and analysis for topics such as queueing and queueing networks, simulation, and network reliability; and in communications network design for topics such as facility location, topological design and optimization, capacity optimization and allocation. Finally, communications networking problems have a great deal of commonality with problems arising in other domains, for example, modeling, analysis and design of transportation systems, water resource distribution systems, etc.

A discussion of the wealth of communications networking issues arising in the application of OR/MS techniques would be quite extensive. Here the focus is on several classes of modeling, analysis, and design problems arising in a variety of modern communications technologies.

Basic Structure and Concepts

A typical communications network comprises a set of subscribers that offer subscriber-to-subscriber traffic requirements to be supported on the given network architecture. For example, a typical household (subscriber) makes telephone calls (traffic requirements) to be supported on a voice network switching fabric (architecture). In most communications architectures, a hierarchy of communications devices exist to support traffic, but the most basic of these are customer premises equipment, local access equipment, and switching equipment. Customer premises equipment is associated directly or indirectly with the generation of traffic requirements. Local access equipment provides a means of connecting the subscriber to the network, that is, the interface between the subscriber and the network necessary for traffic to enter the network and be routed over it. Switching equipment routes the traffic from its source subscriber to its destination subscriber.

All three types of equipment are determined by the nature of the traffic requirements and their associated technology and architecture. In a voice (i.e., telephone) network, the customer premises equipment is generally just a telephone – in this case, the subscriber is the household whose aggregate traffic (telephone calls) enters and leaves the network at the telephone. The local access equipment in this case is owned and provided by the local telephone company. Although there typically is switching in the local access in this case (for local calls), for purposes of the discussion here, the long haul switching equipment is owned and provided by a common carrier such as AT&T. Analogous examples can be provided for data communications networks, video teleconferencing networks, etc.

Communications networks differ on the manner in which they carry traffic requirements. Considerable attention has been paid to quality-of-service (QoS) issues in communications networks, with major trends in standards and implementations increasingly focused on assuring that different applications receive the QoS they require across the technologies they traverse. Most voice networks set-up calls from source to destination in a circuit switched manner, that is, dedicating capacity along the entire path of the call. Most data networks segment information into streams of packets or cells which are routed independent from one another from source to destination and reassembled into the original information at the destination. Data networks can operate with or without functions that route traffic according to QoS needs and with or without reservation or dedication of capacity along the path. Many variations and hybrids of these basic approaches exist and the evolution of technology is becoming increasingly toward supporting traffic sources with differing traffic characteristics and differing service requirements differently. For example, voice traffic is error tolerant (one can tolerate a little static on the line) but delay sensitive (one cannot tolerate long delays between the time a word is spoken and the time it is received at the destination). Some data traffic (e.g, file transfer) is typically error intolerant but delay insensitive. Consideration of these kinds of issues is addressed in network modeling and simulation.

A common thread among most network modeling, analysis and design conceptualizations is the view of a network as a graph comprising nodes and links. A node is used to abstractly represent a device location (e.g., a subscriber location or a switching location). A link is used to represent connections between subscribers and switches and between switches. A link typically has a capacity for supporting traffic. One can view a link as analogous to a pipe and the capacity of the link as analogous to the diameter of the pipe, but with one caveat. A communications link of a given capacity typically supports traffic at that capacity in both directions, that is, it is more properly viewed as two pipes of equal capacity in parallel, each flowing in a direction opposite the other. In addition, a single link can support many "logical" entities as well – for example, it can have its bandwidth dedicated in some proportions to support different service classes or priorities for purposes of providing differential QoS to traffic applications. Design of communications networks typically addresses selecting the number of and the placement of backbone (central) nodes, selecting and sizing the links between subscribers and backbone nodes, selecting and sizing the links between pairs of backbone nodes, and configuring logical constructs (virtual links sharing physical links, bandwidth allocations among service classes, etc.).

Modeling

Communications networks are large scale systems with enormous complexity. As with most such systems, modeling relies heavily on computer-based techniques and the nature of the models developed depends strongly on the questions the model is intended to answer. For example, a simulation may be used to answer detailed questions regarding the interaction of communications devices or protocols. Often these studies address questions as to the feasibility of a given device or protocol to support certain types of traffic requirements with acceptable performance. Such models can be used to design the devices or protocols as well. Simulation of communications systems typically models the generation, transfer, and disposition of each unit of information (e.g., call, packet, cell), the protocol decisions as the system operates and the physical behavior of the devices that make up the network. As with any simulation, various

aspects of the system may be ignored or aggregated to improve the computational speed of the simulation.

An alternative to simulation approaches is analytical modeling (a classic reference is Kleinrock 1976), which typically implicitly aggregates traffic units into flows whose characteristics are captured using statistical or probabilistic models. The advantage of analytical modeling is that the behavior of a network can be predicted by a system of equations more quickly computed than the operation of the network can be simulated. The disadvantage is in the aggregation and averaging of detail, effectively capturing the behavior of the network on average rather than accurately depicting a realization of performance over time. Most analytical models of communications systems employ individual and network queueing models. Information units (calls, packets, cells) are the customers in these queueing systems and communications devices (switches, links, etc.) are the servers.

Hybrid simulation/analytical modeling is a third and increasingly popular approach to communications network modeling (Sage and Sykes 1994). The tenets of this approach are to use simulation techniques in capturing key protocol decisions in traffic admission, routing, congestion control, and resource allocation, but to use analytical techniques for modeling the behavior of the traffic itself, thus avoiding the computational complexity incurred if each packet or cell were to be simulated individually. Hybrid simulation/analytical models of communications networks also have been described as "flow-based simulations," in which the paths that traffic flows take are simulated while the flows themselves are modeled analytically.

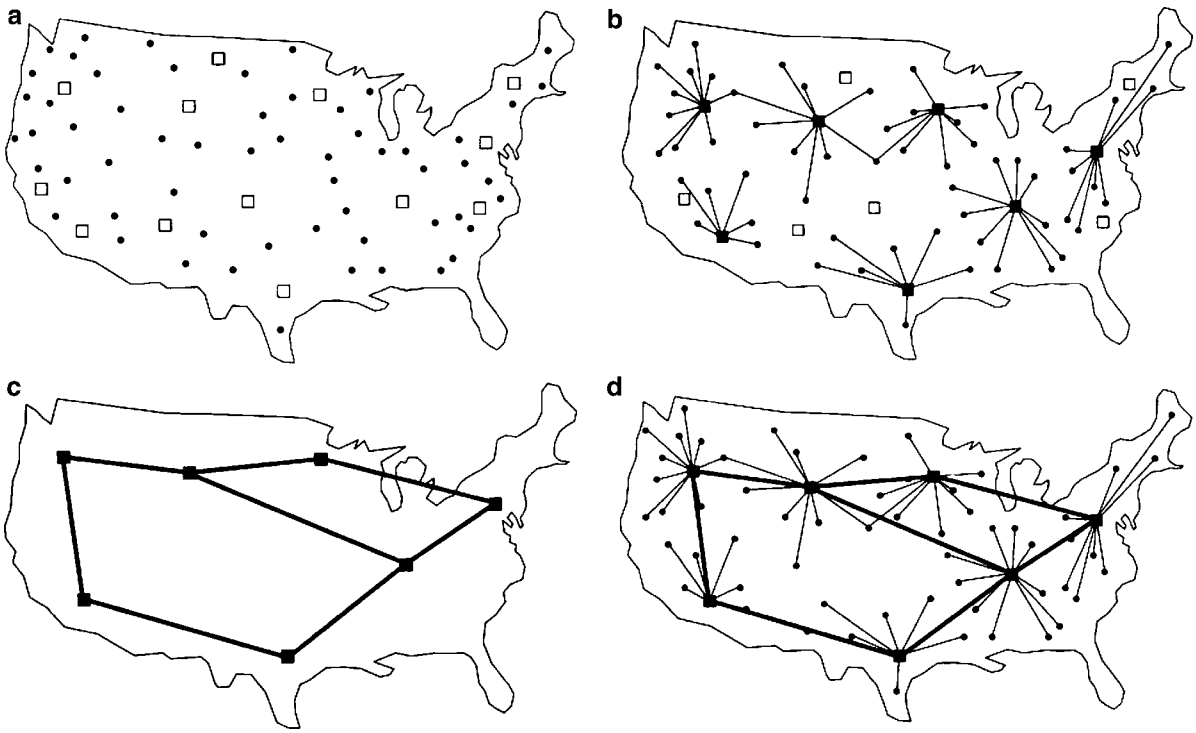
Selection of modeling approach depends strongly on the purpose to which the model is applied. For purposes of protocol or device design, where many replications of realizations of performance are required to observe the entity under a wide variety of operational conditions and circumstances, simulation approaches dominate. For analysis and design purposes, where often the intent is to assess the quality of the design or to compare alternative designs, models which provide average behavior over many potential realizations of performance are useful. Performance can be computed over multiple simulation replications, but analytical tools or simulation analytical hybrids which compute those aver-ages directly and more efficiently are dominant.

Analysis

Network analysis is the application of one or more network models to characterize a communications network. In many communications network design contexts, the central step of the design process is to characterize a design on a number of categories of measures: cost, topological properties, performance, behavior under failures (survivability) being the major ones. For each of these categories of measures, models which compute specific measures of interest can be applied, with the aggregate network analysis being produced in summary from the results of the individual models. Cost measures can include one-time (e.g., device purchase) and recurring costs (e.g., link leasing), often commensurate to the same units. Topological measures are generally technology independent characterizations of the network structure along gross lines (e.g., measures summarizing path availability and diversity, path lengths in number of links or hops from source to destination, etc.). Performance measures are generally technology dependent characterizations of the ability of the network to support the offered traffic and the quality of that support. Survivability measures are indications of what traffic can be supported under various failure scenarios and what the performance of the network will be in those scenarios.

Design

A common paradigm for design of communications networks is one in which the design process is broken down into two phases: *access area design* and *backbone design* (Boorstyn and Frank 1977). Access area design determines the number and location of backbone nodes and homes (i.e., provides a link from) each subscriber to a backbone node. Backbone design determines the interconnections among (links between) backbone nodes. The process is depicted in Fig. 1. Figure 1(a) represents the starting point, where the subscriber locations (black circles) and candidate backbone node locations (squares) are given.



Communications Networks, Fig. 1 (**a**) Network design – starting point. (**b**) Network design – access area design. (**c**) Network design – backbone design. (**d**) Network design – integrated solution

Figure 1(b) represents the completion of the access area design phase, where the black squares are the selected backbone nodes and the lines from the subscribers to the backbone nodes are the homings (implicit in the homings is the assumption that the communications links from each subscriber to its switch is of type and capacity to support the subscriber's offered traffic requirements). The output of the access area design phase is the input to the backbone design phase: the number and location of backbone nodes and the aggregate traffic requirements among the backbone nodes. The aggregate traffic is computed based on the homings. In the backbone phase, the interconnections among backbone nodes are designed to support the backbone traffic with adequate performance, to meet other constraints and typically to minimize cost. Figure 1(c) depicts a backbone design and Fig. 1(d) depicts the final overall solution.

It is notable that solution of the global design problem (including all access and backbone components) is precluded by the computational complexity of the design problem for all but a few special cases which will be ignored here. It also is notable that the structure of the decomposition of the global problem into access area and backbone design phases can lead to gross suboptimalities in the overall solution. To illustrate this assertion, consider a global design problem in which the total cost of the network includes three components:

- homing link costs, the sum of the costs of links homing subscribers to nodes, which can vary for each subscriber-node pair;
- backbone node costs, typically the cost associated with purchasing each node selected as part of the backbone, which typically is uniform over all candidate nodes; and
- backbone link costs, the sum of the costs of links between backbone nodes, which can vary for each subscriber-node pair.

Under fairly general assumptions, the following relationships hold as the number of nodes selected for the backbone increases:

- the access area homing costs tend to decrease (because the access links tend to decrease in length and hence cost);

- the node activation costs increase linearly (directly with the number of nodes selected); and
- the backbone link costs tend to increase (as the number of backbone nodes increases, more backbone links are required).

Thus, if the access area design phase optimizes solely on the basis of homing and node activation costs, it tends to select too many nodes. Two remedies to this pathology are commonly employed: (i) using some estimate of the backbone cost in the access area design problem; or (ii) iterating on the number of backbone nodes selected (i.e., fixing the number of activated nodes to given number, solving the access area and backbone problems in sequence for that number, and computing the total solution cost, but doing so over a wide range of numbers of nodes and selecting the best total cost solution obtained). Neither remedy guarantees global optimality, but both approaches can improve solution costs substantially.

Access area design problems often are formulated as 0-1 integer-programming problems (Fischer et al. 1993) that are strongly related to discrete location problems and/or facility location problems generally (Mirchandani and Francis 1990). In many cases, these integer programs are too large to be solved directly, so a variety of solution approaches are used; for example, linear-programming relaxation methods, Lagrangian relaxation methods, cutting plane and column generation methods, etc. (Ahuja et al. 1993). Alternatively, heuristic algorithms can be used to solve the access problem, and perhaps, more generally, clustering techniques can be used as a solution approach. The basic access area design problem can be stated as follows:

Given:	Subscriber-to-subscriber traffic requirements; Candidate node locations.
Minimize:	Sum of costs of homing each subscriber to a backbone node + Sum of node activation costs.
Over:	Node activations; Subscriber homings.
Subject to:	Node port constraints (limit on the number of subscribers than can be homed to a node); Node traffic constraints (limit on the total amount of subscriber traffic that can be homed to a node); Each subscriber must be homed to a node (occasionally subscribers must be homed to more than one node); (Optionally, a constraint fixing the number of node activations).

Backbone design problems can be formulated as 0-1 or general integer-programming problems (Gavish 1986), however, it is difficult if not impossible to accurately capture or predict network performance in that context. Moreover, many of the critical aspects of the backbone problem that can be captured in the integer-programming formulation (e.g., topological constraints) can also cause a combinatorial explosion in its solution time. Nonetheless, OR/ MS literature is replete with many IP backbone design formulations. In these cases, the solution techniques again typically rely on LP relaxation or Lagrangian relaxation approaches.

An alternative to the mathematical-programming approach to backbone design is commonly employed in interactive software based tools for solving design problems (Stiffler and Sykes 1990; Monma and Shallcross 1989). This iterative approach:

- starts the design process with an initial design;
- analyzes the design using a series of models assessing measures of various aspects of cost, topological properties, performance, and physical constraints on feasibility;
- makes an assessment as to whether the design is satisfactory, stopping if so; and if not
- improves one or more design deficiencies and returns to the analysis step.

This iterative paradigm for backbone design has been used extensively and successfully for design of communications networks with a wide range of architectures (e.g., voice, packet data, multiplexer, asynchronous transfer mode). It also can capture directly a broader set of design objectives and constraints than mathematical-programming methods, as well as be implemented in ways which more accurately predict network performance. All of this is possible through the embedding of the comprehensive network analysis at the core of the process, along with the decomposition of the optimization process into smaller steps aimed at initial design generation and design improvement. Unlike mathematical-programming approaches, which often can be solved to optimality or at least provide bounds from optimality for the solutions they produce, iterative approaches typically cannot guarantee nor bound optimality.

A typical backbone design problem can be stated as follows:

Given:	Backbone node-to-node traffic requirements; Node locations; Link availability and costing.
Minimize:	Sum of link costs.
Over:	Link Placement.

(*continued*)

Subject to:	Topological Constraints, such as – Node connectivity (lower bound on the number of node-disjoint paths available between each node pair); Diameter (upper bound on the minimum number of links a node pair must traverse in order to communicate); and Node port degree (upper bound/physical limit on the number of links that can be incident to each node). Performance Constraints, such as – Constraints on throughput, utilization, delay, blocking, etc., as appropriate for a given network architecture; and Constraints on achieving the QoS requirements of individual traffic demands or, on the achievement of service level agreements computed as a function of QoS of individual traffic demands.

Concluding Remarks

For an overview of telecommunications systems and their operations, see Bertsekas and Gallager (1987); Schwartz (1987); Tanenbaum (2010). For an introduction to network design problems and optimization approaches, see Cahn (1998) or Kershenbaum (1993). For a classical introduction to data communications networking, see Kleinrock (1976), that contains extensive basic modeling and optimization discussions. Also see Pattavina (1998); Ross (1995) or Woodward (1994) for modeling work, and Schmidt and Minoli (1998) or Partridge (1994) for technologies of communications networking. Of a general interest are the books by Ball et al. (1995); León-Garcia and Widjaja (2004), Koster and Muñoz (2010).

See

- ▶ Integer and Combinatorial Optimization
- ▶ Network Optimization
- ▶ Networks of Queues
- ▶ Queueing Theory

References

Ahuja, R. K., Magnanti, T. L., & Orlin, J. B. (1993). *Network flows*. Englewood Cliffs, NJ: Prentice Hall.

Ball, M., Magnanti, T., Monma, C., & Nemhauser, G. (Eds.). (1995). *Network routing*. New York: Elsevier Science.

Bertsekas, D., & Gallager, R. (1987). *Data networks*. Englewood Cliffs, NJ: Prentice Hall.

Boorstyn, R. R., & Frank, H. (1977). Large scale network topological optimization. *IEEE Transactions on Communications, 25*, 29–47. COM.

Cahn, R. S. (1998). *Wide area network design: Concepts and tools for optimization*. San Francisco: Morgan Kaufmann.

Fischer, M. J., Swinsky, G. W., Garland, D. P., & Stanfel, L. E. (1993). A methodology for designing large private line transmission networks with multiple facilities. *Telecommunication Systems, 1*, 243–261.

Gavish, B. (1986). A general model for the topological design of communications networks. *Proceedings GLOBCOM '86*, 1584–1588.

Kershenbaum, A. (1993). *Telecommunications network design algorithms*. New York: McGraw-Hill.

Kleinrock, L. (1975). *Queueing systems, volume I: Theory*. New York: John Wiley.

Kleinrock, L. (1976). *Queueing systems, volume II: Computer applications*. New York: John Wiley.

Koster, A., & Muñoz, X. (Eds.). (2010). *Graphs and algorithms in communication networks*. New York: Springer.

León-Garcia, A., & Widjaja, I. (2004). *Communication networks* (2nd ed.). New York: McGraw-Hill.

Mirchandani, P. B., & Francis, R. L. (Eds.). (1990). *Discrete location theory*. New York: John Wiley.

Monma, C. L., & Shallcross, D. F. (1989). Methods for designing communications networks with certain two-connected survivability constraints. *Operations Research, 37*, 531–541.

Partridge, C. (1994). *Gigabit networking*. Reading, MA: Addison-Wesley.

Pattavina, A. (1998). *Switching theory: Architecture and performance in broadband ATM networks*. Chichester, UK: John Wiley.

Ross, K. W. (1995). *Multiservice loss models for broadband telecommunication networks*. New York: Springer.

Sage, K. M., & Sykes, E. A. (1994). Evaluation of routing-related performance for large scale packet-switched networks with distributed, adaptive routing policies. *Information and Decision Technologies, 19*, 545–562.

Schmidt, A. G., & Minoli, D. (1998). *Multiprotocol over ATM: Building state of the Art ATM intranets*. Greenwich, CT: Manning.

Schwartz, M. (1987). *Telecommunication networks, protocols, modeling and analysis*. Reading, MA: Addison-Wesley.

Stiffler, J. A. & Sykes, E. A. (1990). An AI/OR hybrid expert system for data network design. *Proceedings of the 1990 IEEE International Conference on Systems, Man and Cybernetics*, 307–313.

Tanenbaum, A. S. (2010). *Computer networks* (5th ed.). Englewood Cliffs, NJ: Prentice Hall.

Woodward, M. E. (1994). *Communication and computer networks: Modeling with discrete-time queues*. Los Alamitos, CA: IEEE Computer Society Press.

Community OR

Rebecca Herron
University of Lincoln, Lincoln, UK

Introduction

Community OR is perhaps best understood as a subdiscipline of OR that focuses on communities

as alternative clients for (and users of) OR and related activity.

One defining characteristic of this has been the explicit consideration of two key questions:

1. How can communities benefit from OR approaches, i.e., what can OR offer?
2. Who are the clients/beneficiaries?

These questions have also prompted the subsidiary methodological question: "How should OR practice and theory change in light of this?" These are non-trivial questions which can radically change the view that OR (or management science) as a discipline, only serves traditional managerial interests.

Community OR analysts, or facilitators, as they more frequently refer to themselves, aim to develop engagements within community groups, not-for-profit organizations and/or local multi-agency partnerships so as to provide them analytical support that build capacity and resilience within these communities. Such engagements are frequently related to improving awareness of options and choices available to individuals and groups, and the likely outcomes of various courses of action, e.g. planning and decision-making. They also often involve exploring issues that directly affect people's lives and the strengthening of discourse, dialogue, and local/global agency in relation to these (dialogue, critical awareness, and encouraging self-organization).

The idea of a wider client group for OR and an interdisciplinary approach is not a new idea or pursuit – indeed it has its origins deep in OR's early history – nor is it a concept that has developed identically with different Community OR contributors since. Despite these important differences, common themes have emerged between researchers and practitioners alike that unite the endeavor under the term Community OR.

The practice of Community OR has been organized in many different ways. Much work has been carried out by individuals acting in a voluntary capacity within their own communities, often as a natural extension/application of their personal OR background. Other work has been generated through formalized units and research centers. Both provide contrasting experiences and insights, with different roles and practical and ethical considerations for the Operational Researcher concerned.

Much debate has taken place within the Community OR community about what approaches and methods are appropriate. Given the broad nature of community issues, problems, and choices to be considered, it should not be surprising to find a wide variety of OR/Management Science tools being put to use. While some authors have reported the importance of developing the capacity for quantitative analysis within community groups, a large proportion of the Community OR work has involved softer OR methods that help structure issues: Problem Structuring Methods, understanding inter-related issues and exploring choices; Strategic Choice Approach and decision making/planning, and helping communities reflect on current and future situations and to organize themselves in light of these reflections; and workshops, organizational learning, or creative methods.

Links to systems research are a particularly strong feature of many Community OR studies, but this is not a universal precondition for Community OR. In particular, ideas of socio-technical systems, viable systems modeling (cybernetics), soft systems methodology, boundary critique, and critical systems have played key roles in the development of Community OR.

There is a strong U.K. tradition of Community OR, encouraged by the U.K. OR Society, but global perspectives have been just as important in shaping the development of the subject, including work in Venezuela, Columbia, Ghana, Kenya, New Zealand and Mexico. Many links can also be found in the writings of Community OR researchers and practitioners with Social Justice, Community Development, and Participatory Methods in research and evaluation.

The emphasis of Community OR is often on trying to build meaningful engagements with groups of people in the community and creating increased capacity within the groups. Community OR, therefore, pays great attention to issues of engagement, facilitation, group work, ethical interactions, and sustainability. These are not issues unique to Community OR, but they are in sharp focus in this field.

Another consideration is the question of motivating factors for involvement in Community OR. One factor has already been introduced – a volunteer applying OR skills within his/her own personal communities. Others, however, are motivated to work within new communities, usually ones where there is felt to be some kind of social

inequality or collective issue to address. Community OR researchers tend not to see themselves as the expert in these situations; rather, authors write about bringing another viewpoint/set of skills to a community who are already expert in their own situations. As such, there are some very interesting issues about knowledge and power to unpick in Community OR, and political, religious or ethical beliefs to reflect upon. These have often shaped the way Community OR has been developed by individuals or groups of researchers; seen particularly in the theoretical foundations of the approaches developed and, indeed, in the choices of community issues that are addressed in practice.

History

The desire that OR should be put to use for societal aims is not a new one. As early as the 1930s, Patrick Blackett (often referred to as "the father of OR"), John Bernal and others were calling for science to be put to use for wider social benefit (Blackett 1935; Bernal 1939). In the post-war years in the U.K., OR departments were set up across governmental and nationalized industries.

By the 1970s, some of the initial social objectives of OR seemed to have been somewhat forgotten. There were, however, several key figures who campaigned for redressing this balance and restoring the interdisciplinary, action-focused aspects of the OR enterprise.

In the U.S., Russell Ackoff and colleagues at the University of Philadelphia pioneered work with local residents. In the title of his 1970 paper, "*A black ghetto's research on a university,*" Ackoff made an important distinction that was picked up later by Community OR researchers (Ackoff 1970). It was not the University's research on the ghetto, but the research participants' research on a University. Ackoff continued developing his program and calling for a systems approach to societal problems (Ackoff 1974). Other researchers of the same era famously also called for new perspectives (Churchman 1979) and the recognition and appropriate handling of wicked, i.e. messy & complex, social problems (Rittel and Webber 1973).

In the U.K. there were also calls within the OR community to turn attention again to the social application of OR. The Institute of OR had been established in 1963, later to become attached to the Tavistock Institute of Human Relations, both with an interest in understanding social and public affairs and to work to improve planning processes; Friend and Hickling (2005) discuss some of the long term legacies from this work. Other researchers continued to argue strongly that OR should be used to benefit society and lead to improved well-being (Cook 1973; Thunhurst 1973).

By the 1980s, the movement pressing for OR to move beyond its now well-established scope and client-base to encompass community beneficiaries had gathered considerable momentum. In 1985, under the presidency of Jonathon Rosenhead, the U.K. OR Society wanted to challenge perceived views about who the clients of OR were and to find a significant social role for OR:

> The idea behind the Unit is that is should give extensive experience of how formal problem-structuring approaches can assist non-hierarchical organizations, disposing of few resources, but attempting to represent the interests of their members … we see the unit as extending the range of OR's potential clients … we shall be expanding the domain of rational argument, tackling a new and exciting range of unstructured problems, and contributing to making our society a better one to live in. (Rosenhead 1987, quoted in Parry and Mingers 1991).

The OR Society set up the original Community Operational Research Unit at Barnsley College, Yorkshire (Ritchie et al. 1994, discuss case studies from this period). A new Community OR Network was created the following year by the Society, and a Centre for Community OR was also set up at Hull University, building on the work of Jackson and Keys (Jackson 1987, gives an overview of these 3 initiatives).

Other work in this era also fed into the discussion of the newly emerging concept of Community OR. This included the work of Jones and Eden (1981) and parallel, but related systemic researches such as Ulrich's Critical Heuristics (Ulrich 1983).

By the 1990s, sufficient examples of Community OR were appearing in the literature (Thunhurst 1987; Mar Molinero 1993; Taket and White 1994; Ritchie et al. 1994, and others) to enable the community to reflect on Community OR as an entity and approach in its own right (Jackson 1987;

Mar Molinero 1992; Midgley and Ochoa-Arias 1999; Ochoa-Arias 1994; Parry and Mingers 1991; White and Taket 1993; Wong and Mingers 1994). These papers demonstrate a breadth of OR activity drawn from a range of OR techniques (hard and soft OR methods), drawing on different intellectual traditions (e.g. different forms of systems modeling or political theory), but sharing common aims and interests, such as participation, social justice, and community empowerment.

The following decade saw a period of consolidation for Community OR with the publication of three core OR texts that in different ways document and shape the record of Community OR theory and practice. *Planning under Pressure* (Friend and Hickling 2005) shares the authors' and 21 other contributors' experiences of using the Strategic Choice Approach. *Rational Analysis for a Problematic World Revisited* (Rosenhead and Mingers 2001) brings together the work of several authors associated with the development of Community OR and illustrates their approaches to Problem Structuring in a range of decision contexts. *Community Operational Research, OR and Systems Thinking for Community Development* (Midgley and Ochoa-Arias 2004) focuses specifically on Community OR and brings together several key papers as well as setting the scene for some future development, e.g. in environmental OR (Midgley and Reynolds 2004).

At the end of the 1990's the Community Operational Research Unit (CORU) moved to the University of Lincolnshire and Humberside (soon to become the University of Lincoln). During the subsequent decade researchers from CORU developed citizen learning networks and participatory evaluations - exploring issues for social justice, well-being, community self-organization and social action research (Herron (2006) and Herron & Mendiwelso Bendek (2007) introduce examples in two Special Editions of *OR Insight*, vol. 19 issue 2 and vol. 20 issue 2).

The expansion possibilities of the scope of Community OR, in terms of geographical spread of activity and the issues under consideration, are extensive. There are important global issues that call for the continued and renewed attention: poverty alleviation, social justice, community well-being, environmental responsibility, fair trade, community organization and resilience, as well as local, national, and international calls for greater community participation in local decision making and planning.

Emerging Themes for Community OR

There has been much interesting debate about the variety within Community OR that reflects the range of different professional contexts of researchers and practitioners and different contributing areas of expertise within OR/Systems/Action Research. Accounts of different engagements provide a rich source of case studies of using different OR methods and approaches with different communities (Ritchie et al. 1994; Midgley and Ochoa-Arias 2004).

The nature of the community situations encountered has also clearly had a large impact in what has been done under the banner of Community OR, and creating engagements that are meaningful and have value for those taking part may mean the selection, and modification, customization of different OR methods. It does not seem very practicable to attempt to define Community OR by the choice of methods or tools used, although a familiarity with Soft OR/Problem Structuring Methods is useful for understanding much of the existing literature or to develop skills likely to be of use within a number of community contexts (Rosenhead and Mingers 2001; Friend and Hickling 2005).

Community OR could also be defined as the resulting body of work of a community of practice (the socially constructed definition). But, to define the subject this way provides little insight for the initial enquirer, as it requires a familiarity and further knowledge of the outputs of the community of practice concerned.

Importantly, all this does not mean that it is not possible to identify Community OR themes that have emerged over the past decades. Rather than discussing a single unified methodology for Community OR, it seems productive to describe similarities of approach that transcend discussion of which methods have been developed or applied, and focuses instead on some of the general characteristics and values emergent in Community OR. An introduction would not be complete without at least starting to draw out a few of these themes that recur in many, if not necessarily all, Community OR activities. Alongside the other elements presented above, these will help to provide a fuller introduction to the subject.

Engaging with Communities

Interventions and Interactions with Local Groups

Community OR is generally understood as a form of action research. In this sense, it returns to some of the original intentions and conceptions of OR where interdisciplinary teams work together to support solutions to problematic situations and/or work with problem-owners to explore improvements in how they operate (Jackson 1987). In Community OR this usually always involves some form of engagement with community groups, not-for-profit organizations, or multi-agency partnerships working toward a social/community aim.

To work in meaningful ways with community groups, regardless of the type of method used, requires the establishment of good working relations with the relevant parties and the identification of key stakeholders, particularly those made vulnerable or marginalized by the current situation. Many Community OR approaches start with some form of stakeholder analysis and the scoping of different perspectives and points of view, along with other forms of collective sense-making or mess-structuring activity (Rittel and Webber 1973; Rosenhead and Mingers 2001).

Encouraging the full participation of all the stakeholders identified can be very challenging and requires the building of trust and the careful consideration of issues of access and appropriate delivery. Flexibility and creativity of approach is likely to be valuable – and many Community OR engagements include the need to adapt methods and delivery styles for the particular group concerned. Thus, in addition to content discussions, much Community OR literature also explores these softer issues of group facilitation, interacting with communities and sense-making activities that handle multiple perspectives and conflict of opinion and objective.

Analytical Process Support

A common motivation for undertaking Community OR work is the desire to support a community or communities. Often this support is in terms of providing some structured intervention such as workshop, training, or participant research that helps strengthen the groups' ability to think through and analyze a situation, identify or create new resources, build robust arguments and narratives (sometimes even models), or create improved dialogue and awareness of a situation amongst stakeholders.

Much of the work done in this respect has links to, and has implications for, planning and decision making, awareness of options, and choice of action. Community OR interventions are very often workshop-based or use other forms of community-based learning. Community OR facilitators usually provide process-facilitation and spaces for reflection and dialogue: the aim being to support increased capacity within communities to understand and be more resilient to changes in their external environment.

Emphasizing once again that Community OR is not defined by the application of any particular method, approaches applied in community contexts have included:

- Cognitive Mapping
- Community Visioning
- Critical Systems Heuristics (CSH)
- Drama Theory
- Strategic Choice Approach (SCA)
- Strategic Options Development and Analysis (SODA)
- Soft Systems Methodology (SSM)
- Viable Systems Modeling (VSM)

The exact form of analytical support provided by each Community OR researcher/practitioner will, of course, be context dependent—specific to the exact needs of a particular group and the issue concerned at a certain time—and it will also be bounded by the choice of method or approach chosen. Certain themes, however, emerge in terms of what Community OR may be said to support, including:

- Processes and structures (exploring, building, and rethinking)
- Dialogue and supporting groups to build stronger arguments (logical argumentation)
- Information and enabling groups to make more informed choices (handling information)
- Reasoning about local issues, including exploring links to global issues
- Negotiation and creating rules of engagement
- Engaging relevant others; exploring and extending stakeholders
- Sweeping-in new elements to the model such as individuals, issues, values, ethics

- Critical awareness of learning processes, political impact on decision making, changing environments, and the possible impacts and side effects of choices and actions

Reasons for doing this include:

- Organizational learning – facilitating community-learning processes
- Knowledge transfer – co-creating knowledge and analysis
- Addressing inequalities – providing access to analytical resources
- Increasing choice and resilience for communities - managing uncertainty
- Increasing community knowledge and confidence to act in changing environments

Emancipation and Social Justice

The U.K. OR Society has described OR as "The Science of Better." For Community OR, this highlights the key issues of improvement, i.e., improvement for whom, and how?

Community OR, as much as any subdiscipline of OR, has highlighted the complexity of working for improvement in contexts where there are multiple goals and perspectives. It has provided many examples and ways of proceeding in situations where there are multiple world-views, value systems, and objectives interacting to build up a complex collective logic. In this, Community OR has often been seen as working in the way suggested by Rittel and Webber (1973): continually solving and resolving tricky, messy, and complex situations and providing ongoing support for situations that continue to evolve.

Community OR practice seems to generate several key themes related to making the notion of a Science of Better meaningful for work with communities:

- Ethical dimensions: considering the likely impact on multiple stakeholders
- Locus of control: who has ownership of the goals? processes? outcomes?
- Surfacing issues: creating models participants find authentic and provide insight
- Increasing understanding and fairer dialogue
- Collective improvement: critical reflection from different perspectives
- Consideration of likely side-effects and issues of robustness and sustainability
- Increasing individual and collective control and agency
- Supporting vulnerable people, readdressing inequalities, and rethinking the client
- Exit strategies: building community capacity for learning, analysis, and reflection

Different Community OR facilitators have approached many of these issues in different styles and by using different methodologies, especially SSM, VSM, CSH, SCA, the choice of which depends very much on their experiences, cultural and intellectual traditions, and personal beliefs. However some common emergent themes are worth noting:

- Inclusion of vulnerable groups in decision making and new forms of participation
- Empowerment of communities, emancipation, and addressing social inequality
- Democratic decision making: dialogue, interaction, and community learning
- Handling plurality: multiple realities and objectives
- Self-organization and local control: strengthening civil society
- Feedback, communication, and the interlinking of issues
- Linking local issues to global concerns

Community OR also continues a long tradition in OR of valuing the co-creation of knowledge. The Community OR facilitator will usually have certain process knowledge to contribute (e.g., problem structuring, mess-handling, restructuring, drama theory, game theory), but other participants in the group will have been involved because of their local context knowledge and experience, or other specific knowledge bases. Community OR practice usually explicitly values these other knowledge forms and encourages groups to take ownership of the process of exploration, and idea and solution generation. This is also consistent with the emancipatory interests of many researchers and practitioners, often underpinned by distinct philosophical positions including those shaped by the works of Habermas and Foucault.

Core to any discussion about any Science of Better must also be a critical reflection of who has the power to determine decisions – and the directions chosen for improvement. More critically still, those who do not have any say in these decisions are excluded from the dialogue for any number of reasons, or only able to have a very small voice in the decision-making

processes that affect their lives? Thus, while much of OR strives to remain firmly outside political discourse, Community OR practitioners work in situations where strengthening collective critical awareness of the rules of engagement, and the fairness of these, is seen as an important aspect of the work of the analyst.

See

- ▶ Cognitive Mapping
- ▶ Critical Systems Thinking
- ▶ Deep Uncertainty
- ▶ Delphi Method
- ▶ Developing Countries
- ▶ Group Decision Making
- ▶ Soft Systems Methodology
- ▶ Strategic Assumption Surfacing and Testing (SAST)
- ▶ Strategic Choice Approach (SCA)
- ▶ Strategic Options Development and Analysis (SODA)
- ▶ System Dynamics
- ▶ Systems Analysis

References

Ackoff, R. L. (1970). A black ghetto's research on a university. *Operations Research, 18*, 761–771.

Ackoff, R. L. (1974). *Redesigning the future: A systems approach to societal problems*. New York: Wiley.

Bernal, J. D. (1939). *The social function of science*. London: Routledge.

Blackett, P. M. S. (1935). The frustration of science. In D. Hall et al. (Eds.), *The frustration of science* (pp. 129–144). London: Allen and Unwin.

Churchman, C. W. (1979). *The systems approach and its enemies*. New York: Basic Books.

Cook, S. L. (1973). Operational research, social well-being and the zero-growth concept. *Omega, 1*(6).

Friend, J., & Hickling, A. (2005). *Planning under pressure* (3rd ed.). Oxford: Elsevier/Butterworth-Heinemann.

Herron, R. (2006). Editorial – Special issue for community operational research. *OR Insight, 19*(2), 2–3.

Herron, R., & Bendek, M. (2007). Take part: Active learning for active citizenship contributing to community OR reflections and practices. *OR Insight, 20*(2), 3–7.

Jackson, M. C. (1987). Community operational research: Purposes, theory and practice. *Dragon, 2*(2), 47–73.

Johnson, M. (Ed.). (2011). *Community-based operations research: Decision modeling for local impact and diverse populations*. New York: Springer.

Jones, S., & Eden, C. (1981). OR in the community. *Journal of the Operational Research Society, 32*, 335–345.

Mar Molinero, C. (1992). Operational research: From war to community. *Socio-Economic Planning Sciences, 26*, 203–212.

Mar Molinero, C. (1993). Aldermoor School: The operational researcher on the side of the community. *Journal of the Operational Research Society, 44*, 237–245.

Midgley, G., & Ochoa-Arias, A. E. (1999). Visions of *community OR*. *Omega, 27*, 259–274.

Midgley, G., & Ochoa-Arias, A. E., (Eds.) (2004). *Community operational research, OR and systems thinking for community development*. Kluwer Academic/Plenum.

Midgley, G., & Reynolds, M. (2004). Community and environmental OR: Towards a New Agenda. In G. Midgley & A. E. Ochoa-Arias (Eds.), *Community operational research, OR and systems thinking for community development*. Kluwer Academic/Plenum.

Ochoa-Arias, A. E. (1994). The possibilities for community OR in a third world country. *International Transactions in Operational Research, 1*, 345–352.

Parry, R., & Mingers, J. (1991). Community operational research: Its context and its future. *Omega, 19*, 577–586.

Ritchie, C., Taket, A., & Bryant, J. (Eds.). (1994). *Community works: 26 case studies showing community operational research in action*. Sheffield: Pavic Press.

Rittel, H. J. W., & Webber, M. M. (1973). Dilemmas in a general theory of planning. *Policy Science, 4*, 155–169.

Rosenhead, J. (1987). From management science to workers' science. In M. C. Jackson & P. Keys (Eds.), *New directions in management science*. Aldershot: Gower.

Rosenhead, J. V., & Mingers, J. (Eds.). (2001). *Rational analysis for a problematic world revisited*. Chichester: Wiley.

Taket, A. R., & White, L. A. (1994). Doing community operational research with multicultural groups. *Omega: International Journal of Management Science, 22*(6), 579–588.

Thunhurst, C. (1973). Who does OR operate for?. *Presented at OR Society Conference*, Torquay.

Thunhurst, C. (1987). Doing OR with the community. *Dragon, 2*, 143–153.

Ulrich, W. (1983). *Critical heuristics of social planning*. Berne: Paul Haupt.

White, L., & Taket, A. (1993). The death of the expert. *Journal of the Operational Research Society, 45*, 733–748.

Wong, N., & Mingers, J. (1994). The nature of community OR. *Journal of the Operational Research Society, 45*, 245–254.

Complementarity Applications

Steven A. Gabriel
University of Maryland, College Park, MD, USA

Introduction

This article emphasizes complementarity applications found in the infrastructure industries. A number of

such industries in the U.S. and overseas have been restructured with the goals of making them more competitive and transparent. Examples of these industries in the U.S. include: energy (production, transmission, and distribution), air transportation, and telecommunications. The business model for these industries dramatically changed with a greater emphasis on the microeconomic, game theory effects of individual market players seeking to maximize their own profits or utilities rather than just having a central facility optimizing a stream of regulated profits. Along with these changes came the rise of data-gathering culminating in the present-day Internet which in some cases is nearly real-time. These two sources of change (no doubt with other contributing factors) have led to the rise of complementarity models in the operations research community.

This broad class of mathematical programs includes many optimization problems (via their Karush-Kuhn-Tucker conditions), n-person Nash games, solving nonlinear equations, and many other interesting problems in a variety of engineering and economic settings (Cottle et al. 1992; Facchinei and Pang 2003). A related class of problems, variational inequalities, also benefitted from these contributing factors (see Facchinei and Pang (2003) for a discussion of the relationship between these two problem classes). Collectively, complementarity or variational inequality problems are sometimes called equilibrium problems in that they both seek to arrive at a solution so that the system under study is balanced or has no incentive to change.

One advantage of complementarity problems over traditional optimization is the ability to simultaneously manipulate both primal variables as well as shadow prices for resources, usually expressed as Lagrange multipliers for constraints. This ability, coupled with the complicated picture of restructured industries which often are composed of both regulated and deregulated portions, can usually be approached from the complementarity perspective resulting in richer, more realistic models.

As complementarity models have become more mainstream like linear programs that preceded them, other more complicated and potentially more realistic problem classes have been studied. These extensions of complementarity problems include: mathematical programs with equilibrium constraints (MPECs) which are optimization problems having two or more levels with the bottom ones potentially complementarity problems (Luo et al. 1996), quasi-variational inqualities corresponding to Generalized Nash games (Harker 1991), (Facchinei and Pang 2003), and equilibrium problems with equilibrium constraints (EPECs), problems with two or more levels with an equilibrium at multiple levels, which are some of the hardest problems to solve (Facchinei and Pang 2003).

There are several forms for the complementarity problem, the most common of which is the mixed one abbreviated as MCP (mixed complementarity problem). Note that the term mixed refers to the presence of both equations and inequalities. Having a function $F : R^n \to R^n$, MCP(F) is to find vectors $x \in R^{n_1}, y \in R^{n_2}$ such that the following conditions hold:

$$F_x(x, y) \geq 0, x \geq 0, x^T F_x(x, y) = 0 \tag{1a}$$

$$F_y(x, y) = 0, y \quad \text{free} \tag{1b}$$

where the notation $F_x(x, y), F_y(x, y)$ refers, respectively, to those components of F that match up with the vectors x and y. Equivalently, the first set of conditions (1a) can be stated as $F_i(x, y) \geq 0, x_i \geq 0, x_i \cdot F_i(x, y) = 0$, for $i = 1, \ldots, n$ with the last set referred to as "complementary conditions" (either x_i or $F_i(x, y)$ or both must equal zero). A more compact representation of this first set of conditions is often stated as $0 \leq F_x(x, y) \perp x \geq 0$ with the perpendicular operator $\perp$ denoting the inner product of two vectors equal to zero. The statement of MCP(F) is deceptively simple– just a set of inequalities and complementarity conditions, as well as equations that must simultaneously be satisfied. This very general form, however, includes many problems in optimization, game theory, as well as a host of other areas some of which are described below; additional examples and/or related theory can be found in Cottle et al. (1992), Harker and Pang (1990), Harker (1993), Ferris and Pang (1997), Ferris et al. (2001), Facchinei and Pang (2003), Gabriel et al. (2013).

Discussion

To demonstrate the generality and flexibility of (1), a few representative examples are next shown.

A Simple Complementarity Example

Let $n_1 = 2$, $n_2 = 1$ and $F : R^3 \to R^3$ be defined as:

$$F(x_1, x_2, y_1) = \begin{pmatrix} F_1(x_1, x_2, y_1) \\ F_2(x_1, x_2, y_1) \\ F_3(x_1, x_2, y_1) \end{pmatrix} = \begin{pmatrix} x_1 - x_2 \\ x_1 + y_1 \\ x_1 - y_1 + 1 \end{pmatrix}. \quad (2)$$

The corresponding MCP is to find x_1, x_2, y_1 that simultaneously solve the following conditions:

$$\begin{aligned} F_1(x_1, x_2, y_1) = x_1 - x_2 \geq 0, \\ x_1 \geq 0, (x_1 - x_2) \cdot x_1 = 0; \\ F_2(x_1, x_2, y_1) = x_1 + y_1 \geq 0, \\ x_2 \geq 0, (x_1 + y_1) \cdot x_2 = 0; \\ F_3(x_1, x_2, y_1) = x_1 - y_1 + 1 = 0, \\ y_1 \text{ free}. \end{aligned} \quad (3)$$

The first question with any mathematical programming problem is to try to find the solution set. For small problems like (3), the set of solutions can often be determined by enumeration of several cases and then by some algebra. Doing so additionally provides some insight into the structure of complementarity problems.

The first step is to eliminate the free variable y_1 by using the equation $x_1 - y_1 + 1 = 0 \Leftrightarrow y_1 = x_1 + 1$ and then making the substitution in the remaining two sets of conditions to obtain an equivalent set of conditions:

$$\begin{array}{lll} x_1 - x_2 \geq 0, & x_1 \geq 0, & (x_1 - x_2) \cdot x_1 = 0, \\ 2x_1 + 1 \geq 0, & x_2 \geq 0, & (2x_1 + 1) \cdot x_2 = 0. \end{array} \quad (4)$$

Next, the following four cases can be analyzed to determine the solution set:

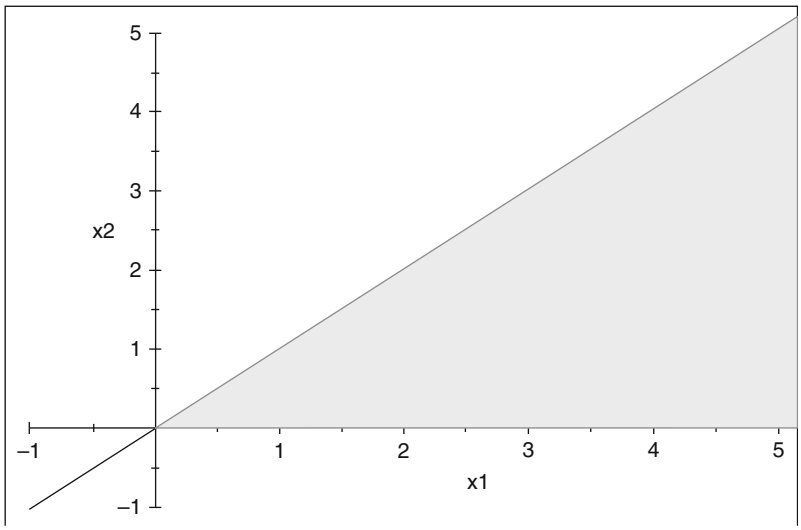
1. $x_1 > 0, x_2 > 0$
2. $x_1 = 0, x_2 > 0$
3. $x_1 > 0, x_2 = 0$
4. $x_1 = 0, x_2 = 0$

Using the complementary conditions, the first case implies that

$$(x_1 - x_2) = 0, (2x_1 + 1) = 0,$$

or that $x_1 = x_2 = -\frac{1}{2}$, which is not possible since both these variables must be positive. Case 2 also is not possible since by complementarity it would imply that $x_1 = -\frac{1}{2} \not\geq 0$. Analyzing Case 3 shows that by complementarity, $x_1 - x_2 = 0$ or that both values must be the same. This is not possible under this case as $x_1 > 0, x_2 = 0$. Lastly, if both values are equal to zero, then all the inequalities as well as complementarity conditions hold. Thus, $(x_1, x_2, y_1) = (0, 0, 1)$ is the unique solution to this linear MCP. This simple, three-variable linear MCP can also be viewed from a geometric point of view in $x_1 - x_2$ space as shown in Fig. 1. The conditions: $x_1 - x_2 \geq 0, x_1 \geq 0, x_2 \geq 0$ represent a polyhedron a sample of which is shown by the shaded region. The condition $2x_1 + 1 \geq 0 \Leftrightarrow x_1 \geq -\frac{1}{2}$ is superfluous given that x_1 must be nonnegative. The first complementarity condition $(x_1 - x_2) \cdot x_1 = 0$ can be interpreted as: if $x_1 > 0$ then the potential solution must be on the line $x_1 = x_2$ which in turn would make $x_2 > 0$. The second complementarity condition would then force $x_1 = -\frac{1}{2}$ which is not in the shaded region. Thus, the only other choice it to have $x_1 = 0$ (satisfying the first complementarity condition) but forcing $x_2 = 0$ by the second one in (4). Hence, the only point in the shaded region that satisfies all the six conditions of (4) is the origin.

While this simple example had a unique solution, in general this will not be the case for MCPs. Indeed, there can be no solutions, one solution, any finite number of solutions or an infinite number since MCPs generalize solving nonlinear (or linear) equations, optimization problems (one or more), or combinations thereof. To see that (1) includes solving equations, consider the case when there are no inequalities, i.e., just $F_y(x, y) = 0, y$ free. In the next sections, the connection with optimization problems and extensions is explored.



Complementarity Applications, Fig. 1 Geometric Depiction of Simple Example

Connection between Optimization and Complementarity Problems

Consider the following standard (primal) linear program and its corresponding dual problem:

$$\min_{x} \boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{x} \tag{5a}$$

$$s.t.\ \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \tag{5b}$$

$$\boldsymbol{x} \geq 0 \tag{5c}$$

$$\max_{y} \boldsymbol{b}^{\boldsymbol{T}}\boldsymbol{y} \tag{6a}$$

$$s.t.\ \boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{y} \leq \boldsymbol{c} \tag{6b}$$

$$\boldsymbol{y} \geq 0 \tag{6c}$$

where $\boldsymbol{A}$ is a real-valued $m \times n$ matrix, $\boldsymbol{c} \in R^n, \boldsymbol{b} \in R^m$. The m primal constraints $\boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}$ are associated with the dual vector $\boldsymbol{y} \in R^m$ and likewise the n dual constraints match up with the primal variables $\boldsymbol{x} \in R^n$. The Complementarity Slackness Theorem (Luenberger 1984) states that if:

1. x is feasible to the primal problem (5)
2. y is feasible to the dual problem (6)

then a necessary and sufficient condition for (x, y) to be optimal solutions to their respective problems is that complementary slackness is satisfied, namely: $(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})_j \cdot y_j = 0, j = 1, \ldots, m$ and $\left(\boldsymbol{c} - \boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{y}\right)_i \cdot x_i = 0, i = 1, \ldots, n.$

However, the feasibility and complementary conditions amount to:

1. $\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \geq 0, \boldsymbol{x} \geq 0$
2. $\boldsymbol{c} - \boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{y} \geq 0, \boldsymbol{y} \geq 0$
3. $(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})_j \cdot y_j = 0, j = 1, \ldots, m$ and $\left(\boldsymbol{c} - \boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{y}\right)_i \cdot x_i = 0, \quad i = 1, \ldots, n.$

After realizing that complementarity slackness can be re-expressed as $(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})^T \boldsymbol{y} = 0$ and $\left(\boldsymbol{c} - \boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{y}\right)^T \boldsymbol{x} = 0$ given the nonnegativity of the quantities involved, these three sets of optimality conditions can be expressed succinctly as the linear complementarity problem with only inequalities (i.e., "pure complementarity problem") with $F : R^{n+m} \rightarrow R^{n+m}$ given as follows:

$$F(\boldsymbol{x}, \boldsymbol{y}) = \begin{pmatrix} \boldsymbol{c} - \boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{y} \\ \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \end{pmatrix} = \begin{pmatrix} 0 & -\boldsymbol{A}^{\boldsymbol{T}} \\ \boldsymbol{A} & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} + \begin{pmatrix} \boldsymbol{c} \\ -\boldsymbol{b} \end{pmatrix}$$

Moreover, if the original primal LP had equalities, via a similar line of reasoning, the result would be a mixed (as opposed to pure) complementarity problem. This shows that every linear program is an instance of an MCP. A key distinction to be made here between optimization and complementarity problems is that the latter's formulation involves both primal and dual variables whereas the former's formulation is only in terms of primal (or just dual) variables. As will be shown in some of the examples below, the complementarity approach can lead to richer models that manipulate the dual variables (i.e., prices) while

also considering the primal (usually physical) variables in many infrastructure applications. First, the next section shows that any nonlinear program via its Karush-Kuhn-Tucker (KKT) conditions is also an instance of an MCP generalizing the results for linear programs.

Consider a standard nonlinear program as follows:

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) \tag{7a}$$

$$s.t.\ g_i(\boldsymbol{x}) \leq 0, i = 1, \ldots, m \tag{7b}$$

$$h_j(\boldsymbol{x}) = 0, j = 1, \ldots, p \tag{7c}$$

where $f, g_i(\boldsymbol{x}), h_j(\boldsymbol{x}) : R^n \rightarrow R$ are respectively, the objective function and the constraint functions. The KKT conditions (Bazaraa et al. 1993) are to find $\boldsymbol{x} \in R^n$, Lagrange multipliers $\boldsymbol{\lambda} \in R^m$ (for the inequality constraints) and $\boldsymbol{\gamma} \in R^p$ (for the equality constraints) such that the following conditions hold:

$$\nabla f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i \cdot \nabla g_i(\boldsymbol{x}) + \sum_{j=1}^{p} \gamma_j \cdot \nabla h_j(\boldsymbol{x}) = 0, \boldsymbol{x} \text{ free} \tag{8a}$$

$$-g_i(\boldsymbol{x}) \geq 0, \lambda_i \geq 0, g_i(\boldsymbol{x}) \cdot \lambda_i = 0, i = 1, \ldots, m \tag{8b}$$

$$h_j(\boldsymbol{x}) = 0, \gamma_j \text{ free}, j = 1, \ldots, p. \tag{8c}$$

Clearly the KKT conditions are just a set of equations with corresponding free variables and inequalities with associated nonnegative variables and complementarity conditions. As such, the KKT conditions are also an instance of an MCP (Gabriel 2008; Gabriel et al. (2013)) with $F : R^{n+m+p} \rightarrow R^{n+m+p}$ given as:

$$F\begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \\ \boldsymbol{\gamma} \end{pmatrix} = \begin{pmatrix} \nabla f(x) + \sum_{i=1}^{m} \lambda_i \cdot \nabla g_i(\boldsymbol{x}) + \sum_{j=1}^{p} \gamma_j \cdot \nabla h_j(\boldsymbol{x}) \\ -g_i(\boldsymbol{x}), i = 1, \ldots, m \\ h_j(\boldsymbol{x}), j = 1, \ldots, p \end{pmatrix}$$

with the first and third sets of constraints being equations and the second set inequalities (≥ 0). Optimization problems that do not have valid KKT conditions are not directly special cases of MCPs.

Since KKT conditions for integer programs (IPs) are not generally valid, there is not a direct connection between MCPs and IPs. However, there is an indirect association between these two classes of problems. In particular, consider the following mixed, linear complementarity problem as an example. This problem in general form, is to find vectors $\boldsymbol{x}, \boldsymbol{y}$ such that:

$$0 \leq \boldsymbol{q}_1 + (\boldsymbol{M}_{11}\boldsymbol{x} + \boldsymbol{M}_{12}\boldsymbol{y}) \perp \boldsymbol{x} \geq 0 \tag{9a}$$

$$0 = \boldsymbol{q}_2 + (\boldsymbol{M}_{21}\boldsymbol{x} + \boldsymbol{M}_{22}\boldsymbol{y}), \boldsymbol{y} \text{ free} \tag{9b}$$

where the matrices $\boldsymbol{M}_{11}, \boldsymbol{M}_{12}, \boldsymbol{M}_{21}, \boldsymbol{M}_{22}$ are of order $r_1 \times n_1, r_1 \times n_2, r_2 \times n_1, r_2 \times n_2$, respectively, coinciding with $\boldsymbol{x} \in R^{n_1}, \boldsymbol{y} \in R^{n_2}$. Also, the constant vectors $\boldsymbol{q}_1, \boldsymbol{q}_2$ are of size r_1 and r_2, respectively. This system can be re-expressed as the following set of polyhedral constraints with additional binary variables $b \in \{0, 1\}^{r_1}$:

$$0 \leq \boldsymbol{q}_1 + (\boldsymbol{M}_{11}\boldsymbol{x} + \boldsymbol{M}_{12}\boldsymbol{y}) \leq K\boldsymbol{b} \tag{10a}$$

$$0 \leq \boldsymbol{x} \leq K(1 - \boldsymbol{b}) \tag{10b}$$

$$0 = \boldsymbol{q}_2 + (\boldsymbol{M}_{21}\boldsymbol{x} + \boldsymbol{M}_{22}\boldsymbol{y}), \boldsymbol{y} \text{ free} \tag{10c}$$

with K a suitably chosen positive constant (could vary for each of the r_1 constraints). To see why this works it suffices to realize that complementarity conditions are either-or type restrictions. Either one term equals zero or the other does (or both). This disjunction is equivalently expressed in (10) via the binary variables $\boldsymbol{b}$ and the constant K (Fortuny-Amat and McCarl 1981). In principle then, one could replace a linear complementarity problem's conditions by a set of linear conditions with binary variables as shown above. The problem arises in determining an appropriate constant K. Too small a value will unnecessarily restrict the problem and result in (3) being infeasible. Too large a value may result in numerical ill-conditioning that could make the problem hard to solve. See Gabriel and Leuthold (2010) for an example of disjunctive constraints and some guidance on computing the constant K relative to energy modeling in the context of a two-level optimization problem of which the bottom level is an MCP.

Nash-Cournot Production Game

The next example concerns a classical Nash-Cournot production game (Shy 1995) with two producers. Such a model is applicable in a variety of areas such as energy, manufacturing, as well as many others. Also, the model can easily be extended to more than two producers with additional producer-level constraints included or marketing-clearing conditions as depicted in the network example shown below. The particular instance of this duopoly is from Gabriel (2008); Gabriel et al. (2013).

Unlike a perfectly competitive production environment, in the current setting each producer can affect market prices by adjusting its own production level. This market power feature is encoded in the objective function of each of the players $i = 1, 2$. More specifically, each player must decide on their own production level $q_i, i = 1, 2$ given that they have knowledge of the (inverse) market demand function $p(q_1 + q_2) = \alpha - \beta(q_1 + q_2)$ where $\alpha, \beta > 0$. This function gives the price of the produced good but takes into account both producers' production levels. If only one player decided to increase production, the total price for the market would go down (since $\beta > 0$) but that producer's profit might go up due to a more favorable market share. If both producers are considering production levels under these circumstances, it is not immediately clear what might be an equilibrium solution (i.e., one in which neither player has an incentive to deviate). The Nash concept is to have each player optimally solve for their production level which maximizes net profit (or other suitable function), given that the other player's level is fixed at its own optimal level.

Assuming for ease of presentation that both players have a linear production cost function given by $c_i(q_i) = \delta_i q_i$ for $\delta_i > 0, i = 1, 2$, then the resulting profit-maximization problem that player i solves is:

$$\max_{q_i} p(q_1 + q_2) \cdot q_i - c_i(q_i) \tag{11a}$$

$$s.t.\ q_i \geq 0 \tag{11b}$$

or

$$\max_{q_i} (\alpha - \beta(q_1 + q_2)) \cdot q_i - \delta_i q_i \tag{12a}$$

$$s.t.\ q_i \geq 0 \tag{12b}$$

The KKT conditions are both necessary and sufficient for this problem. Necessity follows by the linearity of the constraints and sufficiency is because the objective function is a (strictly) concave function of q_i (in addition to the linear constraints). To see the concavity result, note that the second derivative of the objective function relative to q_i is just $-2\beta < 0$ (Bazaraa et al. 1993). The resulting KKT conditions for each player taken together form a Nash-Cournot equilibrium and are given as:

$$0 \leq 2\beta q_1 + \beta q_2 - \alpha + \delta_1, q_1 \geq 0,$$

$$(2\beta q_1 + \beta q_2 - \alpha + \delta_1) \cdot q_1 = 0 \text{ (producer 1)} \tag{13a}$$

$$0 \leq \beta q_1 + 2\beta q_2 - \alpha + \delta_2, q_2 \geq 0,$$

$$(\beta q_1 + 2\beta q_2 - \alpha + \delta_2) \cdot q_2 = 0 \text{ (producer 2)} \tag{13b}$$

These conditions taken together constitute the following pure linear complementarity problem with function F:

$$F\begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} 2\beta q_1 + \beta q_2 - \alpha + \delta_1 \\ \beta q_1 + 2\beta q_2 - \alpha + \delta_2 \end{pmatrix} = \begin{pmatrix} 2\beta & \beta \\ \beta & 2\beta \end{pmatrix}\begin{pmatrix} q_1 \\ q_2 \end{pmatrix} + \begin{pmatrix} -\alpha + \delta_1 \\ -\alpha + \delta_2 \end{pmatrix}.$$

If one can assume that both quantities $q_i > 0$ in an equilibrium solution, then the above conditions become a set of two unknowns (production) in two equations, namely:

$$0 = 2\beta q_1 + \beta q_2 - \alpha + \delta_1, \text{ (producer 1)} \tag{14a}$$

$$0 = \beta q_1 + 2\beta q_2 - \alpha + \delta_2, \text{ (producer 2)} \tag{14b}$$

Solving for the positive production levels in these two equations amounts to using a best reaction or best response function (Osborne and Rubinstein 1994), essentially closed-form expressions for q_i as a function of the other production quantity. Due to limitations on assuming positive production for all producers, or for example, the need for considering more challenging constraints (apart from nonnegativity), it is much more efficient to use the complementarity approach. Indeed, for more realistic models, it is not always possible to use this best response approach. Some MCP examples with

realism that have been presented (in energy markets) include Hobbs (2001), Gabriel et al. (2005a), Gabriel et al. (2005b), Hobbs et al. (2008).

PIES Energy Equilibrium

While the above models have considered either one or more optimization problems or nonlinear equations as instances of MCPs, the PIES (Project Independence Evaluation System) energy planning model is an important example that combines both these two approaches into a complementarity problem (Hogan 1975; Josephy 1979; Ahn 1979; Ahn and Hogan 1982). A stylized version of a much later and more complicated generation of PIES is the National Energy Modeling Systems (NEMS). NEMS has also been shown to be an instance of an MCP (Gabriel et al. 2001) and is currently used by the U.S. Department of Energy for a variety of energy market studies and reports.

The PIES model considers the supply and demand sides of the energy market separately with x denoting the vector of decision variables in energy production. The supply side is modeled as a production cost minimization given as follows with $\boldsymbol{c}$ a vector of costs conformal with $\boldsymbol{x}$:

$$\min_{\boldsymbol{x}} \boldsymbol{c}^T\boldsymbol{x} \tag{15a}$$

$$s.t.\ \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{q} \qquad (\pi) \tag{15b}$$

$$\boldsymbol{B}\boldsymbol{x} \geq \boldsymbol{b} \qquad (\gamma) \tag{15c}$$

$$\boldsymbol{x} \geq 0 \tag{15d}$$

In this linear-programming problem, besides the nonnegativity restrictions, there are two sets of constraints: meeting demand q by a combination of energy production types Ax (15b) and other, non-demand related conditions to be met (15c). The dual prices are π and γ in (15b) and (15c), respectively. As opposed to a straightforward linear program, these two prices will enter directly into another part of the MCP formulation for this problem. In particular, the demand side of the energy market is given by econometric equations of the following form where p is price:

$$q_i(p) = q_i^0 \prod \left(\frac{p_i}{p_i^0}\right)^{e_{ij}} \tag{16}$$

Here q_i^0, p_i^0 are, respectively, reference demands and prices for energy product i, and e_{ij} is an elasticity between energy products i and j. The supply and demand sides of the energy market are combined by the equilibration condition:

$$\pi = p \tag{17}$$

In a nutshell, this condition states that the price used in the demand equation (16) should reflect the value of the resources involved, i.e., be the dual price to the demand equation from (15b).

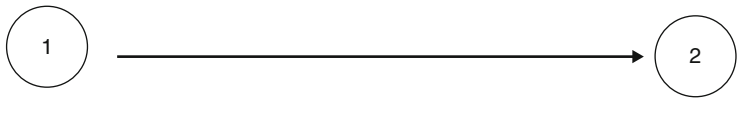
One way to join these two parts of the energy market is to substitute $\pi = p$ into (16) and then restrict π to be the dual vector to (15b). This can be done by considering the (necessary and sufficient) optimality conditions to (15) taking q as a fixed quantity. Then, the formula for q from (16) is used. The resulting MCP function for PIES (Cottle et al. 1992) is thus the following:

$$F\begin{pmatrix} x \\ \pi \\ \gamma \end{pmatrix} = \begin{pmatrix} c - A^T\pi - B^T\gamma \\ Ax - q(\pi) \\ Bx - b \end{pmatrix}$$

To try to directly incorporate both sides of the market as described above with just an optimization problem is not possible. In particular, to compute the vector of dual prices π, the LP (15) must first be solved. But to solve it, the optimal demand quantity $q = q(\pi)$ is required which in turn depends on the optimal π via (16) and (17). The MCP formulation gets around this computational issue by simultaneously determining both primal and dual variables. This feature is a strength of complementarity problems and thus easily allows combining both equations and optimization problems in one formulation.

Market Equilibrium with Underlying Network

The Nash-Cournot model described above as well as it's n-player counterpart assumes that each player has some ability to manipulate market prices by adjusting their own production levels. In fact, it is only in the objective functions of the players' problems that their separate decision variables interact. Two interesting variations on this paradigm are: Generalized Nash



Complementarity Applications, Fig. 2 Sample Two-Node Network

equilibrium problems (Facchinei and Pang 2003) and network equilibria. Generalized Nash problems allow for other players' variables to enter into the constraint set of a player. Using the two-player example shown above, in a Generalized Nash version, there might be a common constraint for each player of the form:

$$q_1 + q_2 + Inv \geq Dem \tag{18}$$

where *Inv* is the amount of inventory (for a region) and *Dem* its demand level. This common constraint would then say that the total supply (production + inventory) must at least meet the demand level. These sorts of problems belong to a class of mathematical programs that are generally harder to solve than MCPs but under certain conditions on these common constraints are expressable as complementarity problems or the related variational inequality problem. See Harker (1991) for a discussion of this result, as well as Facchinei and Pang (2003) for a theoretical treatment of variational inequalities and extensions that relate to Generalized Nash problems.

Another variation on the previous Nash equilibrium is to have multiple players each optimizing their objectives but without the ability to directly influence prices, i.e., they are price-takers. Rather, there is a market-clearing equation (or equations) whose dual variables are the associated market prices. Since the decision variables for each player contribute to the market-clearing conditions, these players have some indirect influence on the prices. As compared to the Generalized Nash model, these conditions do not appear in the constraint set of the players though. Often such problems have an underlying infrastructure network related perhaps to distribution of energy, water, or other products to transport.

Consider the following sample network equilibrium problem from Gabriel et al. (2013). There are two nodes in the network as depicted in Fig. 2. These nodes can represent cities, countries, regions, or just a market for a particular product. Production can occur at either node but only node 2 can receive additional product from the other node as indicated by the uni-directional arc. The product in question could be energy (e.g., electricity), fuels (natural gas, oil, coal), treated water, manufactured goods (e.g., televisions) or raw materials to name a few choices. There are a number of key questions that such an equilibrium model should answer.

For example, in meeting the demand at node 2, how much should be locally produced at node 2 and how much should be imported from the other node? What will be the equilibrium prices at each node if all players are acting in their own interests to maximize profits?

The production aspects can be modeled by the following optimization problem (shown here for producer A) in which net profit is to be maximized subject to production, balance, and nonnegativity constraints:

$$\max_{s_1^A, q_1^A, f_{12}^A} \pi_1 s_1^A + \pi_2 f_{12}^A - c_1^A\left(q_1^A\right) - \left(\tau_{12}^{\text{Reg}} + \tau_{12}\right) f_{12}^A \tag{19a}$$

$$s.t.\ q_1^A \leq \overline{q}_1^A \ \left(\lambda_1^A\right) \tag{19b}$$

$$s_1^A = q_1^A - f_{12}^A \ \left(\delta_1^A\right) \tag{19c}$$

$$s_1^A, q_1^A, f_{12}^A \geq 0 \tag{19d}$$

where

- $p \in \{A, B, C, D\}$ is the index for the producers
- $i \in \{1, 2\}$ is the index for the nodes
- q_n^p is the production quantity for producer p at node n
- $\overline{q}_n^p$ is the maximum production capacity for producer p at node n
- s_n^p is the amount sold by producer p at node n
- f_{12}^A, f_{12}^B are respectively, the amount of exports from node 1 to 2 by producers A and B (the other two producers do not have that option)
- π_n is the price at node n determined by market-clearing conditions
- $\tau_{12}^{Reg}, \tau_{12}$ are respectively, the exogenous, regulated export tariff when sending product from node 1 to 2 and the endogenously-determined congestion tariff between the two nodes (but exogenous from the perspective of the producer's optimization problem)

- $c_n^p\left(q_n^p\right)$ is the (marginal) production cost function for producer p at node n. For simplicity, this function is assumed linear and of the form $c_n^p\left(q_n^p\right) = \gamma_n^p q_n^p$ with $\gamma_n^p > 0$
- λ_n^p, δ_n^p are Lagrange multipliers (e.g., dual variables) for the associated constraints

Producer B's problem is similar to the one for A but the other two producers do not have any export-related terms. Since each producer is solving a linear program, the KKT conditions are both necessary and sufficient (Bazaraa et al. 1993). These KKT conditions for each of the four producers are as follows.

Producer A, node 1

$$0 \le -\pi_1 + \delta_1^A \perp s_1^A \ge 0 \tag{20a}$$

$$0 \le \gamma_1^A + \lambda_1^A - \delta_1^A \perp q_1^A \ge 0 \tag{20b}$$

$$0 \le -\pi_2 + \left(\tau_{12}^{Reg} + \tau_{12}\right) + \delta_1^A \perp f_{12}^A \ge 0 \tag{20c}$$

$$0 \le \overline{q}_1^A - q_1^A \perp \lambda_1^A \ge 0 \tag{20d}$$

$$0 = s_1^A - q_1^A + f_{12}^A, \delta_1^A \text{ free} \tag{20e}$$

Producer B, node 1

$$0 \le -\pi_1 + \delta_1^B \perp s_1^B \ge 0 \tag{21a}$$

$$0 \le \gamma_1^B + \lambda_1^B - \delta_1^B \perp q_1^B \ge 0 \tag{21b}$$

$$0 \le -\pi_2 + \left(\tau_{12}^{Reg} + \tau_{12}\right) + \delta_1^B \perp f_{12}^B \ge 0 \tag{21c}$$

$$0 \le \overline{q}_1^B - q_1^B \perp \lambda_1^B \ge 0 \tag{21d}$$

$$0 = s_1^B - q_1^B + f_{12}^B, \delta_1^B \text{ free} \tag{21e}$$

Producer C, node 2

$$0 \le -\pi_2 + \delta_2^C \perp s_2^C \ge 0 \tag{22a}$$

$$0 \le \gamma_2^C + \lambda_2^C - \delta_2^C \perp q_2^C \ge 0 \tag{22b}$$

$$0 \le \overline{q}_2^C - q_2^C \perp \lambda_2^C \ge 0 \tag{22c}$$

$$0 = s_2^C - q_2^C, \delta_2^C \text{ free} \tag{22d}$$

Producer D, node 2

$$0 \le -\pi_2 + \delta_2^D \perp s_2^D \ge 0 \tag{23a}$$

$$0 \le \gamma_2^D + \lambda_2^D - \delta_2^D \perp q_2^D \ge 0 \tag{23b}$$

$$0 \le \overline{q}_2^D - q_2^D \perp \lambda_2^D \ge 0 \tag{23c}$$

$$0 = s_2^D - q_2^D, \delta_2^D \text{ free} \tag{23d}$$

The market-clearing conditions forcing supply to equal demand are:

$$0 = \left[s_1^A + s_1^B\right] - D_1(\pi_1),\ \pi_1 \text{ free} \tag{24a}$$

$$0 = \left[s_2^C + s_2^D + f_{12}^A + f_{12}^B\right] - D_2(\pi_2),\ \pi_2 \text{ free} \tag{24b}$$

where $\left[s_1^A + s_1^B\right]$, $\left[s_2^C + s_2^D + f_{12}^A + f_{12}^B\right]$ are the supply amounts for nodes 1 and 2, respectively and $D_n(\pi_n), n = 1, 2$ are the demands at each node taking into account the nodal price π_n.

Besides production and market-clearing, in some applications (e.g., energy, water) there is a player that makes sure the network is running smoothly. This network system operator (NSO) also solves an optimization problem which can take on a variety of forms maximizing for example, social welfare or net profit to name two. Using net profit, a stylized network operator problem is as follows:

$$\max_{g_{12}} \left(\tau_{12}^{\text{Reg}} + \tau_{12}\right) g_{12} - c^{NSO}(g_{12}) \tag{25a}$$

$$s.t.\ \ g_{12} \le \overline{g}_{12}\ (\varepsilon_{12}) \tag{25b}$$

$$g_{12} \ge 0 \tag{25c}$$

where g_{12} represents the flow from node 1 to node 2 that the NSO manages, $c^{NSO}(g_{12})$ is a network operations cost function (assume linear for simplicity, i.e., $c^{NSO}(g_{12}) = \gamma^{NSO} g_{12}$ where $\gamma^{NSO} > 0$) and ε_{12} is the dual variable associated with the capacity constraint involving the flow upper bound $\overline{g}_{12}$. Like the producers' problems, this is a linear program so that the KKT conditions are both necessary and sufficient and are the following:

$$0 \le -\tau_{12}^{\text{Reg}} - \tau_{12} + \gamma^{NSO} + \varepsilon_{12} \perp g_{12} \ge 0 \tag{26a}$$

$$0 \le \overline{g}_{12} - g_{12} \perp \varepsilon_{12} \ge 0 \tag{26b}$$

To determine the congestion tariff τ_{12}, the following market-clearing conditions can be used:

$$0 = g_{12} - \left[f_{12}^A + f_{12}^B\right],\ \tau_{12} \text{ free} \tag{27}$$

The overall market equilibrium on this network can be expressed as an MCP by collecting the KKT conditions of the producers: (20)–(23) the supply–demand market-clearing conditions (24), the KKT conditions of the NSO (26) and the market-clearing conditions of the network flows (27).

As discussed in Gabriel et al. (2013), suppose the following input data are used.

$$\begin{aligned}
\tau_{12}^{\text{Reg}} &= 0.5 \\
\gamma_1^A = 10, \gamma_1^B &= 12, \gamma_2^C = 15, \gamma_2^D = 18 \\
a_1 = 20, b_1 &= 1, a_2 = 40, b_2 = 2 \\
\overline{q}_1^A = 10, \overline{q}_1^B &= 10, \overline{q}_2^C = 5, \overline{q}_2^D = 5 \\
\overline{g}_{12} &= 5 \\
\gamma^{NSO} &= 1 \\
D_1(\pi_i) &= a_i - b_i \pi_i
\end{aligned}$$

Then, an MCP solution in terms of production quantities, flows, prices, and tariffs is as follows:

$$\begin{aligned}
&q_1^A = 10, q_1^B = 3, q_2^C = 5, q_2^D = 0 \\
&f_{12}^A = 2.561, f_{12}^B = 2.439 \\
&\quad (\text{the sum is } 5 = \text{the capacity of the link}) \\
&\pi_1 = 12, \pi_2 = 15 \\
&\tau_{12} = 2.5 \\
&\quad (\tau_{12}^{\text{Reg}} + \tau_{12} = 3, \text{ the difference in the nodal prices})
\end{aligned}$$

Traffic Equilibrium

One of the classical problems in complementarity modeling is that of predicting steady state flows of cars (or other vehicles) along a congested road. Consider as a simple example, an origin (node 1) and two destinations (nodes 4 and 5) as well as intermediate nodes 2 and 3 as shown in Fig. 3 Wardrop 1952; Aashtiani and Magnanti 1981; Magnanti 1984; Florian 1986, 1989). These nodes can relate to intersection points, cities, regions, etc. The idea is to try to predict how many drivers will be using the individual paths in the network if for example, the price (e.g., time, disutility) of a particular path is taken into account in the decision-making process. That is, if the flow is price-based.

In this simple example, there are two origin–destination (OD) pairs: (1, 4) and (1, 5) which represent where drivers begin and end their trip. In going from node 1 to node 4, drivers can choose either to travel along path 1-2-4 or 1-3-4; for the OD pair (1, 5) there is only one path: 1-2-5. Wardrop (1952) stated an equilibrium where no driver had an incentive to deviate from a particular chosen path resulting in:

- Paths with positive flow serving the same OD pair all having equal costs (otherwise drivers would deviate to the less costly ones)
- Paths with costs higher than the minimum having no flow

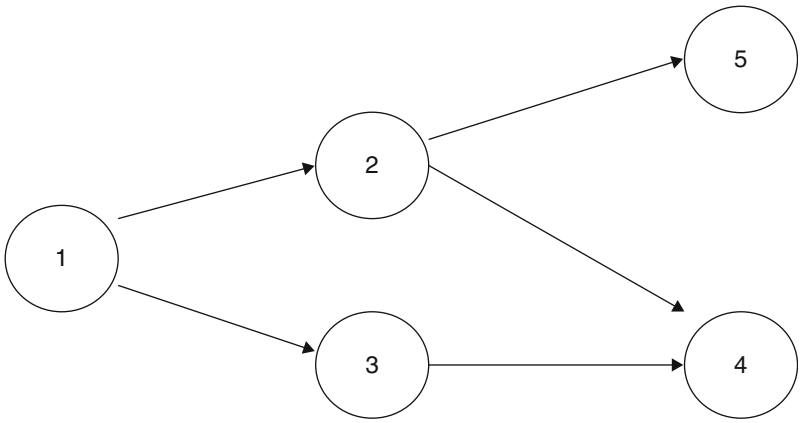
Essentially, such an equilibrium needs to take into account that all drivers are doing what is in their own best interests but that there should be no incentive for any one driver to deviate from a path they pick on their own. As compared to the previous examples of MCPs, in this case there is no explicit optimization problem(s), just an indirect acknowledgement that drivers want to minimize the time or cost of the path chosen.

To present the associated complementarity problem, it is necessary to define some related terms. First, path flows on a path p will be denoted by h_p, e.g., the flow on path 1-2-4 is h_{1-2-4}. The vector of all path flows is given by h which for this example is lexicographically given as:

$$h = \begin{pmatrix} h_{1-2-4} \\ h_{1-2-5} \\ h_{1-3-4} \end{pmatrix}.$$

Flows on an arc a are given by f_a with the vector of all such flows denoted as f. For the sample network shown above:

$$f = \begin{pmatrix} f_{1,2} \\ f_{1,3} \\ f_{2,4} \\ f_{2,5} \\ f_{3,4} \end{pmatrix}.$$



Complementarity Applications, Fig. 3 Sample Network for Traffic Equilibrium Problem

Both h and f are related by the equation:

$$f = \Delta h \tag{28}$$

where $\Delta = \left[\delta_{ap}\right]$ is the arc-path incidence matrix with $\delta_{ap} = 1$ if arc a is on path p and is equal to zero otherwise. Thus, for the network shown above:

$$\Delta = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Parallel to the flow vectors h and f are the cost vectors for paths and arcs, given respectively as $C(h)$ and $c(f)$ which for the sample network are as follows:

$$C(h) = \begin{pmatrix} C_{1-2-4}(h) \\ C_{1-2-5}(h) \\ C_{1-3-4}(h) \end{pmatrix}, c(f) = \begin{pmatrix} c_{1,2}(f) \\ c_{1,3}(f) \\ c_{2,4}(f) \\ c_{2,5}(f) \\ c_{3,4}(f) \end{pmatrix}.$$

The relationship between these two vectors is:

$$C(h) = \Delta^T c(f) \tag{29}$$

indicating that the paths costs are the sum of the arc costs for those arcs on the path. This is the standard additive model which is well-studied but not always realistic when one considers for example tolls. In that case, nonadditive approaches such as those given in Gabriel and Bernstein (1997) and Bernstein and Gabriel (1997) may be more appropriate. An important point to note is that the path costs for a particular path are a function of all the path flows in the network. Likewise, the arc costs for a given arc are a function of all the arc flows. This is a very realistic representation of the network indicating the interaction effects. A more restrictive version is to assume that path p (arc a) only affects the costs for that path (arc) in essence a separability argument. This was an initial assumption used early on in part because it led to solving an equivalent optimization problem (Magnanti 1984; Florian 1989) which was easier to solve before the large growth in complementarity problem algorithms in the 1990s.

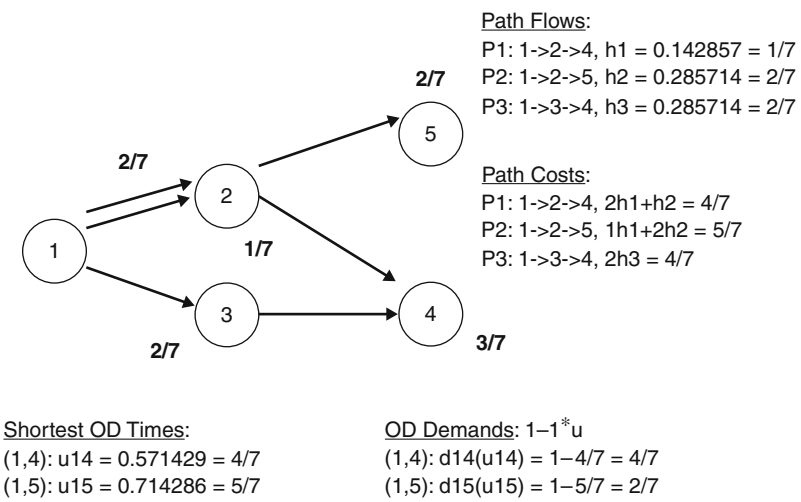
Besides the flow, the complementarity problem associated with a traffic equilibrium also needs to account for meeting the OD demand. For each OD pair i, such demand is denoted by D_i which itself is a function of the shortest time u_i (or least disutility) between the origin and destination i. In the network from Fig. 3, the vector versions of these quantities are thus:

$$D(u) = \begin{pmatrix} D_{(1,4)}(u) \\ D_{(1,5)}(u) \end{pmatrix}, \text{ with } u = \begin{pmatrix} u_{(1,4)} \\ u_{(1,5)} \end{pmatrix}.$$

There is one last notational element to define: the path-OD pair incidence matrix $\Gamma = \left[\gamma_{pi}\right]$ where $\gamma_{pi} = 1$ if path p serves OD pair i and is equal to zero otherwise. For the example above,

$$\Gamma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}$$

for paths 1-2-4, 1-2-5, 1-3-4, and OD pairs, (1,4) and (1,5), respectively.



Complementarity Applications, Fig. 4 Solution to Sample Traffic Equilibrium Problem

The formal statement of the (additive) traffic equilibrium problem is thus to find path flows h and shortest times u such that:

$$0 \leq C_p(h) - u_i \text{ for all } p \in P_i, i \in I, \; h \geq 0 \tag{30a}$$

$$0 = \left(C_p(h) - u_i\right) \cdot h_p \text{ for all } p \in P_i, i \in I \tag{30b}$$

$$0 = \sum_{p \in P_i} h_p - D_i(u) \text{ for all } i \in I, u \geq 0 \tag{30c}$$

where P_i is the set of paths that serve OD pair i and I is the set of OD pairs. Equation (30a) simply states that the path cost $C_p(h)$ must be by definition greater than or equal to the shortest time u_i for all paths serving that OD pair i; also only nonnegative path flows are allowed. Equation (30b) is a translation of the Wardrop statement that appeared above. Namely, if the path p has any positive flow then the path cost must be equal to the shortest time and this must be true for all paths serving that OD pair. Also, if the path cost is strictly greater than the shortest time, there should be no flow along that path. Lastly, (30c) indicates that the total path flow across all paths that serve OD pair i must equal the demand; also, only nonnegative shortest times u_i are allowed. As stated, (30) is not an MCP since the equations in (3) do not match up with free variables u. As shown in Aashtiani and Magnanti (1981), (30c) can be relaxed to $0 \leq \sum_{p \in P_i} h_p - D_i(u)$ for all $i \in I, u \geq 0$ as long as some mild conditions on the path cost and demand functions hold and the corresponding MCP will have a solution that matches up with (30). Another important result is that if the demand function is invertible (or just fixed demand), an arc formulation instead of the more cumbersome path version can be used (Magnanti 1984; Florian 1989). In that case, taking into account (28) and (29), the resulting MCP is as follows:

$$0 \leq \left(\Delta^T c(\Delta h) - \Gamma u\right) \perp h \geq 0 \tag{31a}$$

$$0 \leq \Gamma^T h - D(u) \perp u \geq 0 \tag{31b}$$

assuming the mild restrictions on C and D are also in effect.

To make this formulation (31) concrete, consider the following specific choice for costs and demand functions for the sample network shown above:

$$c_a(f) = f_a \tag{32a}$$

$$D_i(u) = 1 - 1u_i \tag{32b}$$

The resulting solution is shown in Fig. 4. Note that both paths 1-2-4 and 1-3-4 serve OD pair (1,4) and since they both have positive flow ($h_{1-2-4} = \frac{1}{7}$, $h_{1-3-4} = \frac{2}{7}$), by Wardrop's principle they should both

have the same costs and equal to the lowest cost (shortest time) $u_{(1,4)}$ as is shown below. But from (28):

$$f = \begin{pmatrix} f_{1,2} \\ f_{1,3} \\ f_{2,4} \\ f_{2,5} \\ f_{3,4} \end{pmatrix} = \Delta h = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1/7 \\ 2/7 \\ 2/7 \end{pmatrix} = \begin{pmatrix} 3/7 \\ 2/7 \\ 1/7 \\ 2/7 \\ 2/7 \end{pmatrix}$$

$$C_{1-2-4}(h) = \frac{3}{7} + \frac{1}{7} = \frac{4}{7}$$
$$C_{1-3-4}(h) = \frac{2}{7} + \frac{2}{7} = \frac{4}{7}$$
$$u_{(1,4)} = \frac{4}{7}$$
$$C_{1-2-5} = \frac{3}{7} + \frac{2}{7} = \frac{5}{7}$$
$$u_{(1,4)} = \frac{5}{7}$$

Concluding Remarks

In this article, complementary problems have been defined and their relevance to certain infrastructure models has been emphasized. Complementarity problems generalize optimization, game theory, and a host of other interesting problems in engineering and economics. As such, this flexible class of mathematical programs has great relevance to many important operations research problems.

See

- ▶ Bilevel Linear Programming
- ▶ Complementarity Problems
- ▶ Complementary Slackness Theorem
- ▶ Constrained Optimization Problem
- ▶ Constraint Qualification
- ▶ Convex Optimization
- ▶ Dual Linear-Programming Problem
- ▶ Duality Theorem
- ▶ Game Theory
- ▶ Integer and Combinatorial Optimization
- ▶ Karush-Kuhn-Tucker (KKT) Conditions
- ▶ Lagrange Multipliers
- ▶ Linear Programming
- ▶ Network Optimization
- ▶ Nonlinear Programming

References

Aashtiani, H. Z., & Magnanti, T. L. (1981). Equilibria on a congested transportation network. *SIAM Journal on Algebraic Discrete Methods, 2*, 213–226.

Ahn, B. H. (1979). *Computation of market equilibria for policy analysis*. New York: Garland Publishing, Inc.

Ahn, B. H., & Hogan, W. W. (1982). On convergence of the PIES algorithm for computing equilibria. *Operations Research, 30*, 281–300.

Bazaraa, M. S., Sherali, H. D., & Shetty, C. M. (1993). *Nonlinear programming theory and algorithms*. New York: John Wiley & Sons, Inc.

Bernstein, D., & Gabriel, S. A. (1997). Solving the nonadditive traffic equilibrium problem. In P. M. Pardalos, D. W. Hearn, & W. W. Hager (Eds.), *Lecture notes in economics and mathematical systems network optimization* (pp. 72–102). Berlin/New York: Springer.

Cottle, R. W., Pang, J.-S., & Stone, R. E. (1992). *The linear complementarity problem*. San Diego: Academic Press.

Facchinei, F., & Pang, J. S. (2003). *Finite-dimensional variational inequalities and complementarity problems volumes I and II*. New York: Springer.

Ferris, M. C., & Pang, J. S. (Eds.). (1997). *Complementarity and variational problems state of the art*. Baltimore: SIAM.

Ferris, M. C., Mangasarian, O. L., & Pang, J. S. (Eds.). (2001). *Complementarity: Applications, algorithms and extensions*. Dordrecht: Kluwer Academic Publishers.

Florian, M. (1986). Nonlinear cost network models in transportation analysis. *Mathematical Programming Studies, 26*, 167–196.

Florian, M. (1989). Mathematical programming applications in national, regional and urban planning. In M. Iri & K. Tanabe (Eds.), *Mathematical programming: Recent developments and applications* (pp. 57–81). Tokyo: Kluwer Academic Publishers.

Fortuny-Amat, J., & McCarl, B. (1981). A representation and economic interpretation of a two-level programming problem. *Journal of the Operational Research Society, 32*(9), 783–792.

Gabriel, S. A. (2008). Optimization and equilibrium models in energy. College Park, MD: Department of Civil and Environmental Engineering, University of Maryland. Manuscript, December 12, 2008.

Gabriel, S. A., & Bernstein, D. (1997). The traffic equilibrium problem with nonadditive path costs. *Transportation Science, 31*(4), 337–348.

Gabriel, S. A., & Leuthold, F. U. (2010). Solving discretely-constrained MPEC problems with applications in electric power markets. *Energy Economics, 32*, 3–14.

Gabriel, S. A., Kydes, A. S., & Whitman, P. (2001). The national energy modeling system: A large-scale energy-economic equilibrium model. *Operations Research, 49*(1), 14–25.

Gabriel, S. A., Kiet, S., & Zhuang, J. (2005a). A mixed complementarity-based equilibrium model of natural gas markets. *Operations Research, 53*(5), 799–818.
Gabriel, S. A., Zhuang, J., & Kiet, S. (2005b). A large-scale complementarity model of the north American natural gas market. *Energy Economics, 27*, 639–665.
Gabriel, S. A., Conejo, A. J., Fuller, J. D., Hobbs, B. F., & Ruiz, C. (2013). *Complementarity modeling in energy markets*. New York: Springer. Chapter 1.
Harker, P. T. (1991). Generalized Nash games and quasi-variational inequalities. *European Journal of Operational Research, 54*(1), 81–94.
Harker, P. T. (1993). *Lectures on computation of equilibria with equation-based methods* (CORE lecture series). Louvain-La-Neuve: CORE Foundation.
Harker, P. T., & Pang, J.-S. (1990). Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory. *Algorithms and Applications, Mathematical Programming, 48*, 161–220.
Hobbs, B. F. (2001). Linear complementarity models of nash-cournot competition in bilateral and POOLCO power markets. *IEEE Transactions on Power Systems, 16*(2), 194–202.
Hobbs, B. F., Drayton, G., Fisher, E. B., & Lise, W. (2008). Improved transmission representations in oligopolistic market models: Quadratic losses, phase shifters, and DC lines. *IEEE Transactions on Power Systems, 23*(3), 1018–1029.
Hogan, W. W. (1975). Energy policy models for project independence. *Computers & Operations Research, 2*, 251–271.
Josephy, N. H. (1979). *Hogan's PIES example and Lemke's algorithm*. University of Wisconsin-Madison: Mathematics Research Center.
Luenberger, D. G. (1984). *Linear and nonlinear programming* (2nd ed.). Reading, MA: Addison-Wesley.
Luo, Z. Q., Pang, J.-S., & Ralph, D. (1996). *Mathematical programs with equilibrium constraints*. Cambridge, UK: Cambridge University Press.
Magnanti, T. L. (1984). Models and algorithms for predicting urban traffic equilibria. In M. Florian (Ed.), *Transportation planning models* (pp. 153–186). (North-Holland), Amsterdam: Elsevier Science Publishers B.V.
Osborne, M. J., & Rubinstein, A. (1994). *A course in game theory*. Cambridge, MA: The MIT Press.
Shy, O. (1995). *Industrial organization theory and applications*. Cambridge, MA: The MIT Press.
Wardrop, J. G. (1952). Some theoretical aspects of road traffic research. *Proceedings of the ICE Part II, 1*, 325–378.

Complementarity Condition

A relation between two nonnegative vectors in which, whenever a given component of one of the vectors is positive, the corresponding component of the other vector must be zero. For example, two nonnegative n-dimensional vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ satisfy a complementarity condition if their ith components are such that $x_i\, y_i = 0$, $i = 1, \ldots, n$.

See

- ▶ Complementarity Applications
- ▶ Complementarity Problems
- ▶ Complementary Slackness Theorem

Complementarity Problems

Richard W. Cottle
Stanford University, Stanford, CA, USA

Introduction

In its most elementary form, a complementarity problem CP($\boldsymbol{f}$) is an inequality system stated in terms of a mapping $\boldsymbol{f} : R^n \to R^n$. Given $\boldsymbol{f}$, one seeks a vector $\boldsymbol{x} \in R^n$ such that

$$\boldsymbol{x}_i \geq 0, \quad f_i(\boldsymbol{x}) \geq 0, \quad \text{and} \quad x_i f_i(\boldsymbol{x}) = 0 \quad i = 1, \ldots, n. \tag{1}$$

When the mapping $\boldsymbol{f}$ is affine, say of the form $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{q} + \boldsymbol{M}\boldsymbol{x}$, problem (1) is called a linear complementarity problem, denoted LCP$(\boldsymbol{q}, \boldsymbol{M})$ or sometimes just $(\boldsymbol{q}, \boldsymbol{M})$. Otherwise, it is called a nonlinear complementarity problem and is denoted NCP($\boldsymbol{f}$).

If $\bar{\boldsymbol{x}}$ is a solution to (1) satisfying the additional nondegeneracy condition $\bar{x}_i + f_i(\bar{\boldsymbol{x}}) > 0, i = 1, \ldots, n$, the indices i for which $\bar{x}_i > 0$ or $f_i(\bar{\boldsymbol{x}}) > 0$ form complementary subsets of $\{1, \ldots, n\}$. This is believed to be the origin of the term complementary slackness as used in linear and nonlinear programming. It was this terminology that inspired the name complementarity problem.

Sources of Complementarity Problems

The complementarity problem is intimately linked to the Karush-Kuhn-Tucker necessary conditions of local optimality found in mathematical programming

theory. This connection was brought out in Cottle (1964, 1966) and again in Cottle and Dantzig (1968). Finding solutions to such systems was one of the original motivations for studying the subject. Another was the finding of equilibrium points in bimatrix and polymatrix games. This kind of application was emphasized by Howson (1963) and Lemke and Howson (1964). These early contributions also included essentially the first algorithms for these types of problems. There are numerous applications of the linear and nonlinear complementarity problems in computer science, economics, various engineering disciplines, finance, game theory, statistics, and mathematics. Descriptions of—and references to—these applications can be found in the books by Murty (1988), Cottle et al. (1992, 2009), Isac (1992), Isac et al. (2002), and Facchinei and Pang (2003). The survey article by Ferris and Pang (1997) is the richest compendium yet published on engineering and economic applications of complementarity problems.

Equivalent Formulations

The problem CP($\boldsymbol{f}$) can be formulated in several equivalent ways. An obvious one calls for a solution $(\boldsymbol{x}, \boldsymbol{y})$ to the system

$$\boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}) = 0, \quad \boldsymbol{x} \geq 0, \quad \boldsymbol{y} \geq 0, \quad \boldsymbol{x}^{\mathrm{T}}\boldsymbol{y} = 0. \tag{2}$$

Another is to find a zero $\boldsymbol{x}$ of the mapping

$$\boldsymbol{g}(\boldsymbol{x}) = \min\{\boldsymbol{x}, \boldsymbol{f}(\boldsymbol{x})\} \tag{3}$$

where the symbol $\min\{\boldsymbol{a}, \boldsymbol{b}\}$ denotes the componentwise minimum of the two n-vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. Yet another equivalent formulation asks for a fixed point of the mapping

$$\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{x} - \boldsymbol{g}(\boldsymbol{x}),$$

i.e., a vector $\boldsymbol{x} \in R^n$ such that $\boldsymbol{x} = \boldsymbol{h}(\boldsymbol{x})$.

The formulation of CP($\boldsymbol{f}$) given in (3) is related to the (often nonconvex) optimization problem:

$$\begin{aligned} &\text{minimize } \boldsymbol{x}^{\mathrm{T}} f(\boldsymbol{x}) \\ &\text{subject to } \boldsymbol{f}(\boldsymbol{x}) \geq 0 \\ &\qquad \boldsymbol{x} \geq 0 \end{aligned} \tag{4}$$

In such a problem, the objective is bounded below by zero, thus any feasible solution of (4) for which the objective function $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{x}) = 0$ must be a global minimum as well as a solution of CP($\boldsymbol{f}$). As it happens, there are circumstances (for instance, the monotonicity of the mapping $\boldsymbol{f}$) under which all the local minima for the mathematical programming problem (4) must in fact be solutions of (3).

Also noteworthy is a result of Eaves and Lemke (1981) showing that the LCP is equivalent to solving a system of equations $\boldsymbol{y} = \varphi(\boldsymbol{x})$ where the mapping $\varphi : R^n \to R^n$ is piecewise linear. In particular, LCP($\boldsymbol{q}$, $\boldsymbol{M}$) is equivalent to finding a vector $\boldsymbol{u}$ such that

$$\boldsymbol{q} + \boldsymbol{M}\boldsymbol{u}^{+} - \boldsymbol{u}^{-} = \boldsymbol{0}$$

where (for $i = 1, \ldots, n$) $u_i^{+} = \max\{0, u_i\}$ and $u_i^{-} = -\min\{0, u_i\}$.

The Linear Complementarity Problem

The LCP has quite an extensive literature, far more so than the NCP. This is most likely attributable to the LCP's relatively greater accessibility. Within this field of study, there are several main directions: the existence and uniqueness (or number of) solutions, mathematical properties of the problem, generalizations of the problem, algorithms, applications, and implementations.

Much of the theory of the linear complementarity problem is strongly linked in various ways to matrix classes. For instance, one of the earliest theorems on the existence of solutions is due to Samelson et al. (1958). Motivated by a problem in structural mechanics, they showed that the LCP($\boldsymbol{q}$, $\boldsymbol{M}$) has a unique solution for every $\boldsymbol{q} \in R^n$ if and only if the matrix $\boldsymbol{M}$ has positive principal minors. (That is, the determinant of every principal submatrix of $\boldsymbol{M}$ is positive.) The class of such matrices has come to be known as **P**, and its members are called **P**-matrices. (It is significant that the Samelson-Thrall-Wesler theorem characterizes a class of matrices in terms of the LCP.) The class **P** includes all positive definite (**PD**) matrices, i.e., those square matrices $\boldsymbol{M}$ for which $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{x} > 0$ for all $\boldsymbol{x} \neq 0$. In the context of the LCP, the term "positive definite" does not

require symmetry. An analogous definition (and usage) holds for positive semi-definite (**PSD**) matrices, namely, $\boldsymbol{M}$ is **PSD** if $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{x} > 0$ for all $\boldsymbol{x}$. Some authors refer to such matrices as monotone because of their connection with monotone mappings. **PSD**-matrices have the property that the associated LCPs ($\boldsymbol{q}$, $\boldsymbol{M}$) are solvable whenever they are feasible, whereas LCPs ($\boldsymbol{q}$, $\boldsymbol{M}$) in which $\mathbf{M} \in \mathbf{PD}$ are always feasible and (since $\mathbf{PD} \subset \mathbf{PSD}$) are always solvable. Murty (1968, 1972) gave this distinction a more general matrix form. He defined $\mathbf{Q}$ as the class of all square matrices for which LCP($\boldsymbol{q}$, $\boldsymbol{M}$) has a solution for all $\boldsymbol{q}$ and $\mathbf{Q}_0$ as the class of all square matrices for which LCP($\boldsymbol{q}$, $\boldsymbol{M}$) has a solution whenever it is feasible. Although the goal of usefully characterizing the classes $\mathbf{Q}$ and $\mathbf{Q}_0$ has not yet been realized, much is known about some of their special subclasses. Indeed, there are now literally dozens of matrix classes for which LCP existence theorems have been established. See Murty (1988), Cottle et al. (1992, 2009), Cottle (2010) and Isac (1992) for an abundance of information on this subject.

Algorithms for Solving LCPs

The algorithms for solving linear complementarity problems are of two major types: pivoting (or, direct) and iterative (or, indirect). Algorithms of the former type are finite procedures that attempt to transform the problem ($\boldsymbol{q}$, $\boldsymbol{M}$) to an equivalent system of the form $(\boldsymbol{q}', \boldsymbol{M}')$ in which $\boldsymbol{q}' \geq 0$. Doing this is not always possible; it depends on the problem data, usually on the matrix class (such as **P**, **PSD**, etc.) to which $\boldsymbol{M}$ belongs. When this approach works, it amounts to carrying out a principal pivotal transformation on the system of equations

$$\boldsymbol{y} = \boldsymbol{q} + \boldsymbol{M}\boldsymbol{x}.$$

To such a transformation there corresponds an index set α (with complementary index set $\bar{\alpha} = \{1, \ldots, n\} \backslash \alpha$) such that the principal submatrix $\boldsymbol{M}_{\alpha\alpha}$ is nonsingular. When this (block pivot) operation is carried out, the system

$$\begin{aligned} \boldsymbol{y}_\alpha &= \boldsymbol{q}_\alpha + \boldsymbol{M}_{\alpha\alpha}\boldsymbol{x}_\alpha + \boldsymbol{M}_{\alpha\bar{\alpha}}\boldsymbol{x}_{\bar{\alpha}} \\ \boldsymbol{y}_{\bar{\alpha}} &= \boldsymbol{q}_{\bar{\alpha}} + \boldsymbol{M}_{\bar{\alpha}\alpha}\boldsymbol{x}_\alpha + \boldsymbol{M}_{\bar{\alpha}\bar{\alpha}}\boldsymbol{x}_{\bar{\alpha}} \end{aligned}$$

becomes

$$\begin{aligned} \boldsymbol{x}_\alpha &= \boldsymbol{q}'_\alpha + \boldsymbol{M}'_{\alpha\alpha}\boldsymbol{y}_\alpha + \boldsymbol{M}'_{\alpha\bar{\alpha}}\boldsymbol{x}_{\bar{\alpha}} \\ \boldsymbol{y}_{\bar{\alpha}} &= \boldsymbol{q}'_{\bar{\alpha}} + \boldsymbol{M}'_{\bar{\alpha}\alpha}\boldsymbol{y}_\alpha + \boldsymbol{M}'_{\bar{\alpha}\bar{\alpha}}\boldsymbol{x}_{\bar{\alpha}} \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{q}'_\alpha &= -\boldsymbol{M}_{\alpha\alpha}^{-1}\boldsymbol{q}_\alpha \\ \boldsymbol{M}'_{\alpha\bar{\alpha}} &= -\boldsymbol{M}_{\alpha\alpha}^{-1} \\ \boldsymbol{M}'_{\alpha\bar{\alpha}} &= -\boldsymbol{M}_{\alpha\alpha}^{-1}\boldsymbol{M}_{\alpha\bar{\alpha}} \\ \boldsymbol{q}'_{\bar{\alpha}} &= \boldsymbol{q}_{\bar{\alpha}} - \boldsymbol{M}_{\bar{\alpha}\alpha}\boldsymbol{M}_{\alpha\alpha}^{-1}\boldsymbol{q}_\alpha \\ \boldsymbol{M}'_{\bar{\alpha}\alpha} &= \boldsymbol{M}_{\bar{\alpha}\alpha}\boldsymbol{M}_{\alpha\alpha}^{-1} \\ \boldsymbol{M}'_{\bar{\alpha}\bar{\alpha}} &= \boldsymbol{M}_{\bar{\alpha}\bar{\alpha}} - \boldsymbol{M}_{\bar{\alpha}\alpha}\boldsymbol{M}_{\alpha\alpha}^{-1}\boldsymbol{M}_{\alpha\bar{\alpha}}. \end{aligned}$$

There are two main pivoting algorithms used in processing LCPs. The more robust of the two is due to Lemke (1965). Lemke's method embeds the LCP($\boldsymbol{q}$, $\boldsymbol{M}$) in a problem having an extra "artificial" nonbasic (independent) variable x_0 with coefficients specially chosen so that when x_0 is sufficiently large, all the basic variables become nonnegative. At the least positive value of x_0 for which this is so, there will (in the nondegenerate case) be (exactly) one basic variable whose value is zero. That variable is exchanged with x_0. Thereafter the method executes a sequence of (almost complementary) simple pivots. In each case, the variable becoming basic is the complement of the variable that became nonbasic in the previous exchange. The method terminates if either x_0 decreases to zero—in which case the problem is solved—or else there is no basic variable whose value decreases as the incoming nonbasic variable is increased. The latter outcome is called termination on a secondary ray. For certain matrix classes, termination on a secondary ray is an indication that the given LCP has no solution. Eaves (1971) was among the first to study Lemke's method from this point of view.

The other pivoting algorithm for the LCP is called the Principal Pivoting Method (see Cottle and Dantzig (1968)). The algorithm has two versions: symmetric and asymmetric. The former executes a sequence of principal (block) pivots or order 1 or 2, whereas the latter does sequences of almost complementary pivots, each of which results in a block principal pivot or order potentially larger than 2. The class of problems to

which the Principal Pivoting Method applies is more restrictive. (See Cottle et al. (1992, 2009) for a treatment of this algorithm.)

Iterative methods are often favored for the solution of very large linear complementarity problems. In such problems, the matrix $\boldsymbol{M}$ tends to be sparse (i.e., to have a small percentage of nonzero elements) and frequently structured. Since iterative methods do not modify the problem data, these features of large-scale problems can be used to advantage. Ordinarily, however, an iterative method does not terminate finitely; instead, it generates a convergent sequence of trial solutions. As is to be expected, the applicability of algorithms in this family depends on the matrix class to which $\boldsymbol{M}$ belongs. Details on several algorithms of this type are presented in the books by Kojima et al. (1991) as well as the one by Cottle et al. (1992, 2009).

Some Generalizations

The linear and nonlinear complementarity problems have been generalized in numerous ways. One of the earliest generalizations was given by Habetler and Price (1971) and Karamardian (1971) who defined the problem $\mathrm{CP}(K, \boldsymbol{f})$ as that of finding a vector $\boldsymbol{x}$ in the closed convex cone K such that $\boldsymbol{f}(x) \in K^*$ (the dual cone) and $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{x}) = 0$. Through this formulation, a connection can be made between complementarity problems and variational inequality problems, that is, problems $\mathrm{VI}(X, \boldsymbol{f})$ wherein one seeks a vector $\boldsymbol{x}^* \in X$ (a nonempty subset of R^n) such that

$$\boldsymbol{f}(\boldsymbol{x}^*)^{\mathrm{T}}(\boldsymbol{y} - \boldsymbol{x}^*) \geq 0 \quad \text{for all } \boldsymbol{y} \in X.$$

Karamardian (1971) established that when X is a closed convex cone, say K, with dual cone K^*, then $\mathrm{CP}(K, \boldsymbol{f})$ and $\mathrm{VI}(X, \boldsymbol{f})$ have exactly the same solutions (if any).

Robinson (1979) has considered the generalized complementarity problem $\mathrm{CP}(K, \boldsymbol{f})$ defined above as an instance of a generalized equation, namely to find a vector $\boldsymbol{x} \in R^n$ such that

$$0 \in \boldsymbol{f}(\boldsymbol{x}) + \partial\psi_K(\boldsymbol{x})$$

where ψ_K is the indicator function of the closed convex cone K and ∂ denotes the subdifferential operator as used in convex analysis.

Among the diverse generalizations of the linear complementarity problem, the earliest appears in Samelson et al. (1958). There—for given $n \times n$ matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ and n-vector $\boldsymbol{c}$—the authors considered the problem of the finding n-vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ such that

$$\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} = \boldsymbol{c}, \quad \boldsymbol{x} \geq 0, \quad \boldsymbol{y} \geq 0 \quad \text{and} \quad \boldsymbol{x}^{\mathrm{T}}\boldsymbol{y} = 0.$$

A different generalization was introduced by Cottle and Dantzig (1970). In this sort of problem, one has an affine mapping $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{q} + \boldsymbol{N}\boldsymbol{x}$ where $\boldsymbol{N}$ is of order $\sum_{j=1}^{k} p_j \times n$ partitioned into k blocks; the vectors $\boldsymbol{q}$ and $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$ are partitioned conformably. Thus,

$$\boldsymbol{y}^j = \boldsymbol{q}^j + \boldsymbol{N}^j\,\boldsymbol{x} \text{ for } j = 1, \ldots, k.$$

The problem is to find a solution of the system

$$\boldsymbol{y} = \boldsymbol{q} + \boldsymbol{N}\boldsymbol{x}, \quad \boldsymbol{x} \geq 0, \quad \boldsymbol{y} \geq 0, \quad \text{and} \quad x_j \prod_{i=1}^{p_j} y_i^j = 0$$

$$(j = 1, \ldots, k).$$

Several authors have further investigated this vertical generalization while others have studied some analogous horizontal generalizations. For representative papers on the vertical LCP, see Ebiefung (1995) and Mohan and Neogy (1997). For the horizontal generalization, Tütüncü and Todd (1995) and Zhang (1994). A further generalization called extended linear complementarity problem (ELCP) was introduced by Mangasarian and Pang (1995) and subsequently developed in Gowda (1995, 1996) and Sznajder and Gowda (1995). Also called the extended linear complementarity problem is another variant expounded by De Schutter and De Moor (1996) that captures the previously mentioned HLCP, VLCP and ELCP.

The Nonlinear Complementarity Problem

The NCP($\boldsymbol{f}$) as an identified problem first appeared in Cottle (1964, 1966). There—under very strong assumptions on f—an existence theorem and an

analogue of the principal pivoting method for the LCP were presented. As described in Pang (1995), contemporary iterative NCP algorithms tend to fall into three categories: (i) the basic Newton method, (ii) nonsmooth-equations approaches, and (iii) interior-point methods. Some algorithms are inspired by the equivalence between the NCP(f) and the variational inequality problem $\mathrm{VI}(X, f)$ in the case where $X = R^n_+$. Some seek zeros of a function such as g defined in (3) whereas others attack the nonlinear program (4) or a variant thereof. Despite the existence of several fine collections of research articles on nonlinear complementarity problems, the authoritative surveys of Harker and Pang (1990) and Pang (1995) came as close as anything then available to a monograph on this topic. The field now benefits from the publication of the masterful two-volume work on variational inequalities and complementarity problems by Facchinei and Pang (2003).

Software for Complementarity Problems

Information about available software for (mixed) complementarity problems can be found by searching the World Wide Web.

See

- ▶ Complementarity Applications
- ▶ Complementary Slackness Theorem
- ▶ Game Theory
- ▶ Matrices and Matrix Algebra
- ▶ Nonlinear Programming
- ▶ Quadratic Programming

References

Cottle, R. W. (1964). *Nonlinear programs with positively bounded Jacobians*, Ph.D. Thesis, Department of Mathematics, University of California, Berkeley. (See also, Technical Report ORC 64–12 (RR), Operations Research Center, University of California, Berkeley.)

Cottle, R. W. (1966). Nonlinear programs with positively bounded Jacobians. *SIAM Journal on Applied Mathematics, 14*, 147–158.

Cottle, R. W. (2010). A field guide to the matrix classes found in the literature of the linear complementarity problem. *Journal of Global Optimization, 46*, 571–580.

Cottle, R. W., & Dantzig, G. B. (1968). Complementary pivot theory of mathematical programming. *Linear Algebra and Its Applications, 1*(1968), 103–125.

Cottle, R. W., & Dantzig, G. B. (1970). A generalization of the linear complementarity problem. *Journal of Combinatorial Theory, 8*, 79–90.

Cottle, R. W., Pang, J. S., & Stone, R. E. (1992). *The linear complementarity problem*. Boston: Academic Press.

Cottle, R. W., Pang, J. S., & Stone, R. E. (2009). *The linear complementarity problem. Classics in applied mathematics*. Philadelphia: SIAM.

De Schutter, B., & De Moor, B. (1996). The extended linear complementarity problem. *Mathematical Programming, 71*, 289–326.

Eaves, B. C. (1971). The linear complementarity problem. *Management Science, 17*, 612–634.

Eaves, B. C., & Lemke, C. E. (1981). Equivalence of LCP and PLS. *Mathematics of Operations Research, 6*, 475–484.

Ebiefung, A. A. (1995). Existence theory and Q-matrix characterization for the generalized linear complementarity problem. *Linear Algebra and Its Applications, 223*(224), 155–169.

Facchinei, F., & Pang, J. S. (2003). *Finite-dimensional variational inequalities and complementarity problems*. New York: Springer.

Ferris, M. C., & Pang, J. S. (1997). Engineering and economic applications of complementarity problems. *SIAM Review, 39*, 669–713.

Gowda, M. S. (1995). On reducing a monotone horizontal LCP to an LCP. *Applied Mathematics Letters, 8*, 97–100.

Gowda, M. S. (1996). On the extended linear complementarity problem. *Mathematical Programming, 72*, 33–50.

Habetler, G. J., & Price, A. J. (1971). Existence theory for generalized nonlinear complementarity problems. *Journal of Optimization Theory and Applications, 7*, 223–239.

Harker, P. T., & Pang, J. S. (1990). Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications. *Mathematical Programming, Series B, 48*, 161–220.

Howson, J. T., Jr. (1963). *Orthogonality in linear systems*, Ph.D. Thesis, Department of Mathematics, Rensselaer Institute of Technology, Troy, New York.

Isac, G. (1992). *Complementarity problems, lecture notes in mathematics 1528*. Berlin: Springer-Verlag.

Isac, G. (2000). *Topological methods in complementarity theory*. Dordrecht: Kluwer.

Isac, G., Bulavsky, V. A., & Kalashnikov, V. V. (2002). *Complementarity, equilibrium, efficiency, and economics*. Dordrecht: Kluwer.

Karamardian, S. (1971). Generalized complementarity problem. *Journal of Optimization Theory and Applications, 8*, 161–168.

Kojima, M., et al. (1991). *A unified approach to interior point algorithms for linear complementarity problems*. Berlin: Springer-Verlag. lecture notes in computer science 538.

Lemke, C. E. (1965). Bimatrix equilibrium points and mathematical programming. *Management Science, 11*, 681–689.

Lemke, C. E., & Howson, J. T., Jr. (1964). Equilibrium points of bimatrix games. *SIAM Journal on Applied Mathematics, 12*, 413–423.

Luo, Z. Q., Pang, J. S., & Ralph, D. (1996). *Mathematical programs with equilibrium constraints*. New York: Cambridge University Press.
Mangasarian, O. L., & Pang, J. S. (1995). The extended linear complementarity problem. *SIAM Journal on Matrix Analysis and Applications, 16*, 359–368.
Mangasarian, O. L., & Solodov, M. V. (1993). Nonlinear complementarity as unconstrained and constrained minimization. *Mathematical Programming, Series B, 62*, 277–297.
Mohan, S. R., & Neogy, S. K. (1997). Vertical block hidden Z-matrices and the generalized linear complementarity problem. *SIAM Journal on Matrix Analysis and Applications, 18*, 181–190.
Murty, K. G. (1968). *On the number of solutions to the complementarity problem and spanning properties of complementary cones*. Ph.D. Thesis, Department of Industrial Engineering and Operations Research, University of California, Berkeley.
Murty, K. G. (1972). On the number of solutions to the complementarity problem and spanning properties of complementary cones. *Linear Algebra and Its Applications, 5*, 65–108.
Murty, K. G. (1988). *Linear complementarity, linear and nonlinear programming*. Berlin: Heldermann-Verlag.
Pang, J. S. (1995). Complementarity problems. In R. Horst & P. Pardalos (Eds.), *Handbook of global optimization* (pp. 271–338). Dordrecht: Kluwer.
Robinson, S. M. (1979). Generalized equations and their solutions, part I: Basic theory. *Mathematical Programming Study, 10*, 128–141.
Samelson, H., Thrall, R. M., & Wesler, O. (1958). A partition theorem for Euclidean n-space. *Proceedings of the American Mathematical Society, 9*, 805–807.
Sznajder, R., & Gowda, M. S. (1995). Generalizations of *Po*-and *P*-properties; extended vertical and horizontal LCPs. *Linear Algebra and Its Applications, 223/224*, 695–715.
Tütüncü, R. H., & Todd, M. J. (1995). Reducing horizontal linear complementarity problems. *Linear Algebra and Its Applications, 223*(224), 717–730.
Zhang, Y. (1994). On the convergence of a class of infeasible interior-point algorithm for the horizontal linear complementarity problem. *SIAM Journal on Optimization, 4*, 208–227.

Complementary Pivot Algorithm

► Quadratic Programming

Complementary Slackness Theorem

For the symmetric form of the primal and dual problems the following theorem holds: For optimal feasible solutions of the primal and dual (symmetric) systems, whenever inequality occurs in the *k*th relation of either system (the corresponding slack variable is positive), then the *k*th variable of its dual is zero; if the *k*th variable is positive in either system, the *k*th relation of its dual is equality (the corresponding slack variable is zero). Feasible solutions to the primal and dual problems that satisfy the complementary slackness conditions are also optimal solutions. A similar theorem holds for the unsymmetric primal-dual problems: For optimal feasible solutions of the primal and dual (unsymmetric) systems, whenever the *k*th relation of the dual is an inequality, then the *k*th variable of the primal is zero; if the *k*th variable of the primal is positive, then the *k*th relation of the dual is equality. This theorem just states the optimality conditions of the simplex method.

See

► Complementarity Applications
► Complementarity Condition
► Complementarity Problems
► Symmetric Primal-Dual Problems
► Unsymmetric Primal-Dual Problems

Complex Problem Analyzing Method (Compram)

Dorien J. DeTombe
International Research Society on Methodology of Societal Complexity, Amsterdam, The Netherlands

Complex societal problems are worldwide natural problems caused by viruses such as the flu pandemic, fowl plague, and HIV/AIDS; local natural disasters especially earthquakes, hurricanes, avalanches and floods; technical dangers caused by industry including pollution (CO2), traffic, and nuclear power plants; climate change and agricultural activities; man-made threats such as wars, terrorism, internet vulnerability, stock exchange manipulation, credit crises, and identity theft. The concept of societal complexity and an approach to their resolution, the Complex Problem Analyzing Method (Compram),

are discussed by DeTombe (2001). Compram is based on the idea that societal problems must be handled in a multi-disciplinary and cooperative manner by experts, stakeholders, and policy makers. Compram combines aspects of different methods into a structured interactive approach for policy making to find possible transitions of the situation that can be mutually accepted and implemented (DeTombe 1994).

The related difficult and complicated group processes are guided and structured by a facilitator. Those involved discuss the content and possible solutions based on a cooperative (simulation) model of the problem. The methodology emphasizes facilitating the exchange of knowledge, and understanding and communication between the participants. Compram has been used on a theoretical basis for handling over sixty real-life cases in the field of societal policymaking. The Organisation for Economic Cooperation and Development (OECD) suggests that the analysis of a complex societal problem be supported by the application of Compram (OECD 2006). (Further information on Compram and Societal Complexity can be found on Web sites maintained by the author).

See

- ▶ Community OR
- ▶ Soft Systems Methodology
- ▶ Wicked Problems

References

DeTombe, D. J. (1994). *Defining complex interdisciplinary societal problems*. A theoretical study for constructing a co-operative problem analyzing method: the method Compram. Thesis publishers Amsterdam, ISBN 90 5170 302-3.

DeTombe, D. J. (2001). Compram, a method for handling complex societal problems. *European Journal of Operational Research, 128*(2), 266–281.

OECD. (2006). *Report on global safety*. Report on the workshop on science and technology for a safer society, July 2006, Paris.

Compram

- ▶ Complex Problem Analyzing Method (Compram)

Computational Biology

Harvey J. Greenberg[1] and Allen G. Holder[2]
[1]University of Colorado-Denver, Denver, CO, USA
[2]Rose-Hulman Institute of Technology, Terre Haute, IN, USA

Computational biology is an interdisciplinary field that applies the techniques of computer science, applied mathematics, and statistics to address biological questions. OR is also interdisciplinary and applies the same mathematical and computational sciences, but to decision-making problems. Both focus on developing mathematical models and designing algorithms to solve them. Models in computational biology vary in their biological domain and can range from the interactions of genes and proteins to the relationships among organisms and species.

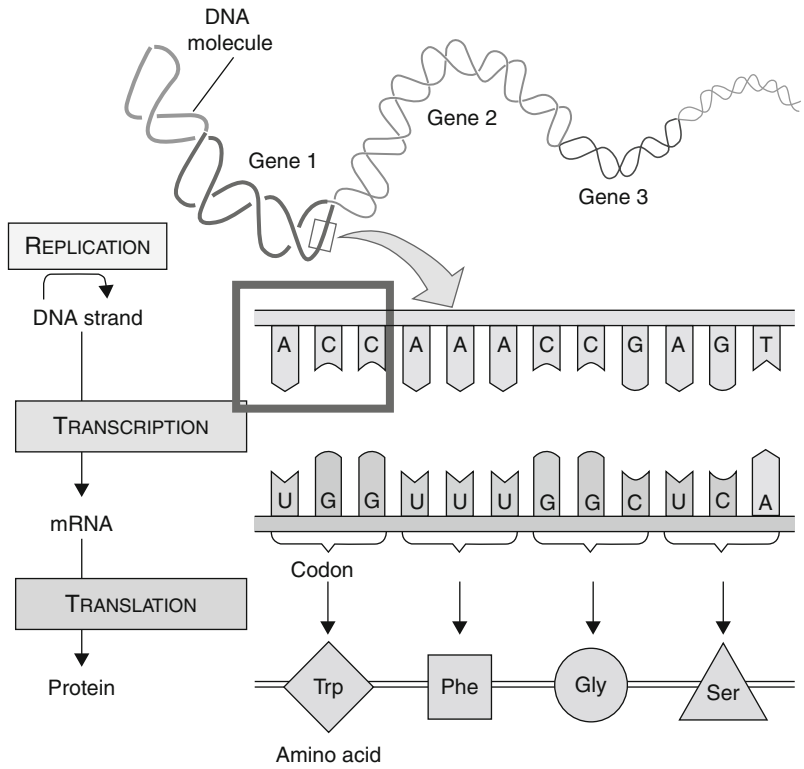
Genes are stretches of deoxyribonucleic acid (DNA), which is sometimes called the user manual for life and is a double-stranded helix of nucleic acids bonded by base-pairs of complements (`a-t`, `c-g`). The central dogma of molecular biology asserts that information in a cell flows from DNA to ribonucleic acid (RNA) to protein (note, Francis Crick used dogma when he introduced this in 1958 to mean without foundation because there was no experimental evidence at that time). Proteins are the workers of the cell, and there is much focus on recognizing, predicting, and comparing their properties (Fig. 1).

Proteins interact either directly by modifying each other's properties through direct contact or indirectly by participating in the production and modification of cellular metabolites. Collectively, the biochemical reactions and the possible intermediates that produce a metabolite comprise a metabolic pathway, and a metabolic network is a collection of these pathways. The study of complex networks like that of the metabolism is called systems biology.

Linear Programming: A linear program (LP) is an optimization problem in which the variables are in $\mathbb{R}^N$, and the constraints and the objective are linear.

Flux Balance Analysis (FBA) – A biochemical process is defined by n reactions that convert m compounds:

$$a_{1j}x_1 + \cdots + a_{mj}x_m \underset{k_-^j}{\overset{k_+^j}{\rightleftharpoons}} b_{1j}x_1 + \cdots + b_{mj}x_m,$$



Computational Biology, Fig. 1 Central dogma of molecular biology

where x_i is the concentration of the ith compound, and $k_{\pm}^{j}$ is the jth reaction rate (for a 2-way reaction the reverse rate need not equal the forward rate). The corresponding ODE is:

$$\frac{\mathrm{d}x_i(t)}{\mathrm{d}t} = \sum_{j=1}^{n} (b_{ij} - a_{ij}) \left(k_{+}^{j} x_1^{a_{1j}} \cdots x_m^{a_{mj}} - k_{-}^{j} x_1^{a_{1j}} \cdots x_m^{a_{mj}} \right)$$
$$= \sum_{j=1}^{n} S_{ij} v_j(x),$$

where v is the flux (production or consumption of mass per unit area per unit time), and S_{ij} is defined as a stoichiometric (pronounced `stoy-kee-uh-me'-trik`) coefficient. These coefficients are interpreted as:

$S_{ij} > 0 \Rightarrow$ rate of compound i produced in reaction j;
$S_{ij} < 0 \Rightarrow$ rate of compound i consumed in reaction j.

The following holds asymptotically provided that the system approaches a steady state toward equilibrium concentrations $\bar{x}$:

$$\lim_{t \to \infty} \frac{\mathrm{d}x(t)}{\mathrm{d}t} = Sv(\bar{x}) = 0. \tag{1}$$

Dropping the dependence of the flux on $\bar{x}$, the flux cone is defined by this homogeneous system plus non-negativity for one-way reactions, indexed by J:

$$\mathcal{F} = \{v : Sv = 0, \;\; v_J \geq 0\}. \tag{2}$$

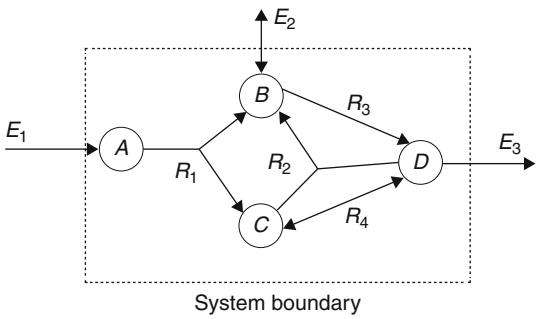
In a metabolic network, reactions are distinguished between external and internal. The flux associated with an external reaction is an exchange between the network of interest and the cell's environment (Fig. 2).

The stoichiometric matrix for the internal reactions is extended to include external reactions, each being a singleton column with ± 1:

$$S = \begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cccc|ccc} R_1 & R_2 & R_3 & R_4 & E_1 & E_2 & E_3 \end{array} \\ \left[\begin{array}{cccc|ccc} -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & -2 & 0 & 0 & -1 & 0 \\ 2 & -1 & 0 & -2 & 0 & 0 & 0 \\ 0 & -2 & 1 & 3 & 0 & 0 & -1 \end{array} \right] \end{array}$$

Computational Biology, Fig. 2 Example metabolic network with four internal and three external reactions

$R_1 : A \rightharpoonup B + 2C$ (multiple output)
$R_2 : C + 2D \rightharpoonup B$ (multiple input)
$R_3 : 2B \rightharpoonup D$ (simple)
$R_4 : 2C \leftrightharpoons 3D$ (simple, 2-way)
$E_1 : \rightharpoonup A$ (supply)
$E_2 : B \leftrightharpoons$ (2-way exchange)
$E_3 : D \rightharpoonup$ (demand)



All reactions are one-way reactions, except R_4 and E_2, so $J = \{1, 2, 3, 5, 7\}$, leaving v_4 and v_6 without sign restriction in the flux cone.

Strictly speaking, a metabolic network is usually not a network in the OR sense because some internal reactions have multiple inputs or outputs (sometimes called a process network in chemical engineering). Hence, LP is used, rather than specialized network algorithms, to find fluxes. The FBA LP model has the form:

$$\max \quad c^T v : \quad v \in \mathcal{F} \cap \mathcal{B}, \tag{3}$$

where $\mathcal{B}$ is a bounding set so that the linear program has an optimal solution if it is feasible. A common objective is to maximize the rate of growth defined in terms of metabolites, where the objective coefficients (c) depend on the organism. Other objectives include maximizing some metabolite production, minimizing by-product production, minimizing substrate requirements, and minimizing mass nutrient uptake (Palsson 2006).

An optimal basis depends on the definition of $\mathcal{B}$. Three possibilities, which may be combined, are:

$$\begin{aligned} &\text{simple bound :} && L_K \le v_K \le U_K \\ &\text{fixing inputs and/or outputs :} && v_K = \bar{v}_K \\ &\text{normalization :} && \textstyle\sum_{j \in K} v_j = b, \end{aligned}$$

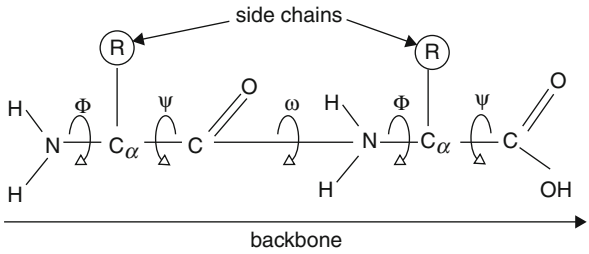
where K is a subset of reactions. Inputs and outputs are generally a subset of the exchanges. Normalization applies to one-way reactions – i.e., $K \subseteq J$. Each extreme ray of the flux cone corresponds to an extreme point of the polytope. The converse is generally not true – viz., fixing the flux of a reaction that transports metabolites in or out of the cell can introduce extreme points with no extreme ray of the flux cone passing through them.

Pathways are subnetworks with a single biological effect. In an ordinary network, where each internal reaction has a single input and output, this is a path. A cut set is defined as a set of reactions whose removal renders the stoichiometric (1) infeasible for a specified output. For an ordinary network, the OR terminology is a disconnecting set. A minimal cut set for a specified output is, in OR terminology, simply a cut set. For the example, a cut set that separates D from the rest of the network is $\{R_1, R_3, R_4, E_1\}$. Finding a (minimal) cut set in the general case becomes an IP, using binary variables to block pathways to some specified output.

Nonlinear Programming: A nonlinear program (NLP) is defined by having the objective or some constraint function be nonlinear in the decision variables.

Protein Folding – Most proteins go through a process that twists and turns the molecules from their primary state of a linear progression of amino acids to a native three-dimensional state in which it remains. That process is called folding, and it is theoretically possible to predict a protein's native state, or structure, by knowing its primary state. This determines a protein's function, and some diseases (e.g., Alzheimer's, Huntington's, and cystic fibrosis) are associated with protein misfolding.

Predictive models became possible following the work of Christian B. Anfinsen, who in 1961 published experimental results supporting the Thermodynamic Hypothesis: A protein's native state is uniquely determined by its primary sequence; it transitions to a state of minimum free energy. This leads to a nonlinear program with the decision space defined as the spatial coordinates of atoms, constrained by the biochemistry of a protein's defining amino acid sequence. The objective function is a free energy determined by potential energies from atomic bonds and non-bond interactions.



Computational Biology, Fig. 3 Covalent bonds along the backbone result in a residue for each of the amino acids. The torsion angles are denoted by Ψ and Φ; ω is the dihedral angle

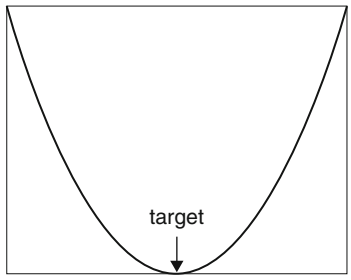
The bonds for the sequence of amino acids shown in Fig. 3 are covalent, meaning that they share electrons, and these strong bonds hold the backbone together. Objective terms for the ith covalent bond include the energies required to stretch, bend, and twist the bond.

Action	Energy	
Stretching	E^{stretch}	$= \sum_i K_i^L \left(L_i - L_i^0\right)^2$
Bending	E^{bend}	$= \sum_i K_i^\theta \left(\theta_i - \theta_i^0\right)^2$
Twisting	E^{twist}	$= \sum_i K_i^\phi (1 - \cos(\omega_i))$

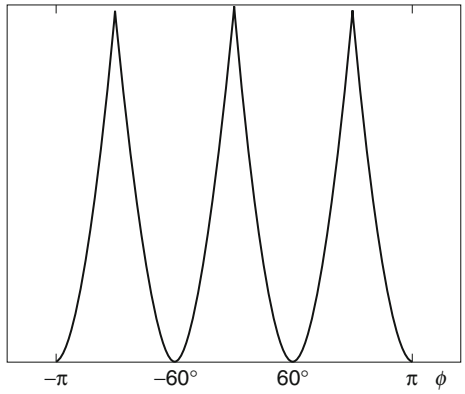
The variables are the bond length (L) and the bond angles, ω and $\theta = (\Psi, \Phi)$, which are determined by atomic coordinates. Parameters include target values (L^0, θ^0). Weight parameters (K) are scale factors that put the energy terms in the same unit; those values can be measured or derived. For example, if it requires 100 kcal/mole to break a bond, and two positive charges within 3.3 Å (Angstrom) have at least 100 kcal/mole, then the total energy is reduced by breaking a bond to keep positive charges distant. Estimating these values to determine weight parameters is not an exact science, so even these basic energy functions are inexact, and there are other energy functions for non-covalent bonds and among non-bonding atoms.

Two common energy functions estimate the electrostatic and Van der Waals interactions:

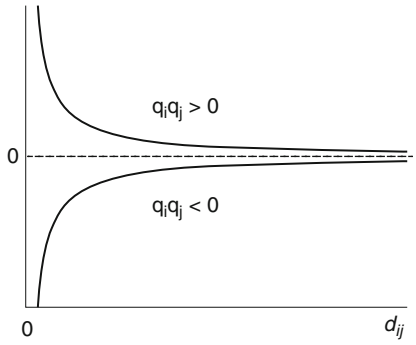
Action	Energy	
Electrostatic	E^{elec}	$= \sum_{i<j} K_{ij}^{\text{elec}} \frac{q_i q_j}{d_{ij}}$
Van der Waals	E^{vdw}	$= \sum_{i<j} K_{ij}^{\text{vdw}} \left(\left(\frac{d_{ij}^*}{d_{ij}}\right)^{12} - \alpha_{ij} \left(\frac{d_{ij}^*}{d_{ij}}\right)^{6} \right)$



Computational Biology, Fig. 4 The squared deviation of E^{stretch} and E^{bend} is convex



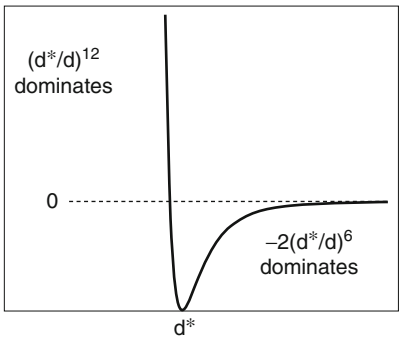
Computational Biology, Fig. 5 E^{twist} with $\omega = 3/2(\phi - \pi)$



Computational Biology, Fig. 6 E^{elec} depends on the sign of $q_i q_j$. Oppositely signed atoms attract, so the energy is negative and favors them being close

The variables are the pairwise distances (d), which are determined by the atomic coordinates. Parameters are the atomic charges (q) and equilibrium distances (d^*) (Figs. 4–7).

The NLP approach (Floudas and Pardalos 2000) uses energy principles that underlie molecular



Computational Biology, Fig. 7 Lennard-Jones approximation of E^{vdw} for $\alpha = 2$

dynamics, and these methods attempt to find the native state and a pathway to it. In practice, not all parameters are grounded in some physical law. An energy function could include contributions from non-bonded and uncharged pairs, based on their distance and radii. Alternatively, known structures can be used to predict an unknown structure, based on their evolutionary similarity. This is called homology, and it is focused on determining the native state and not on discerning the dynamic pathways to reach it.

The multi-modal shape of the energy landscape leads to the Levinthal Paradox: Many proteins reach their native state within milliseconds, yet the number of stable conformations grows exponentially in the number of amino acids. One explanation is that proteins fold into a nearby local minimum of the free energy instead of the global minimum. Global optimization methods based on this principle are called funneling methods. Another explanation is that the dimension of the problem is not the length of the amino acid sequence but is instead the number of chains that obey patterns not fully understood. Combinatorial optimization methods based on this principle are called chain growth and zipping and assembly algorithms.

Comparing Protein Function – A protein's function is determined by its 3D native state, of which many confirmations are known. Comparing protein structures relates protein function and collects proteins into functionally similar families that help identify a protein's functions.

Proteins typically have multiple functional domains, each of which would act as an independent protein if its amino acid subsequence had folded independently. Two proteins are considered to be functionally similar if they share a (nearly) common domain. Each domain is composed of secondary structures, notably α-helices and β-sheets, illustrated in Fig. 8. In structure alignment the goal is to best align the secondary structures between two proteins' domains. The input to the alignment problem is a set of coordinates for the C_α atoms for each domain – i.e., the spatial coordinates for the carbon atoms linked to the side chains (c.f., Fig. 3).

To remove a dependency on rigid body motion, structures are often aligned with respect to pairwise distances, d_{ij}, which is a measure between the ith and jth C_α atoms. Let d'_{ij} and d''_{kr} be the intra-distance measures for the two domains, and consider the binary variable:

$$x_{ik} = \begin{cases} 1 & \text{if the } i^{th}\ C_\alpha \text{ atom of the first domain is paired with} \\ & \text{the } k^{th}\ C_\alpha \text{ atom of the second domain;} \\ 0 & \text{otherwise.} \end{cases}$$

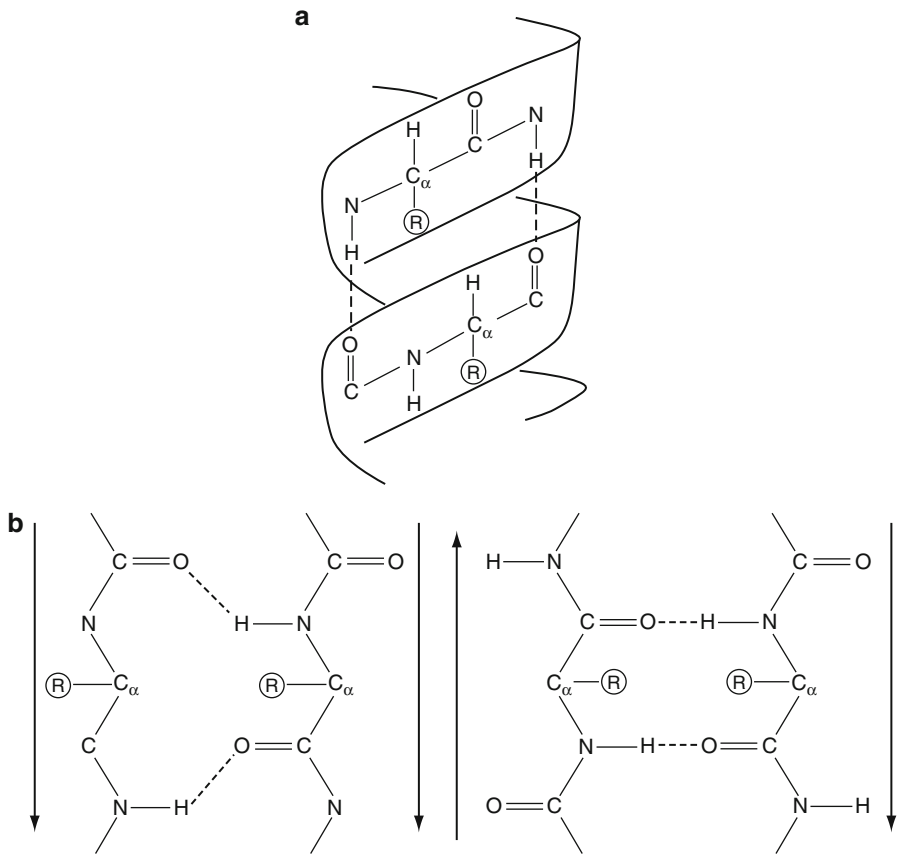
An optimal pairing between the two domains can be calculated by solving a quadratic integer program:

$$\max \sum_{i,k,j,r} x_{ik} x_{jr} d'_{ij} d'_{kr} : \sum_k x_{ik} \le 1,$$
$$\sum_i x_{ik} \le 1,\ x_{ik} = 0,\ (i,k) \in \mathcal{S},$$

where $(i, k) \in \mathcal{S}$ if the ith and kth C_α atoms are in different types of secondary structures.

Besides the choice of metric, a variation is to allow pairings between C_α atoms whose secondary structures are different. This is accommodated by removing the restriction that $x_{ik} = 0$ for $(i, k) \in \mathcal{S}$ and adding penalty terms in the objective: $-\sum_{(i,k)\in\mathcal{S}} p_{ik} x_{ik}$. The problem as stated includes the possibility of a non-sequential alignment, i.e., one in which the C_α atoms can be paired independent of the amino acid sequence. A combinatorial optimization model of alignments that requires the same ordering of the amino acid residues is called contact map optimization (Burkowski 2009; Glodzik and Skolnick 1994; Goldman et al. 1999).

Integer Programming: An integer program (IP) is an optimization problem in which some or all of the variables are restricted to be integer valued. For combinatorial optimization, the integer values are simply $\{0, 1\}$.



Computational Biology, Fig. 8 Secondary structures formed along the backbone define a protein's shape. *Dotted lines* represent hydrogen bonds; ® represents a side chain. (**a**) α-Helix, most closely packed arrangement of residues, defined by three parameters: pitch, rise, and turn. (**b**) *β*-Sheets form if the backbone is loosely packed, almost fully extended; they can be parallel (*left*), antiparallel (*right*), or a mixture

Pathway Analysis – Consider the FBA model (3) with added binary variables associated with each process with finite bounds (given or derived), $L_j \le v_j \le U_j$:

$$y_j = \begin{cases} 1 & \text{if } v_j \neq 0; \\ 0 & \text{otherwise.} \end{cases}$$

Replacing the bound constraints with $L_j y_j \le v_j \le U_j y_j$ forces $v_j = 0$ if $y_j = 0$. This corresponds to excluding reaction j, which is called a knock-out. Drug side effects are caused by unintended knock-outs, which, if cannot be avoided, can at least be identified and minimized. In drug design, one may want to block all pathways to some final output. If P is a pathway leading to the targeted output, then adding the constraint

$$\sum_{j \in P} y_j \le |P| - 1$$

removes the pathway, where $j \in P$ if pathway P contains reaction j.

A cut set can be computed with successive pathway-generation for a specified output and adding its pathway-elimination constraint. For the example in Fig. 2, pathways to produce D can be generated by fixing $v_7 = 1$ (and not have y_7). The first basic optimal solution uses reactions R_1, R_3, R_4, E_1, E_3. This leads to the addition of the constraint:

$$y_1 + y_3 + y_4 + y_5 \le 3.$$

The next pathway generated is R_3, E_1, and $y_3 = 0$ satisfies both pathway constraints. After eliminating R_3, the solution is R_1, R_4, E_1, E_3.

Other logical constraints include process conflict, $y_j + y_{j'} \le 1$ (i.e., inclusion of j requires exclusion of j'), and process dependence, $y_j \ge y_{j'}$ (i.e., exclusion of j requires exclusion of j'), for $j \neq j'$.

Rotamer Assignment – Part of the protein folding problem is knowing the side-chain conformations – i.e., knowing the torsion angles of the bonds (c.f., Fig. 3). The rotation about a bond is called a rotamer, and there are libraries that give configuration likelihoods, for each amino acid (from which energy values can be derived). The rotamer

assignment (RoA) problem is to find an assignment of rotamers to sites that minimizes the total energy of the molecule. For the protein folding problem, the amino acid at each site is known. There are about 10–50 rotamers per amino acid, depending on what else is known (such as knowing that the amino acid is located in a helix), so there are about 10^n to 50^n rotamer assignments for a protein of length n.

Let r be in the set of rotamers that can be assigned to site i, denoted by $\mathcal{R}_i$, and let

$$x_{ir} = \begin{cases} 1 & \text{if rotamer } r \text{ is assigned to site } i; \\ 0 & \text{otherwise.} \end{cases}$$

An optimal assignment is a solution to the quadratic semi-assignment problem:

$$\min \sum_i \sum_{r \in R_i} \left(\mathcal{E}_{ir} x_{ir} + \sum_{j>i} \sum_{t \in R_j} E_{irjt} x_{ir} x_{jt} \right): \\ \sum\nolimits_{r \in R_j} x_{ir} = 1 \, \forall i, \quad x \in \{0, 1\}.$$

The objective function includes two types of energy: (1) within a site, E_{ir}, and (2) between rotamers of two different sites, $\mathcal{E}_{irjt}$ for $i \neq j$. The summation condition $j > i$ avoids double counting, where $E_{irjt} = E_{jtir}$.

Besides its role in determining a protein's structure, the RoA problem is useful in drug design. Specifically, the RoA problem can be used to determine a minimum-energy docking site for a ligand, which is a small molecule such as a hormone or neurotransmitter that binds to a protein and modifies its function. The ligand-protein docking problem is characterized by only a few sites, and if the protein is known, the dimensions are small enough that the RoA problem can be solved exactly. However, if the protein is to be engineered, then there can be about 500 rotamers per site (20 acids @ 25 rotamers each), in which case solutions are computed with metaheuristics or approximation algorithms. There are other bioengineering problems associated with the RoA problem, such as determining protein-protein interactions. While the mathematical structure is the same, the applications have different energy data, which can affect algorithm performance (Forrester and Greenberg 2008).

Also see Clote and Backofen (2000), Jones and Pevzner (2004), and Lancia (2006).

Dynamic Programming: This is a computational approach to sequential decision making. Two fundamental biological sequences are taken from the alphabet of nucleic acids, {a, c, g, t}, and from the alphabet of amino acids, {A, R, N, D, C, Q, E, G, H, I, L, K, M, F, P, S, T, W, Y, V}. The former is a segment of DNA (or RNA if u replaces t – i.e., uracil instead of thymine); the latter is a protein segment.

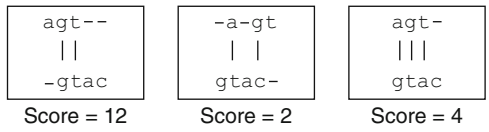
Sequence Alignment – Two sequences can be optimally aligned by dynamic programming, where optimal is one that maximizes an objective that has two parts:

1. A *scoring function*, given in the form of an $m \times m$ matrix S, where m is the size of the alphabet. The value of S_{ij} measures a propensity for the i^{th} alphabet-character in one sequence to align with the j^{th} alphabet-character in some position of the other sequence.
   Example: Let $s =$ agt and $t =$ gtac. If the first character of s is aligned with the first character of t, then the score is S_{ag}, which is the propensity for a to be aligned with g.
2. A *gap penalty function*, expressed in two parts: a fixed cost of beginning a gap, denoted G_{open}, and a cost to extend the gap, denoted G_{ext}.
   Example: Let $s =$ agt and $t =$ gtac. One alignment is $\begin{matrix} \mathtt{agt-} \\ \mathtt{gtac} \end{matrix}$, which puts a gap at the end of the first sequence.

A gap is called an indel because it can be either an insertion into one sequence or a deletion from the other sequence: insert $\downarrow$ $\begin{bmatrix} - \\ \mathtt{a} \end{bmatrix}$ delete $\uparrow$ If one sequence evolved directly from the other, the evolutionary operation is determined by their time-order. If they have a common ancestor, they evolved along different paths, resulting in the indel when comparing them. The evolutionary biology explains why sequences can be more similar than a simple alignment (without gaps) may suggest.

Figure 9 shows three different alignments for the two nucleic acid sequences, agt and gtac. Scores are shown for the following scoring matrix and do not account for gapping:

$$S = \begin{matrix} \mathtt{a} & \mathtt{c} & \mathtt{g} & \mathtt{t} \end{matrix} \\ \begin{bmatrix} 6 & 1 & 2 & 1 \\ 1 & 6 & 1 & 2 \\ 2 & 1 & 6 & 1 \\ 1 & 2 & 1 & 6 \end{bmatrix} \begin{matrix} \mathtt{a} \\ \mathtt{c} \\ \mathtt{g} \\ \mathtt{t} \end{matrix}$$



Computational Biology, Fig. 9 Three alignments for two sequences

If the objective is a linear affine function of gap lengths, the total objective function for the 2-sequence alignment problem is:

$$\sum_{i,j} S_{s_i t_j} - G_{\text{open}}(N_s + N_t) - G_{\text{ext}}(M_s + M_t),$$

where the sum is over aligned characters, s_i from sequence s with t_j from sequence t. The number of gaps opened is N_s in sequence s and N_t in sequence t; the number of gap characters (–) is M_s in sequence s and M_t in sequence t. In the examples of Fig. 9, if $G_{\text{open}} = 2$ and $G_{\text{ext}} = 1$, the gap penalties are 7, 9, and 3, respectively.

The alphabet is extended to include the gap character, with S extended to include gap extension, as $S_{a-} = S_{-a} = G_{\text{ext}}$ for all a in the alphabet. (So, G_{ext} includes the penalty for the first alignment with -.) Let s^i denote the subsequence $(s_1, \ldots, s_i)$, with $s^0 = \varnothing$. Here is the DP recursion for $G_{\text{open}} = 0$:

$$F(s^i, t^j) = \max \begin{cases} F(s^{i-1}, t^{j-1}) + S_{s_i t_j} & \text{match;} \\ F(s^{i-1}, t^j) + S_{s_i -} & \text{insert} - \text{into } t; \\ F(s^i, t^{j-1}) + S_{-t_j} & \text{insert} - \text{into } s. \end{cases} \tag{4}$$

The initial conditions are:

$$\begin{aligned} &F(\emptyset, \emptyset) = 0; \\ &F(s^i, \emptyset) = F(s^{i-1}, \emptyset) + S_{s_i -}, \quad i = 1, \cdots |s|; \\ &F(\emptyset, t^j) = F(\emptyset, t^{j-1}) + S_{-t_j}, \quad j = 1, \cdots |t|. \end{aligned}$$

The DP recursion (4) is for global alignment, and it has been extended to allow $G_{\text{open}} > 0$ and to not penalize leading or trailing gaps (allowing a short sequence to be aligned with a large one meaningfully). Local alignment is finding maximal substrings (contiguous subsequences) with an optimal global alignment having maximum score (Gusfield 1997; Waterman 1995).

Sequences from many species can be compared simultaneously in a Multiple Sequence Alignment (MSA). One way to evaluate an MSA is by summing pairwise scores. Figure 10 shows an example. The sum-of-pairs score, based on the scoring matrix S, is shown for each column. For example, column 1 has $3S_{\texttt{aa}} + 3S_{\texttt{ac}} = 3$. The sum-of-pairwise scores for column 2 is zero because gap scores are not shown by columns; they are penalized for each sequence (rows of alignment) with $G_{\text{open}} = 2$ and $G_{\text{ext}} = 1$. The total objective value is $152 - 37 = 115$.

MSA is a computational challenge to exact DP due to the combinatorial explosion of the state space, but one could use approximate DP or formulate MSA as an IP.

Phylogenetic Tree Construction – Phylogeny is the evolutionary history of some biological entity. A phylogenetic tree (PT) is a graphical presentation of a phylogeny. A leaf represents an Operational Taxonomic Unit (OTU), which can be various levels – e.g., species, genes, pathways, and enzymes. Each edge, or branch, is a relation between a pair of OTUs. Each internal node is constructed so that the resulting PT is consistent with the OTU data, and the root represents a common ancestor of the OTUs.

Example. Consider five OTUs and an MSA of DNA sites with six base-pairs (Fig. 11):

	site					
OTU	1	2	3	4	5	6
A	c	a	g	a	c	a
B	c	a	g	g	t	a
C	c	g	g	g	t	a
D	t	g	c	g	t	a
E	t	g	c	a	c	t

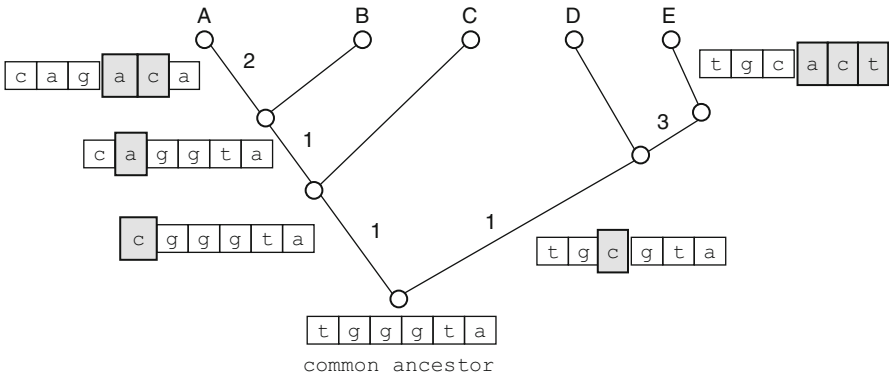
If the number of mutations is the distance between two sequences, then the distance between OTUs is the length of the unique path between them in the PT. The example has the distance matrix:

$$D = \begin{array}{c} \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{bmatrix} 0 & & & & \\ 2 & 0 & & & \\ 3 & 1 & 0 & & \\ 5 & 3 & 2 & 0 & \\ 8 & 6 & 5 & 3 & 0 \end{bmatrix} \end{array} \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix}$$

Computational Biology, Fig. 10 A multiple alignment of four sequences

													Gap penalty
	a	-	g	a	g	t	-	a	c	t	-	-	-	11
	a	a	g	t	a	t	-	-	a	t	-	-	-	9
	a	-	-	t	a	t	a	a	-	-	-	-	t	10
	c	-	g	t	a	-	-	a	c	t	c	c	t	7
score:	21	0	18	21	24	18	0	18	8	18	0	0	6	37

Total = 152



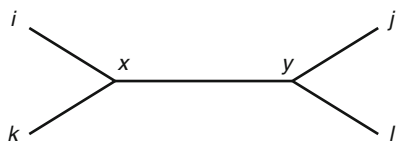
Computational Biology, Fig. 11 The example maximum-parsimony PT has eight mutations, shown on the branches. (All other PTs have more than 8.)

This is not the same as the MSA distance. For example, $D(A, E) = 8$ in the PT but is only 4 in the MSA.

Regardless of how the distance matrix is derived (MSA or not), there may not exist a PT that satisfies specified distances. For that to be true, it is necessary and sufficient that the metric be additive – i.e., for any four leaves, there exist labels i, j, k, ℓ such that

$$D(i,j) + D(k,\ell) = D(i,\ell) + D(j,k) \geq D(i,k) + D(j,\ell).$$

The reason for this is that there must be some splitting i, k from j, ℓ with an internal branch:



Additivity does not usually hold, so the problem is to construct a PT whose associated leaf-distance matrix, D, minimizes some function of nearness to the given D^0, such as $\|D - D^0\|$. This problem is NP-hard. Heuristics include sequential clustering: Un-weighted/Weighted Pair Group Method with Arithmetic Mean (UPGMA/WPGMA) and neighbor-joining algorithms.

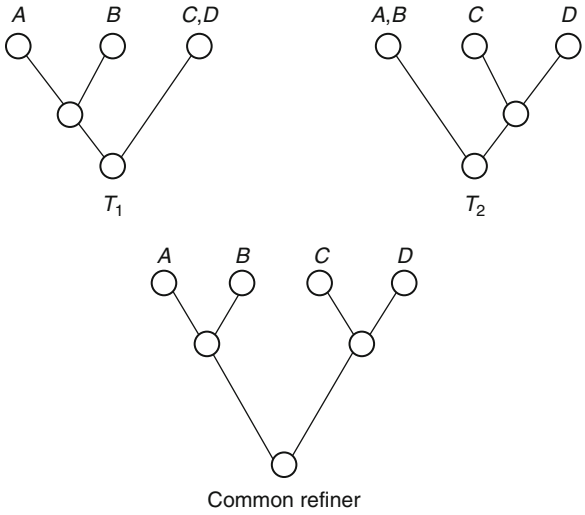
There may be multiple PTs, which generally come from different data – e.g., one from an MSA of a DNA segment, another from the maximum likelihood of some property. If a series of edge-contractions is applied to a PT, the resulting PT is called a refinement and the original is called a refiner. Two trees are compatible if they have a common refiner. One problem is to determine whether two PTs are compatible, and if so, what is their common refiner? If incompatible, how is a PT constructed that has some agreement with the given PTs?

A Matrix Representation with Parsimony (MRP) of a PT with k internal nodes is a binary matrix defined as:

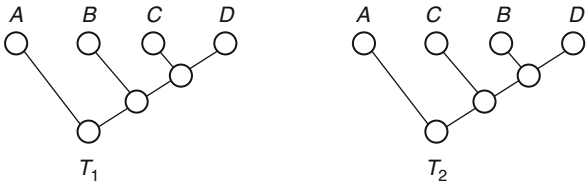
$$M_{ij} = \begin{cases} 1 & \text{if internal node } j \text{ is in the (unique) path} \\ & \text{from the root to OTU } i; \\ 0 & \text{otherwise.} \end{cases}$$

Conversely, given a binary matrix, if it has an associated PT, it is called a perfect phylogeny.

Given two PTs for the same OTUs with MRPs, M^1, M^2, their column-union is $[M^1\ M^2]$.



Computational Biology, Fig. 12 PTs T_1, T_2 are compatible



Computational Biology, Fig. 13 PTs T_1, T_2 are incompatible

Theorem. Two PTs are compatible if, and only if, their MRP column-union represents a perfect phylogeny.

The trees in Fig. 12 have the MRP column-union:

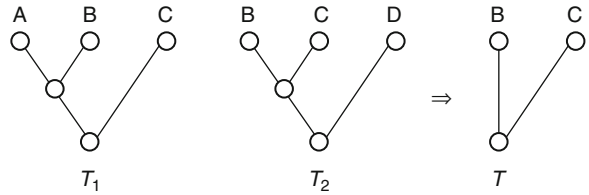
$$M = \begin{array}{c} M^1M^2 \\ \left[\begin{array}{c|c} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{array}\right] \end{array} \begin{array}{l} A \\ B \\ C \\ D \end{array}$$

This is the MRP of the common refiner in Fig. 12 and represents a perfect phylogeny.

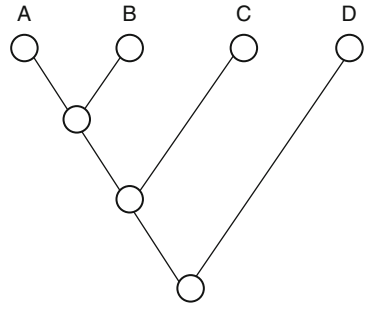
The MRP column-union of the PTs in Fig. 13 is:

$$M = \begin{array}{c} \quad M^1 \quad\quad M^2 \\ \left[\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{array}\right] \end{array} \begin{array}{l} A \\ B \\ C \\ D \end{array}$$

M does not correspond to any PT. (After drawing A, C, D with four internal nodes as the path to D, OTU B cannot be drawn with the path 0-1-3-4 without introducing the cycle, 1-2-3-1.)



Computational Biology, Fig. 14 A maximum agreement subtree with 2 of the 4 OTUs



Computational Biology, Fig. 15 An agreement supertree of the trees in Fig. 14

Suppose the trees are incompatible. A Maximum Agreement Subtree (MAST) is a refined subtree with the greatest number of leaves (Fig. 14).

The DP recursion for two subtrees (Steel and Warnow 1993) is nontrivial. The state is a pair of subtrees with specified roots, $\left(T_1^r, T_2^s\right)$. Each tree has an inclusion-ordered sequence of such subtrees, which is computed during the recursion. The decision space to compute $MAST\left(T_1^r, T_2^s\right)$, given $MAST\left(T_1^{r'}, T_2^{s'}\right)$ for $\left(T_1^{r'}, T_2^{s'}\right) \prec \left(T_1^r, T_2^s\right)$, requires the computation of a maximum weighted-matching on the complete r-s bipartite graph, weighted with $\{MAST(r', s')\}$.

Whereas MAST uses an intersection of PT information, a supertree uses their union. Construction methods vary, and some of the criteria address common order preservation. An agreement supertree, T, is a minimal tree such that each T_i is a refined subtree of T (Fig. 15).

Markov Chains and Processes: A stochastic process has the Markov property if the transition from one state to the next depends on only the current state. Classical models include the evolution of some biological states over time (Allen 2003; Wilkinson 2006). Molecular applications of Markov models also

consider ordered sequences of nucleotides (viz., DNA and RNA) and amino acids (viz., proteins).

CpG Island Recognition – In the human genome the appearance of the dinucleotide CG is rare because it causes the cytosine (C) to be chemically modified by methylation, which causes it to mutate into thymine (T). Methylation is suppressed around the promoters, or start regions, of many genes, and there are more CG dinucleotides than elsewhere. Such regions are called CpG islands, and they are typically a few hundred bases long. (CpG is used instead of CG to avoid confusion with a C–G base pair; the p is silent.) The recognition problem is: Given a short segment of a genomic sequence, decide if it is part of a CpG island.

Two Markov chains are defined: P^+ is the state-transition matrix within a CpG island; P^- is the state-transition matrix outside a CpG island. Each is applied to the given sequence and the log-odds ratio determines which is more likely.

Example. Consider a first-order Markov chain model with transition matrices determined by the frequencies in a database having more than 60,000 human DNA sequences:

$$P^+ = \begin{array}{c} \\ \mathtt{A} \\ \mathtt{C} \\ \mathtt{G} \\ \mathtt{T} \end{array} \begin{array}{c} \begin{array}{cccc} \mathtt{A} & \mathtt{C} & \mathtt{G} & \mathtt{T} \end{array} \\ \begin{bmatrix} 0.18 & 0.27 & 0.43 & 0.12 \\ 0.17 & 0.37 & 0.27 & 0.19 \\ 0.16 & 0.34 & 0.38 & 0.12 \\ 0.08 & 0.36 & 0.38 & 0.18 \end{bmatrix} \end{array}$$

$$P^- = \begin{array}{c} \\ \mathtt{A} \\ \mathtt{C} \\ \mathtt{G} \\ \mathtt{T} \end{array} \begin{array}{c} \begin{array}{cccc} \mathtt{A} & \mathtt{C} & \mathtt{G} & \mathtt{T} \end{array} \\ \begin{bmatrix} 0.30 & 0.20 & 0.29 & 0.21 \\ 0.32 & 0.30 & 0.08 & 0.30 \\ 0.25 & 0.25 & 0.30 & 0.20 \\ 0.18 & 0.24 & 0.29 & 0.29 \end{bmatrix} \end{array}$$

Given the sequence AACTTCG, its total log-odds ratio is

$$\sum_{i=1}^{6} \log_2\left(P^+_{s_i s_{i+1}} / P^-_{s_i s_{i+1}}\right) = -0.737 + 0.433 - 0.659 - 0.688 + 0.585 + 1.755 = 0.6888.$$

The conclusion is that the DNA segment is in a CpG island.

There is enough data to support the use of the more accurate 5th-order Markov chain, whose six-tuples correspond to two coding regions. At least 4^5 six-tuples are required in the database to estimate the conditional probabilities, $\Pr(x_6|x_1x_2x_3x_4x_5)$, which directly yield the state-transition probabilities:

$$\Pr(y_1y_2y_3y_4y_5|x_1x_2x_3x_4x_5) = \begin{cases} \Pr(x_6|x_1x_2x_3x_4x_5) & \text{if } y = (x_2x_3x_4x_5x_6); \\ 0 & \text{otherwise.} \end{cases}$$

For the particular example, there are only two state transitions, and the same database gives the transition probabilities:

$$P^+(\mathtt{C}|\mathtt{AACTT}) = 0.4 \qquad P^-(\mathtt{C}|\mathtt{AACTT}) = 0.2$$
$$P^+(\mathtt{G}|\mathtt{ACTTC}) = 0.1 \qquad P^-(\mathtt{G}|\mathtt{ACTTC}) = 0.3$$

In this case, the more accurate 5th-order chain yields the log-odds ratio $\log_2 0.4/0.2 + \log_2 0.1/0.3 = -0.585$, and the conclusion is that the DNA segment is not in a CpG island.

A host of related problems use the same Markov model. For example, transcription splices the DNA into coding regions, called exons, removing the remainder, called introns (misnamed junk DNA). A structure recognition problem is to identify exons versus introns.

Many of the structure recognition, comparison, and prediction problems have hidden states, but emissions are observed according to a known probability. These are Hidden Markov Models (HMMs) and are central in modern biology (Durbin et al. 1998).

Queueing Theory: A queue in a system is any set of objects awaiting service, and service is some process (es) involving the object.

T-Cell Signaling – A T-cell is a type of white blood cell distinguished by having a receptor – i.e., an ability to bind to other molecules. The receptor interacts with intracellular pathway components, starting a cascade of protein interactions called signal transduction. A way to view this process is that a T-cell receptor (TCR) enters a queue upon activation and goes through a series of processes, such as phosphorylation (Wedagedera and Burroughs 2006). Service completion is defined by the deactivation of the TCR, returning it to the inactive pool; however, it is possible that the T-cell's service is aborted before it completes service. Of interest is the probability of activation – i.e., in service for some threshold of

time. If it completes service and detects infection, the T-cell signals cell death (called apoptosis; the second p is silent).

Other queueing models apply to genetic networks, allowing signals that affect the population to enter and leave the system (Arazi et al. 2004; Jamalyaria et al. 2005). This applies queueing to a broad range of self-assembly systems – i.e., form an arrangement without external guidance.

Simulation: Dynamical state evolution is fundamental in both classical mathematical biology and modern systems biology. Evolution and biochemical pathways are prime examples; the underlying state-transition structure and the sheer size are sufficient to need simulation.

The kinetic laws of a biosystem depend upon the objects, particularly their scale (viz., molecules vs. cells). The deterministic rate equations have the form:

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = f_i(x;k) \qquad \text{for } i = 1,\ldots,m,$$

where x is the system state (e.g., concentrations of m metabolites) and k is a vector of parameters, called rate constants.

Sources of randomness can be intrinsic – e.g., errors in parameter estimation, or extrinsic – e.g., protein production in random pulses (Meng et al. 2004). To deal with reaction uncertainty, Gillespie (2008, 1977) introduced the probability equation:

$$\Pr(x;t+\mathrm{d}t) = \sum\nolimits_r \Pr(x-v_r;t)a_r(x-v_r)\,\mathrm{d}t + \Pr(x;t) \times \left(1 - \sum\nolimits_r a_r(x)\mathrm{d}t\right),$$

where $a_r(x)\,\mathrm{d}t$ is the probability that reaction r occurs in the time interval $(t, t+\mathrm{d}t)$, changing the state from x to $x + v_r$. The first summation represents being one reaction removed from the state x; the last term represents having no reaction during the interval.

Auto-regulatory Network – Puchalka and Kierzek (2004) consider a metabolic network with regulatory processes and random fluctuations in gene expression. Using Gillespie's equation, given the state x at time t, the probability that the next reaction, r, occurs during $(t+\tau, t+\tau+\mathrm{d}t)$ is given by:

$$\Pr(\tau, r|x,t) = a_r(x)\; e^{-\sum_j a_j(x)\tau}.$$

The simulation is run by generating (τ, r) using this joint density function. The simulation also allows for pulse production – a receptor site may be on or off to regulate gene expression (restricting the choice of r).

Other models use rare-event simulation, such as for tumor development (Abbott 2002). Simulation is used in systems biology to understand how non-dominant pathways affect assembly kinetics (Zhang and Schwartz 2006).

Game Theory: The central idea of game theory is that each player has its own objective to optimize. Historically, evolutionary biologists used game theory to model natural selection (Maynard Smith 1982; Perc and Szolnoki 2010). In OR, game theory is used to model competition for economic resources, and this extends to modeling species-invasion into an existing ecosystem. The same game model applies to the propagation of tumor cells that can mutate to create a cancer population that overwhelms normal cells (Tomlinson 1997). New applications are at the molecular scale, such as the following example.

Protein Binding – There are two sets of players: protein classes (including drugs) and DNA binding sites. Their joint strategies result in allocation of proteins to sites. Sites seek to maximize their occupancy; proteins seek to minimize excess binding. Sites compete for nearby proteins; proteins choose target sites to which they transport. (Mechanisms to achieve these choices are not well understood.) The affinity for protein i to bind to site j is denoted by the constant K_{ij}, but this applies only if the protein is in the proximity of the site.

Let $i = 1,\ldots, N_p$ index proteins and $j = 1,\ldots, N_s$ index sites, and consider the parameters:

$$\begin{aligned} v_i &= \text{nuclear concentration},\\ E_{ij} &= \text{transport affinity},\\ K_{ij} &= \text{binding affinity}. \end{aligned}$$

A protein's decision variable is its fractional transported amounts, $p^i = \left(p^i_0,\ldots,p^i_{N_s}\right)$, where $p^i_0 = 1 - \sum_{j=1}^{N_s} p^i_j$ is the portion of protein i not allocated to a site. A site's decision variable is its choice of binding frequency, $s^j = \left(s^j_0,\ldots,s^j_{N_p}\right)$, where $s^j_0 = 1 - \sum_{i=1}^{N_p} s^i_j$ is the portion of time that site j is unoccupied. There are resource constraints on joint strategies, notably $s^j_i \le p^i_j v_i$ for $i > 0$ – i.e., binding cannot exceed allocated concentration.

A solution is a joint strategy $(\bar{p}, \bar{s})$ that satisfies the optimality criteria:

$$\bar{p}^i \in \underset{p^i \in P(\bar{s})}{\operatorname{argmax}} \left\{ f_p^i(p^i, \bar{s}) \right\} \quad \bar{s}^j \in \underset{s^j \in S(\bar{p})}{\operatorname{argmin}} \left\{ f_s^j(\bar{p}, s^j) \right\},$$

where f_p, f_s denote objective functions for each protein and site, and $P \subseteq \mathbb{R}_+^{N_s+1}$, $S \subseteq \mathbb{R}_+^{N_p+1}$ denote feasible regions, each dependent on the other decisions. An example of objective functions are maximizing total binding affinity and minimizing the amount of protein not assigned:

$$f_p^i(p^i, s) = \sum_{j=1}^{N_s} E_{ij} p_j^i \left(1 - s_0^j\right)$$

$$f_s^j(s^j, p) = s_0^j \sum_{i=1}^{N_p} K_{ij} \left(p_j^i v_i - s_i^j\right).$$

With mild modifications, a solution exists and there is a simple algorithm to find it (Pérez-Breva et al. 2006).

This game model is a simplification of a broader biology, where sites can coordinate, not just compete, and proteins can form complexes to bind to the same site. There are also promoters that bind to a protein in order to send it to another site. Although current thinking is that proteins roam randomly until they bump into an unoccupied site for which they have affinity, the game model attributes a purposeful behavior to proteins, suggesting that they choose to transport to some site. While this rational behavior is not due to intelligence, it could be due to an environmental context that is not yet understood and whose net effect makes proteins behave as if they are rational players.

See

- Dynamic Programming
- Game Theory
- Integer and Combinatorial Optimization
- Linear Programming
- Markov Chains
- Markov Processes
- Network Optimization
- Nonlinear Programming
- Queueing Theory
- Simulation of Stochastic Discrete-Event Systems

References

Abbott, R. (2002). *CancerSim: A computer-based simulation of Hanahan and Weinberg's Hallmarks of Cancer*. Master's thesis, The University of New Mexico, Albuquerque, NM.

Allen, L. J. S. (2003). *An introduction to stochastic processes with applications to biology*. Upper Saddle River, NJ: Pearson Education.

Arazi, A., Ben-Jacob, E., & Yechiali, U. (2004). Bridging genetic networks and queueing theory. *Physica A: Statistical Mechanics and its Applications, 332*, 585–616.

Burkowski, F. (2009). *Structural bioinformatics: An algorithmic approach* (Mathematical and computational biology). Boca Raton, FL: Chapman & Hall/CRC.

Clote, P., & Backofen, R. (2000). *Computational molecular biology*. New York: John Wiley & Sons.

Durbin, R., Eddy, S., Krogh, A., & Mitchison, G. (1998). *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge, UK: Cambridge University Press.

Floudas, C. A., & Pardalos, P. M. (Eds.). (2000). *Local and global approaches. Optimization in computational chemistry and molecular biology*. Dordrecht: Kluwer Academic.

Forrester, R. J., & Greenberg, H. J. (2008). Quadratic binary programming models in computational biology. *Algorithmic Operations Research, 3*(2), 110–129.

Gillespie, D. T. (1977). Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry, 81*(25), 2340–2361.

Gillespie, D. T. (2008). Simulation methods in systems biology. In M. Bernardo, P. Degano, & C. Zavattaro (Eds.), *Formal methods for computational systems biology* (LNCS, Vol. 5016, pp. 125–167). Berlin: Springer.

Glodzik, A., & Skolnick, J. (1994). Flexible algorithm for direct multiple alignment of protein structures and sequences. *Bioinformatics, 10*(6), 587–596.

Goldman, D., Istrail, S., Papadimitriou, C. H. (1999). Algorithmic aspects of protein structure similarity. In *40th Annual Symposium on Foundations of Computer Science (FOCS)* (pp 512–521). IEEE Computer Society Press.

Gusfield, D. (1997). *Algorithms on strings, trees, and sequences: Computer science and computational biology*. Cambridge, UK: Cambridge University Press.

Jamalyaria, F., Rohlfs, R., & Schwartz, R. (2005). Queue-based method for efficient simulation of biological self-assembly systems. *Journal of Computational Physics, 204*(1), 100–120.

Jones, N. C., & Pevzner, P. A. (2004). *An introduction to bioinformatics algorithms*. Cambridge, MA: MIT Press.

Lancia, G. (2006). Applications to computational molecular biology. In G. Appa, P. Williams, P. Leonidas, & H. Paul (Eds.), *Handbook on modeling for discrete optimization* (International series in operations research and management science, Vol. 88, pp. 270–304). Berlin: Springer.

Maynard Smith, J. (1982). *The theory of games and the evolution of animal conflicts*. Cambridge, UK: Cambridge University Press.

Meng, T. C., Somani, S., & Dhar, P. (2004). Modeling and simulation of biological systems with stochasticity. *In Silico Biology, 4*(3), 293–309.

Palsson, B. Ø. (2006). *Systems biology: Properties of reconstructed networks*. New York: Cambridge University Press.

Perc, M., & Szolnoki, A. (2010). Coevolutionary games – a mini review. *BioSystems, 99*(2), 109–125.

Pérez-Breva, L., Ortiz, L. E., Yeang, C.-H., & Jaakkola, T. (2006). Game theoretic algorithms for protein-DNA binding. In *Proceedings of the 12th Annual Conference on Neural Information Processing (NIPS)*, Vancouver, Canada.

Puchalka, J., & Kierzek, A. M. (2004). Bridging the gap between stochastic and deterministic regimes in the kinetic simulations of the biochemical reaction networks. *Biophysical Journal, 86*(3), 1357–1372.

Steel, M., & Warnow, T. (1993). Kaikoura tree theorems: Computing the maximum agreement subtree. *Information Processing Letters, 48*(3), 77–82.

Tomlinson, I. P. M. (1997). Game-theory models of interactions between tumour cells. *European Journal of Cancer, 33*(9), 1495–1500.

Waterman, M. S. (1995). *Introduction to computational biology: Maps, sequences, and genomes (interdisciplinary statistics)*. Boca Raton, FL: Chapman & Hall/CRC.

Wedagedera, J. R., & Burroughs, N. J. (2006). T-cell activation: A queueing theory analysis at low agonist density. *Biophysical Journal, 91*, 1604–1618.

Wilkinson, D. J. (2006). *Stochastic modelling for systems biology*. Boca Raton, FL: Chapman & Hall/CRC.

Zhang, T., & Schwartz, R. (2006). Simulation study of the contribution of oligomer/oligomer binding to capsid assembly kinetics. *Biophysical Journal, 90*, 57–64.

Computational Complexity

Leslie Hall
The Johns Hopkins University, Baltimore, MD, USA

Introduction

The term computational complexity has two usages which must be distinguished. On the one hand, it refers to an algorithm for solving instances of a problem: broadly stated, the computational complexity of an algorithm is a measure of how many steps the algorithm will require in the worst case for an instance or input of a given size. The number of steps is measured as a function of that size.

The term's second, more important use is in reference to a problem itself. The theory of computational complexity involves classifying problems according to their inherent tractability or intractability — that is, whether they are easy or hard to solve. This classification scheme includes the well-known classes *P* and *NP*; the terms *NP*-complete and *NP*-hard are related to the class *NP*.

Algorithms and Complexity

To understand what is meant by the complexity of an algorithm, algorithms, problems, and problem instances must be defined. Moreover, one must understand how one measures the size of a problem instance and what constitutes a step in an algorithm. A problem is an abstract description coupled with a question requiring an answer; for example, the Traveling Salesman Problem (TSP) is: "Given a graph with nodes and edges and costs associated with the edges, what is a least-cost closed walk (or *tour*) containing each of the nodes exactly once?" An instance of a problem, on the other hand, includes an exact specification of the data: for example, "The graph contains nodes 1, 2, 3, 4, 5, and 6, and edges (1, 2) with cost 10, (1, 3) with cost 14, . . ." and so on. Stated more mathematically, a problem can be thought of as a function p that maps an instance x to an output $p(x)$ (an answer).

An algorithm for a problem is a set of instructions guaranteed to find the correct solution to any instance in a finite number of steps. In other words, for a problem p, an algorithm is a finite procedure for computing $p(x)$ for any given input x. Computer scientists model algorithms by a mathematical construct called a Turing machine, but a more concrete model will be considered here. In a simple model of a computing device, a "step" consists of one of the following operations: addition, subtraction, multiplication, finite-precision division, and comparison of two numbers. Thus if an algorithm requires one hundred additions and 220 comparisons for some instance, then the algorithm requires 320 steps on that instance. In order to make this number meaningful, it should be expressed as a function of the size of the corresponding instance, but determining the exact function would be impractical. Instead, since the main concern is how long the algorithm takes (in the worst case) asymptotically as the size of an instance gets large, one formulates a simple function of the input size that is a reasonably tight upper bound on the actual number of steps. Such a function is called the complexity or running time of the algorithm.

Technically, the *size* of an instance is the number of bits required to encode it. It is measured in terms of the inherent dimensions of the instance (such as the number of nodes and edges in a graph), plus the number of bits required to encode the numerical information in the instance (such as the edge costs). Since numerical data are encoded in binary, an integer C requires about $\log_2 |C|$ bits to encode and so contributes logarithmically to the size of the instance. The running time of the algorithm is then expressed as a function of these parameters, rather than the precise input size. For example, for the TSP, an algorithm's running time might be expressed as a function of the number of nodes, the number of edges, and the maximum number of bits required to encode any edge cost. As was seen, the complexity of an algorithm is only a rough estimate of the number of steps that will be required on an instance. In general — and particularly in analyzing the inherent tractability of a problem — an asymptotic analysis is the main interest: how does the running time grow as the size of the instance gets very large? For these reasons, it is useful to introduce Big-O notation. For two functions $f(t)$ and $g(t)$ of a nonnegative parameter t, $f(t) = O(g(t))$ if there is a constant $c > 0$ such that, for all sufficiently large t, $f(t) \leq cg(t)$. The function $cg(t)$ is thus an asymptotic upper bound on f. For example, $100(t^2 + t) = O(t^2)$, since by taking $c = 101$ the relation follows for $t \geq 100$; however, $0.0001\ t^3$ is not $O(t^2)$. Notice that it is possible for $f(t) = O(g(t))$ and $g(t) = O(f(t))$ simultaneously.

An algorithm is said to run in polynomial time (is a polynomial-time algorithm) if the running time $f(t) = O(P(t))$, where $P(t)$ is a polynomial function of the input size. Polynomial-time algorithms are generally (and formally) considered efficient, and problems for which polynomial time algorithms exist are considered easy. For the remainder of this article, the term polynomial will mean as a function of the input size.

The Classes P and NP

In order to establish a formal setting for discussing the relative tractability of problems, computer scientists first define a large class of problems called recognition (or decision) problems. This class comprises precisely those problems whose associated question requires the answer yes or no. For example, consider the problem of determining whether an undirected graph is connected (that is, whether there a path between every pair of nodes in the graph). This problem's input is a graph G consisting of nodes and edges, and its question is, "Is G connected?" Notice that most optimization problems are not recognition problems, but most have recognition counterparts. For example, a recognition version of the TSP has as input both a graph G, with costs on the edges, *and* a number K. The associated question is, "Does G contain a traveling salesman tour of length less than or equal to K?" In general, an optimization problem is not much harder to solve than its recognition counterpart. One can usually embed the recognition algorithm in a binary search over the possible objective function values to solve the optimization problem with a polynomial number of calls to the embedded algorithm.

The class P is defined as the set of recognition problems for which there exists a polynomial-time algorithm, where P stands for polynomial time. Thus, P comprises those problems that are formally considered easy. The larger problem class NP contains the class P. The term NP stands for nondeterministic polynomial and refers to a different, hypothetical model of computation, which can solve the problems in NP in polynomial time (for further explanation, see references).

The class NP consists of all recognition problems with the following property: for any yes-instance of the problem there exists a polynomial-length certificate or proof of this fact that can be verified in polynomial time. The easiest way to understand this idea is by considering the position of an omniscient being (say, the wizard Merlin) who is trying to convince a mere mortal that some instance is a yes-instance. Suppose the problem is the recognition version of the TSP, and the instance is a graph G and the number $K = 100$. Merlin knows that the instance does contain a tour with length at most 100. To convince the mortal of this fact, he simply hands her a list of the edges of this tour. This list is the certificate: it is polynomial in length, and the mortal can easily verify, in polynomial time, that the edges do in fact form a tour with length at most 100.

There is an inherent asymmetry between yes and no in the definition of NP. For example, there is no obvious, succinct way for Merlin to convince

a mortal that a particular instance does NOT contain a tour with length at most 100. In fact, by reversing the roles played by yes and no leads to a problem class known as *Co-NP*. In particular, for every recognition problem in *NP* there is an associated recognition problem in *Co-NP* obtained by framing the *NP* question in the negative (e.g., "Do *all* traveling salesman tours in *G* have length *greater* than *K*?"). Many recognition problems are believed to lie outside both of the classes *NP* and *Co-NP*, because they seem to possess no appropriate certificate. An example would be the problem consisting of a graph *G* and two numbers *K* and *L*, with the question, "Is the number of distinct traveling salesman tours in *G* with length at most *K* exactly equal to *L*?"

NP-Complete Problems

To date, no one has found a polynomial-time algorithm for the TSP. On the other hand, no one has been able to prove that no polynomial-time algorithm exists for the TSP. How, then, can one argue persuasively that the TSP and many problems in *NP* are intractable? Instead, an argument is presented that is slightly weaker but also compelling. It is shown that the recognition version of the TSP, and scores of other *NP* problems, are the *hardest* problems in the class *NP* in the following sense: if there is a polynomial-time algorithm for any one of these problems, then there is a polynomial-time algorithm for every problem in *NP*. Observe that this is a very strong statement, since *NP* includes a large number of problems (such as integer programming) that appear to be extremely difficult to solve, both in theory and in practice! Problems in *NP* with this property are called *NP*-complete. Otherwise stated, it seems highly unlikely that a polynomial algorithm will be found for any *NP*-complete problem, since such an algorithm would actually provide polynomial time algorithms for *every* problem in *NP*!

The class *NP* and the notion of complete problems for *NP* were first introduced by Cook (1971). In that paper, he demonstrated that a particular recognition problem from logic, SATISFIABILITY, was *NP*-complete, by showing directly how every other problem in *NP* could be encoded as an appropriate special case of SATISFIABILITY. Once the first *NP*-complete problem had been established, however, it became easy to show that others were *NP*-complete. To do so requires simply providing a polynomial transformation from a known *NP*-complete problem to the candidate problem. Essentially, one needs to show that the known hard problem, such as SATISFIABILITY, is a special case of the new problem. Thus, if the new problem has a polynomial-time algorithm, then the known hard problem has one as well.

Related Terms

The term *NP*-hard refers to any problem that is at least as hard as any problem in *NP*. Thus, the *NP*-complete problems are precisely the intersection of the class of *NP*-hard problems with the class *NP*. In particular, optimization problems whose recognition versions are *NP*-complete (such as the TSP) are *NP*-hard, since solving the optimization version is at least as hard as solving the recognition version.

The polynomial hierarchy refers to a vast array of problem classes both beyond *NP* and *Co-NP* and within. There is an analogous set of definitions which focuses on the space required by an algorithm rather than the time, and these time and space definitions roughly correspond in a natural way. There are complexity classes for parallel processing, based on allowing a polynomial number of processors. There are classes corresponding to randomized algorithms, those that allow certain decisions in the algorithm to be made based on the outcome of a coin toss. There are also complexity classes that capture the notions of optimization and approximability. The most famous open question concerning the polynomial hierarchy is whether the classes *P* and *NP* are the same, i.e., $P \stackrel{?}{=} NP$. If a polynomial algorithm were discovered for any *NP*-complete problem, then all of *NP* would collapse to *P*; indeed, most of the polynomial hierarchy would disappear.

In algorithmic complexity, two other terms are heard frequently: strongly polynomial and pseudo-polynomial. A strongly polynomial-time algorithm is one whose running time is bounded polynomially by a function *only* of the inherent dimensions of the problem and independent of the sizes of the numerical data. For example, most sorting algorithms are strongly polynomial, since they normally require a number of comparisons polynomial in the number of entries and do not depend on the actual values being

sorted; an algorithm for a network problem would be strongly polynomial if its running time depended only on the numbers of nodes and arcs in the network, and not on the sizes of the costs or capacities.

A pseudo-polynomial-time algorithm is one that runs in time polynomial in the dimension of the problem and the magnitudes of the data involved (provided these are given as integers), rather than the base-two logarithms of their magnitudes. Such algorithms are technically exponential functions of their input size and are therefore not considered polynomial. Indeed, some *NP*-complete and *NP*-hard problems are pseudo-polynomially solvable (sometimes these are called weakly *NP*-hard or-complete, or *NP*-complete in the ordinary sense). For example, the *NP*-hard knapsack problem can be solved by a dynamic programming algorithm requiring a number of steps polynomial in the size of the knapsack and the number of items (assuming that all data are scaled to be integers). This algorithm is exponential-time since the input sizes of the objects and knapsack are logarithmic in their magnitudes. However, as Garey and Johnson (1979) observe, "A pseudo-polynomial-time algorithm. . .will display 'exponential behavior' only when confronted with instances containing 'exponentially large' numbers, [which] might be rare for the application we are interested in. If so, this type of algorithm might serve our purposes almost as well as a polynomial time algorithm." The related term strongly *NP*-complete (or unary *NP*-complete) refers to those problems that remain *NP*-complete even if the data are encoded in unary (that is, if the data are small relative to the overall input size). Consequently, if a problem is strongly *NP*-complete then it cannot have a pseudo-polynomial-time algorithm unless $P = NP$.

For textbook introductions to the subject, see Papadimitriou (1993) and Sipser (1997). The most important reference on the subject, Garey and Johnson (1979), contains an outstanding, relatively compact introduction to complexity. Further references, including surveys and full textbooks, are given below.

See

- ▶ Combinatorics
- ▶ Graph Theory
- ▶ Integer and Combinatorial Optimization

References

Arora, S., & Barak, B. (2009). *Computational complexity: A modern approach*. Cambridge, UK: Cambridge University Press.

Bovet, D. P., & Crescenzi, P. (1994). *Introduction to the theory of complexity*. Englewood Cliffs, NJ: Prentice-Hall.

Cook, S. A. (1971). The complexity of theorem-proving procedures. *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing*, 151–158.

Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. New York: W.H. Freeman.

Karp, R. M. (1975). On the computational complexity of combinatorial problems. *Networks, 5*, 45–68.

Lewis, H. R., & Papadimitriou, C. H. (1997). *Elements of the theory of computation* (2nd ed.). Englewood Cliffs, NJ: Prentice-Hall.

Papadimitriou, C. H. (1985). Computational complexity. In E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, & D. B. Shmoys (Eds.), *The traveling salesman problem: A guided tour of combinatorial optimization*. Chichester, UK: Wiley.

Papadimitriou, C. H. (1993). *Computational complexity*. Redwood City, CA: Addison-Wesley.

Papadimitriou, C. H., & Steiglitz, K. (1982). *Combinatorial optimization: Algorithms and complexity*. Englewood Cliffs, NJ: Prentice-Hall.

Shmoys, D. B., & Tardos, E. (1989). Computational complexity of combinatorial problems. In L. Lovász, R. L. Graham, & M. Groetschel (Eds.), *Handbook of combinatorics*. Amsterdam: North-Holland.

Sipser, M. (1997). *Introduction to the theory of computation*. Belmont, CA: PWS-Kent.

Stockmeyer, L. J. (1990). Complexity theory. In E. G. Coffman Jr., J. K. Lenstra, & A. H. G. Rinnooy Kan (Eds.), *Handbooks in operations research and management science* (Computation, Chapter 8, Vol. 3). Amsterdam: North Holland.

Computational Geometry

Isabel M. Beichl[1], Javier Bernal[1], Christoph Witzgall[1] and Francis Sullivan[2]

[1]National Institute of Standards & Technology, Gaithersburg, MD, USA

[2]Supercomputing Research Center, Bowie, MD, USA

Introduction

Computational geometry is the discipline of exploring algorithms and data structures for computing geometric objects and their often extremal attributes. The objects are predominantly finite collections

of points, flats, hyperplanes, arrangements, or polyhedra, all in finite dimensions. The algorithms are typically finite, their complexity playing a central role. Emphasis is on problems in low dimensions, exploiting special properties of the plane and 3-space.

A relatively young field; its name coined in the early 1970s. It has since witnessed explosive growth, stimulated in part by the largely parallel development of computer graphics, pattern recognition, cluster analysis, and modern industry's reliance on computer-aided design (CAD) and robotics (Forrest 1971; Graham and Yao 1990; Lee and Preparata 1984). It plays a key role in the emerging fields of automated cartography and computational metrology.

The *Handbook of Discrete and Computational Geometry*, edited by Goodman and O'Rourke (1997), provides overviews of key topics. For general texts, see Preparata and Shamos (1985), O'Rourke (1987), Edelsbrunner (1987), and, de Berg et al. (2008). Pertinent concepts of discrete geometry are presented in Grünbaum (1967).

There are strong connections to operations research, whose classical problems such as finding a minimum spanning tree, a maximum-length matching, or a Steiner tree become problems in computational geometry when posed in Euclidean or related normed linear spaces. The Euclidean traveling salesman problem remains *NP*-complete (Papadimitriou 1977). Facility location, and shortest paths in the presence of obstacles, are other examples. Polyhedra and their extremal properties, typical topics of computational geometry, also lie at the foundation of linear programming. Its complexity, particularly in lower dimensions, attracted early computational geometric research, heralding the achievement of linear complexity for arbitrary fixed dimension (Megiddo 1982, 1984; Clarkson 1986).

Problems

A fundamental problem is to determine the convex hull conv(S) of a set S of n points in d-dimensional Cartesian space $\mathfrak{R}^d$. This problem has a weak and a strong formulation. Its weak formulation requires only the identification of the extreme points of conv(S). In operations research terms, that problem is well known as (the dual of) identifying redundant constraints in a system of linear inequalities. The strong formulation requires, in addition, characterization of the facets of the polytope conv(S). For dimension $d > 3$, the optimal complexity of the strong convex hull problem in $\mathfrak{R}^d$ is $O(n^{\lceil d/2 \rfloor})$ (Chazelle 1991).

Early $O(n \log n)$ methods for delineating convex hulls in the plane — vertices and edges of the convex hull of a simple polygon can be found in linear time — were based on divide-and-conquer (Graham and Yao 1983) and (Preparata and Hong 1977). In this widely used recursive strategy, a problem is divided into subproblems whose solutions, having been obtained by further subdivision, are then combined to yield the solution to the original problem. Divide-and-conquer heuristics find applications in Euclidean optimization problems such as optimum-length matching (Reingold and Supowit 1983).

The following bridge problem is, in fact, a linear program: given two sets S_1 and S_2 of planar points separated by a line, find two points $p_1 \varepsilon S_1$ and $p_2 \varepsilon S_2$ such that the line segment $[p_1, p_2]$ is an upper edge of the convex hull conv($S_1 \cup S_2$), bridging the gap between the two sets. Or, through which edge does a given directed line leave the — not yet delineated — convex hull of n points in the plane? As a linear program of fixed dimension 2, the bridge problem can be solved in linear time. Kirkpatrick and Seidel (1986) have used it along with a divide-and-conquer paradigm to devise an $O(n \log m)$ algorithm for the planar convex hull of n points, m of which are extreme.

When implementing a divide-and-conquer strategy, one typically wishes to divide a set of points $S \subset \mathfrak{R}^d$ by a straight line into two parts of essentially equal cardinality, that is, to execute a ham-sandwich cut. This can be achieved by finding the median of, say, the first coordinates of the points in S. It is a fundamental result of the theory of algorithms that the median of a finite set of numbers can be found in linear time. The bridge problem is equivalent to a double ham-sandwich cut of a planar set: given a first cut, find a second line quartering the set. Threeway cuts in three dimensions and results about higher dimensions were reported in Dobkin and Edelsbrunner (1984).

The Euclidean post office problem is a prototype for a class of proximity search problems encountered, for instance, in the implementation of expert systems. Sites p_i of n post offices in $\mathfrak{R}^d$ are given, and the task is

to provide suitable preprocessing for efficiently identifying a post office closest to any client location.

Associated with this problem is the division of space into postal regions, that is, sets of locations $V_i \subset \Re^d$ closer to postal site p_i than to any other site p_j. Each such region V_i around site p_i is a convex polyhedron, whose facets are determined by perpendicular bisectors, that is, (hyper)planes or lines of equal distance from two distinct sites. Those polyhedra form a polyhedral complex covering $\Re^d$ known as a Voronoi diagram. The Voronoi diagram and its dual, the Delaunay triangulation, are important related concepts in computational geometry.

Once a Delaunay triangulation of a planar set of n sites has been established, an $O(n \log n)$ procedure, a pair of nearest points among these sites can be found in linear time. The use of Delaunay triangulations for computational geometric problems was pioneered by Shamos and Hoey (1975).

The problem of efficiently finding a Voronoi cell V_i for an arbitrary query point p is an example of point location in subdivisions. Practical algorithms for locating a given point in a subdivision of the plane generated by n line segments in time $O(\log n)$ requiring preprocessing of order $O(n \log n)$ and storage of size $O(n \log n)$ or $O(n)$, respectively, have been proposed (Preparata 1990). For point location in planar Voronoi diagrams, Edelsbrunner and Maurer (1985) utilized acyclic graphs and packing. A probabilistic approach to the post office problem is given in Clarkson (1985).

Whether a given point lies in a certain simple polygon can be decided by an $O(n)$ process of examining the boundary intersections of an arbitrary ray emanating from the point in question. For convex polygons, an $O(n)$ preprocessing procedure permits subsequent point inclusion queries to be answered in $O(\log n)$ time (Bentley and Carruthers 1980).

An important concept with operations research implications is the medial axis of a polygon (Lee 1982), the locus of interior points with equal distance from the boundary; more precisely, those interior points with more than one closest boundary point. Such medial axes may be obtained in $O(n)$ time (Chin et al. 1995).

Let $h_e(x)$ be the truth function expressing point inclusion in the half plane to the left of a directed line segment e. Muhidinov and Nazirov (1978) have shown that a polygonal set can be characterized by a Boolean expression of n such functions, one for each edge e of the polygonal set, where each such function occurs only once in the expression. This Boolean expression transforms readily to an algebraic expression for the characteristic function of the polygon. For 3-dimensional polyhedral bodies, Dobkin, Guibas, Hershberger, and Snoeyink (1988) investigated the existence and determination of analogous constructive solid geometry (CSG) representations (they may require repeats of half space truth functions). In general, CSG representations use Boolean operations to combine primitive shapes, and are at the root of some commercial CAD/CAM and display systems. For a survey of methods for representing solid objects see Requicha (1980).

Given a family of polygons, a natural generalization of point inclusion is to ask how many of those polygons include a query point. This and similar intersection-related problems are subsumed under the term stabbing. The classical 1-dimensional stabbing problem involves n intervals. Here the stabbing number can be found in $O(\log n)$ time and $O(n)$ space after suitable preprocessing. Similar results hold for special classes of polygons such as rectangles (Edelsbrunner 1983).

Sweep-techniques rival divide-and-conquer in popularity. Plane-sweep or line-sweep, for instance, conceptually moves a vertical line from left to right over the plane, registering objects as it passes them. Plane-sweep permits one to decide in $O(n \log n)$ time (optimal complexity) whether n line segments in the plane have at least one intersection (Shamos and Hoey 1976).

Important special cases of the above intersection problem are testing for (self-)intersection of paths and polygons. Polygon simplicity can be tested for in linear time by trying to triangulate the polygon.

Polygon triangulation, more precisely, decomposing the interior of a simple polygon into triangles whose vertices are also vertices of the polygon, is a celebrated problem of computational geometry. In a seminal paper, Garey, Johnson, Preparata, and Tarjan (1978) proposed an $O(n \log n)$ algorithm for triangulating a simple polygon of n vertices. They used a plane sweep approach for decomposing the polygon into monotone polygons, which can each be triangulated in linear time. A related idea is to provide a trapezoidization of the polygon, from which a triangulation can be obtained in linear time. Chazelle (1990) introduced the concept of

a visibility map, a tree structure which might be considered a local trapezoidization of the polygon, and based on it an $O(n)$ triangulation algorithm for simple polygons. In 3-space, an analogous tetrahedralization (without additional Steiner points for vertices) for nonconvex polyhedral bodies may not exist. Moreover, the problem of deciding such existence is *NP*-complete (Ruppert and Seidel 1989).

For algorithms that depend on sequential examination of objects, bucketing or binning may improve performance by providing advantageous sequencing (Devroye 1986). The idea is to partition an area into a regular pattern of simple shapes such as rectangles to be traversed in a specified sequence. The problem at hand is then addressed locally within buckets or bins followed by adjustments between subsequent or neighboring buckets. Bucketing-based algorithms have provided practical solutions to Euclidean optimization problems, such as shortest paths, optimum-length matching, and a Euclidean version of the Chinese Postman Problem: minimizing the pen movement of a plotter (Asano et al. 1985). The techniques of quadtrees and octrees might be considered as hierarchical approaches to bucketing, and are often the methods of choice for image processing and spatial data analysis including surface representation (Samet 1990a, b).

The position of bodies and parts of bodies, relative to each other in space, determines visibility from given vantage points, shadows cast upon each other, and impediments to motion. Hidden line and hidden surface algorithms are essential in computer graphics, as are procedures for shadow generation and shading (Sutherland et al. 1974; Atherton et al. 1978). Franklin (1980) used bucketing techniques for an exact hidden surface algorithm.

Lozano-Pérez and Wesley (1979) used the concept of a visibility graph for planning collision-free paths: given a collection of mutually disjoint polyhedral objects, the node set of the above graph is the set of all vertices of those polyhedral objects, and two such nodes are connected if the two corresponding vertices are visible from each other.

The piano movers problem captures the essence of motion planning (Schwartz and Sharir 1983, 1989). Here a 2-dimensional polygonal figure, or a line segment (ladder), is to be moved, both translating and rotating, amidst polygonal barriers.

Geometric objects encountered in many areas such as Computer-Aided Design (CAD) are fundamentally nonlinear (Dobkin and Souvaine 1990). The major thrust is generation of classes of curves and surfaces with which to interpolate, approximate, or generally speaking, represent data sets and object boundaries (Barnhill 1977; Bartels et al. 1987; Farin 1988). A classical approach, building on the concepts of splines and finite elements, has been to use piecewise polynomial functions over polyhedral tilings such as triangulations. Examples are the TIN (triangulated irregular network) approach popular in terrain modeling, C^1 functions over triangulations, and the arduous solution of the corresponding C^2 problem (Heller 1990; Lawson 1977; Alfeld and Barnhill 1984).

Bézier curves and surfaces involve an elegant concept: the use of control points to define elements of curves and surfaces, permitting intuition-guided manipulation important in CAD (Forrest 1972). In general, polynomials are increasingly supplanted by rational functions, which suffer fewer oscillations per numbers of coefficients (Tiller 1983). All these techniques culminate in NURBS (non-uniform rational B-splines) which are recommended for curve and surface representation in most industrial applications.

In geometric calculations, round-off errors due to floating-point arithmetic may cause major problems (Fortune and Milenkovic 1991). When testing, for instance, whether given points are collinear, a tolerance level, eps, is often specified, below which deviations from a collinearity criterion are ignored. Points p_1, p_2, p_3 and p_2, p_3, p_4, but not p_1, p_2, p_4 may thus be found collinear. Such and similar inconsistencies may cause a computation to abort. Robust algorithms are constructed so as to avoid breakdown due to inconsistencies caused by round-off (Guibas et al. 1989; Beichl and Sullivan 1990). Alternatively, various forms of exact arithmetic are increasingly employed (Fortune and Van Wyck 1993; Yap 1993). Inconsistencies occur typically whenever an inequality criterion is satisfied as an equality. An example is the degeneracy behavior of the simplex method of linear programming. Lexicographic perturbation methods can be employed to make consistent selections of subsequent feasible bases and thus assure convergence. Similar consistent tie breaking, coupled with exact arithmetic, is the aim of the simulation of simplicity approach proposed by Edelsbrunner and Mücke (1988) in a more general computational context.

See

- Chinese Postman Problem
- Cluster Analysis
- Convex Hull
- Facility Location
- Minimum Spanning Tree Problem
- Simplex Method (Algorithm)
- Splines
- Traveling Salesman Problem
- Voronoi Constructs

References

Alfeld, P., & Barnhill, R. E. (1984). A transfinite C 2 interpolant over triangles. *Rocky Mountain Journal of Mathematics, 14*, 17–39.

Asano, T., Edahiro, M., Imai, H., & Iri, M. (1985). Practical use of bucketing techniques in computational geometry. In G. T. Toussaint (Ed.), *Computational geometry*. New York: North Holland.

Atherton, P., Weiler, K., & Greenberg, D. P. (1978). Polygon shadow generation. *Computers and Graphics, 12*, 275–281.

Barnhill, R. E. (1977). Representation and approximation of surfaces. In J. R. Rice (Ed.), *Mathematical software III*. New York: Academic Press.

Bartels, R. H., Beatty, J. C., & Barski, B. A. (1987). *An introduction to splines for use in computer graphics*. Los Altos, CA: Morgan Kaufmann.

Beichl, I., & Sullivan, F. (1990). A robust parallel triangulation and shelling algorithm. *Proceedings of 2nd Canadian Conference on Computational Geometry*, 107–111.

Bentley, J. L., & Carruthers, W. (1980). Algorithms for testing the inclusion of points in polygons. *Proceedings of 18th Allerton Conference on Communication, Control and Computing*, 11–19.

Bentley, J. L., Weide, B. W., & Yao, A. C. (1980). Optimal expected-time algorithms for closest point problems. *ACM Transactions on Mathematical Software, 6*, 563–580.

Chazelle, B. (1990). Triangulating the simple polygon in linear time. *Proceedings of 31st Annual IEEE Symposium on the Foundations of Computer Science*, 220–230.

Chazelle, B. (1991). An optimal convex hull algorithm and new results on cuttings. *Proceedings of 32nd Annual IEEE Symposium on the Foundations of Computer Science*, 29–38.

Chin, F., Snoeyink, J., & Wang, C. A. (1995). Finding the medial axis of a simple polygon in linear time. *Proceedings 6th Annual International Symposium on Algorithms and Computation*. Lecture notes in computer science (Vol. 1004, pp. 382–391). New York: Springer.

Clarkson, K. L. (1985). A probabilistic algorithm for the post office problem. *Proceedings of the 17th Annual ACM Symposium on Theory Computation*, 175–184.

Clarkson, K. L. (1986). Linear programming in$_{0(n3}$ ^{d2}) time. *Information Processing Letters, 22*, 21–24.

de Berg, M., Cheong, O., van Kreveld, M., & Overmars, M. (2008). *Computational geometry: Algorithms and applications* (3rd ed.). New York: Springer.

Devroye, L. (1986). *Lecture notes on bucket algorithms*. Boston: Birkhäuser.

Dobkin, D. P., & Edelsbrunner, H. (1984). Ham-sandwich theorems applied to intersection problems. *Proceedings of 10th International Workshop Graph-Theoretic Concepts in Computer Science (WG 84)*, 88–99.

Dobkin, D., Guibas, L., Hershberger, J., & Snoeyink, J. (1988). An efficient algorithm for finding the CSG representation of a simple polygon. *Computer Graphics, 22*, 31–40.

Dobkin, D. P., & Souvaine, D. L. (1990). Computational geometry in a curved world. *Algorithmica, 5*, 421–457.

Edelsbrunner, H. (1983). A new approach to rectangle intersections, parts I and II. *International Journal of Computer Mathematics, 13*(209–219), 221–229.

Edelsbrunner, H. (1987). *Algorithms in combinatorial geometry*. New York: Springer.

Edelsbrunner, H., & Maurer, H. A. (1985). Finding extreme points in three dimensions and solving the post-office problem in the plane. *Information Processing Letters, 21*, 39–47.

Edelsbrunner, H., & Mücke, E. P. (1988). Simulation of simplicity: a technique to cope with degenerate algorithms. *Proceedings of the 4th Annual ACM Symposium on Computational Geometry*, 118–133.

Farin, G. (1988). *Curves and surfaces for computer aided geometric design*. New York: Academic Press.

Forrest, A. R. (1971). Computational geometry. *Proceedings of the Royal Society of London Series A, 321*, 187–195.

Forrest, A. R. (1972). Interactive interpolation and approximation by bézier polynomials. *The Computer Journal, 15*, 71–79.

Fortune, S., & Milenkovic, V. (1991). Numerical stability of algorithms for line arrangements. *Proceedings of the 7th Annual ACM Symposium on Computational Geometric*, 3342–341.

Fortune, S., & Van Wyck, C. (1993). Efficient exact arithmetic for computational geometry. *ACM Symposium on Computational Geometry*, Vol. 9, 163–172.

Franklin, W. R. (1980). A linear time exact hidden surface algorithm. *Proceedings of the SIGGRAPH '80, Computer Graphics*, Vol. 14, pp. 117–123.

Garey, M. R., Johnson, D. S., Preparata, F. P., & Tarjan, R. E. (1978). Triangulating a simple polygon. *Information Processing Letters, 7*, 175–179.

Graham, R. L., & Yao, F. F. (1983). Finding the convex hull of a simple polygon. *Journal of Algorithms, 4*, 324–331.

Graham, R., & Yao, F. (1990). A whirlwind tour of computational geometry. *The American Mathematical Monthly, 97*, 687–701.

Grünbaum, B. (1967). *Convex polytopes*. New York: Wiley Interscience.

Guibas, L. J., Salesin, D., & Stolfi, J. (1989). Epsilon geometry: Building robust algorithms from imprecise computations. *Proceedings 5th Annual ACM Symposium on Computational Geometry*, 208–217.

Heller, M. (1990). Triangulation algorithms for adaptive terrain modeling. *4th Symposium on Spatial Data Handling*, 163–174.

Kirkpatrick, D. (1983). Optimal search in planar subdivisions. *SIAM Journal on Computing, 12*, 28–35.
Kirkpatrick, D. G., & Seidel, R. (1986). The ultimate planar convex hull algorithm? *SIAM Journal on Computing, 15*, 287–299.
Lawson, C. L. (1977). Software for C^1 surface interpolation. In J. R. Rice (Ed.), *Mathematical software III*. New York: Academic.
Lee, D. T. (1982). Medial axis transformation of a planar shape. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 4*, 363–369.
Lee, D. T., & Preparata, F. P. (1984). Computational geometry–A survey. *IEEE Transactions on Computers, c-33*, 1072–1101.
Lozano-Pérez, T., & Wesley, M. A. (1979). An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM, 22*, 560–570.
Megiddo, N. (1982). Linear-time algorithms for linear programming in R^3 and related problems. *Proceedings of the 23rd Annual IEEE Symposium on the Foundations of Computer Science*, 329–338.
Megiddo, N. (1984). Linear programming in linear time when the dimension is fixed. *Journal of the ACM, 31*, 114–127.
Muhidinov, N., & Nazirov, S. (1978). Computerized recognition of closed plane domains. *Voprosy Vychislitel'noj i Prikladnoj Matematiki (Tashkent)*, *53*, 96–107, 182.
O'Rourke, J. (1987). *Art gallery theorems and algorithms*. New York: Oxford University Press.
Papadimitriou, C. H. (1977). The euclidean traveling salesman problem is *NP*-complete. *Theoretical Computer Science, 4*, 237–244.
Preparata, F. P. (1990). Planar point location revisited. *International Journal of Foundations of Computer Science, 24*(1), 71–86.
Preparata, F. P., & Hong, S. J. (1977). Convex hulls of finite sets of points in two and three dimensions. *Communications of the ACM, 20*, 87–93.
Preparata, F. P., & Shamos, M. I. (1985). *Computational geometry: An introduction*. New York: Springer.
Reingold, E. M., & Supowit, K. J. (1983). Probabilistic analysis of divide-and-conquer heuristics for minimum weighted euclidean matching. *Networks, 13*, 49–66.
Requicha, A. A. G. (1980). Representations for rigid solids: Theory, methods, and systems. *ACM Computing Surveys, 12*, 437–464.
Ruppert, J., & Seidel, R. (1989). On the difficulty of tetrahedralizing 3-dimensional non-convex polyhedra. *Proceedings of the 5-th Annual ACM Symposium on Computational Geometry*, 380–392.
Samet, H. (1990a). *The design and analysis of spatial data structures*. Reading, PA: Addison Wesley.
Samet, H. (1990b). *Applications of spatial data structures: Computer graphics, image processing and GIS*. Reading, PA: Addison Wesley.
Schwartz, J. T., & Sharir, M. (1983). On the 'piano movers' problem, I: The case of a two-dimensional rigid polygonal body moving amidst polygonal barriers. *Communications on Pure and Applied Mathematics, 36*, 345–398.
Schwartz, J. T., & Sharir, M. (1989). A survey of motion planning and related geometric algorithms. In D. Kapur & J. Mundy (Eds.), *Geometric reasoning* (pp. 157–169). Cambridge, MA: MIT Press.
Shamos, M. I., & Hoey, D. (1975). Closest-point problems. *Proceedings of the 16th Annual IEEE Symposium on the Foundations of Computer Science*, 151–162.
Shamos, M. I., & Hoey, D. (1976). Geometric intersection problems. *Proceedings of the 17th Annual IEEE Symposium on the Foundations of Computer Science*, 208–215.
Sutherland, I. E., Sproull, R. F., & Shumacker, R. A. (1974). A characterization of ten hidden surface algorithms. *ACM Computing Surveys, 6*, 1–55.
Tiller, W. (1983). Rational B–splines for curve and surface representation. *IEEE Computer Graphics and Applications, 3*(6), 61–69.
Yap, C. (1993). Towards exact geometric computation. *Proceedings of the 5th Canadian Conference on Computational Geometry*, 405–419.

Computational Intelligence

► Artificial Intelligence

Computational Organization Theory

Terrill L. Frantz[1], Kathleen M. Carley[2] and William A. Wallace[3]
[1]Peking University, Shenzhen, Guangdong, China
[2]Carnegie Mellon University, Pittsburgh, PA, USA
[3]Rensselaer Polytechnic Institute, Troy, NY, USA

Introduction

As inexpensive and massive amounts of computing power have rapidly become more widely available, the operational aspects of computational-based organizational research have become a reality. Today, the concepts of Computational Organization Theory (COT) can be easily implemented and practiced by an ever-increasingly larger group of researchers. Some foresee such computer-science related computational thinking (Wing 2006), as the future of all scholarly research, and COT is part of this broader trend.

COT involves the theorizing about, describing, understanding, and predicting the behavior of organizations and the process of organizing, using

quantitative-based and structured approaches (computational, mathematical and logical models). This involves computational abstractions that are incorporated into organizational research and practice through COT tools, procedures, measures and knowledge.

The notion of an organization, as used here, spans the wide range of human-conceived collections of people, i.e., groups, teams, societies, corporations, industries, and governments, see Carley and Prietula, (1994); Prietula, Carley, and Gasser, (1998); and Gilbert and Doran, (1994). COT practitioners use computational models and analysis to develop a better understanding of fundamental principles for organizing and behaviors within an organization. Organizational members, i.e., people, are considered information-processing actors. They can interact with and adapt to their environment. They can learn, and they can communicate. While their behavior is certainly complex, this behavior and the underlying determinate of the behaviors can be reduced to basic mathematical equations and algorithms. With this formalization, researchers can develop complete computerized models of an organization, which enables the use of computer simulation to create virtual worlds for non-obtrusive experimentation. After running these simulations the collective outcome of these virtual interactions and behaviors can be quantified and collected for extensive analysis. Typically, the results from these experiments are then incorporated into a formalized and thoughtful comparison against findings from controlled lab experiments and real-world empirical cases studies. The history of COT is rich with academic insight, with its research and application proving fruitful to organization researchers and practitioners alike.

History

The field of COT has benefitted from several decades of research. One of the earliest works is Cyert and March's *The Behavioral Theory of the Firm*, (1963), in which a simple information-processing model of an organization is used to address issues of organization design and performance. During the past decade an explosion of interest has occurred for theory development and testing in the organizational and social sciences (Carley 1995). The use is expanding for a number of reasons: (a) there is growing recognition that social and organizational processes are complex, dynamic, adaptive, and nonlinear, and, thus, are hard to study in the real-world; (b) researchers and practitioners have come to realize that organizational and social behavior emerges from interactions within and between ecologies of entities (people, groups, technologies, agents, etc.), which is hard to reproduce and control in the laboratory and real-world; and (c) the relationships among these entities are critical constraints on individual and organizational action, which is hard to control with direct human-based research. Researchers now recognize that organizations are inherently computational since they have a need to scan and observe their environment, store facts and programs, communicate among members and with their environment, and transform information by human or automated decision making (Burton and Obel 1996).

COT has a fundamentally interdisciplinary intellectual history with contributions from social network theory, distributed artificial intelligence and the organizational information processing tradition. Within COT, researchers draw heavily on work in the information/resource processing tradition (Simon 1947; March and Simon 1958; Thompson 1967; Galbraith 1973; Cyert and March 1963; Pfeffer and Salancik 1978) and social information processing (Salancik and Pfeffer 1978), as modified by work in cognitive science (Carley and Newell 1994), institutionalism (Powell and DiMaggio 1991), population ecology (Hannan and Freeman 1977, 1989), and the contemporary contingency theory (Baligh et al. 1990). Within social network and communication/coordination theory, there has been important work done on measures of organizational design and communication (Wasserman and Faust 1994; Malone 1986), cognitive social structures (Krackhardt 1987), network effects on performance, influence, and power (Wasserman and Galaskiewicz 1994; Kaufer and Carley 1993; Granovetter 1985; Burt 1992), and research on inter-organizational networks (Baum and Oliver 1991; Stuart and Podolny 1996). Within the area of distributed artificial intelligence, researchers draw on findings regarding representation (Lesser and Corkill 1988); teams (Decker 1996); coordination (Durfee and Montgomery 1991); and strategy (Gasser and Majchrzak 1994).

Methodological Approaches

Models are both integral and integrating components of theory building in COT. No matter what their disciplinary home, researchers in this area assume that meaningful and predictive models of organizations can be built. Computational organizational theorists use models to (1) describe organizational phenomena observed in the world, including structuring real or hypothetical experiences as described or postulated by individuals or groups, (2) formalize and integrate theoretical principles from science that are relevant to organizational activities, and (3) simulate the dynamics of temporal changes in a particular organizational process, action, or policy.

Models are abstractions of reality, and modeling is the process of creating these abstractions. Because reality is near infinitely complex and all empirical data are processed with reference to that complexity, model building involves the simplification of reality as data are transformed into knowledge. The models created are, essentially, forms of codified knowledge and used to represent the reality of things not known from things that are known (Waisel et al. 1998). Modeling is the sine qua non of science. Virtually all-scientific activities require modeling in some sense, and any scientific theory requires this kind of representational system (Nersessian 1992). Models are usually thought of as being quantitative, and able to be represented mathematically. However, qualitative models are no less and arguably more common, particularly in the context of COT.

Employing a variety of methodologies has made advances in computational organization theory. To illustrate this variety, five of the most significant approaches to modeling will be discussed: (1) general intellective models, (2) distributed artificial intelligence and multi-agent models, (3) organizational engineering models, (4) social network models, and (5) mathematical and/or logic based models.

Organizational theorists are most familiar with the general intellective models. These models often represent the organization or various processes as a set of nonlinear equations and/or a set of interacting agents. In these models, the focus is on explaining and theorizing about a particular aspect of organizational behavior. Consequently, the models often abstract many of the factors in actual organizations, laying bear only the entities and relations essential to the theory. Models embody theory about how the team, group, or organization will behave. Given these models, a series of virtual experiments are run to test the effect of a change in a particular process, action, policy, etc. These models are used to illustrate the theory's story about how the organization will behave under various conditions. These models enable cumulative theory building as multiple researchers rebuild, augment, and develop variations of earlier models.

Many researchers are building organizational models using multi-agent techniques. Multi-agent techniques have grown out of the work in distributed artificial intelligence. Distributed artificial intelligence intended to perform highly specific but stylized tasks such as soccer, navigation or surveillance (Bond and Gasser 1988; Gasser and Huhns 1989; Cohen 1986). Strength of this approach is the focus on representation and knowledge. For example attention is often focused on how to represent the task and knowledge about how to do the task via the agent. Another strength of this approach is a focus on decision making as search. Models are often developed to address issues of communication, coordination, planning, or problem solving, often with the intent of using these models as the brains in artificial agents. These models can explain many organizational phenomena and test the adequacy and efficiency of various definitions or representation schemes. Today, much of this work goes under the rubric of multi-agent modeling. Work in this area is beginning to focus on the role of emotions, the development of team mental models, and coordination of large numbers of agents. From an organizational theory perspective two issues stand out. First, how scalable are these models and representation schemes? That is, do the results from systems of two to five agents performing a highly stylized task generalize to larger more complex organizations? Second, when are these cognitively simple agents adequate or valid representations of human behavior?

Organizational engineering models are characterized by the extensive detail with which they represent the formal sides of organizations or tasks (organizational chart, workflow, communication paths, and rework routines) and the attention to the specific features of particular organization. These models generally focus on predicting overall

organizational or group response rather than the actions and behaviors of individual agents. These models are sufficiently detailed that they can be used to analyze potential policy changes and address what-if questions for the particular organization for which the model has been tuned (Levitt et al. 1994; Gasser and Majchrzak 1994). Model adequacy is often demonstrated by determining whether the parameters can be adjusted so that one or more important team or organizational behaviors is described at least at a qualitative level. Importantly, simply having managers work with the research team to elicit the data on the organization needed to model it often leads the manager to gain important insights into organizational problems. As such, these models are a valuable decision aid. The same is true of system models.

Social network models are characterized by representations of teams, groups, organizations, and markets in terms of the relationships among individuals or organizations. These models emphasize the structural or relational aspect of the organization and demonstrate when and how they can affect individual or organizational behaviors. Work in this field has focused on developing models of network adaptation, evolution, and change, and on developing a better understanding of how agent knowledge affects and is affected by an agent's position in the network. Network models have successfully been used to examine issues such as power and performance, information diffusion, innovation, and turnover. The adequacy of these models is determined using techniques from non-parametric statistics.

Logic models are characterized by representations of organizations and organizational processes using the techniques and formalisms of formal logic. Such models enable researchers to focus on the generative aspects of organizational form given a specific grammar (See Salanick and Leblebici 1998) and to test the consistency of extant verbal theories. These models tend to be among the most limited in their realism and the least likely to capture dynamic aspects of organizational behavior. However, these models are the only ones from which complete proofs and an exhaustive understanding of behavior can be generated. These models provide, independent of a specific machine implementation, a way of assessing the internal validity of extant theories and generating proofs about organizing behavior.

This brief review of these methodological approaches just begins to describe the vast array of modeling techniques and tools that have been used to examine organizations. These and other approaches address a variety of questions about organizations ranging from questions of design, to questions of learning, to questions of culture. As work continues in this field, researchers are beginning to employ models, which contain intellective and emulative elements. These models, for example, draw on the work in cognitive science and contribute to the work on multi-agent systems, use network representations and measures, and use logic in developing formalizations.

Models and Applications

COT models extend from simple intellective principles of general decision-making behavior (Cohen et al. 1972; Carley 1992) to representations of the decision processes and information flow within specific real-world organizations (Levitt et al. 1994; Zweben and Fox 1994). Models may even operationally specific management-decisions, or practices and polices (Gasser and Majchrzak 1992, 1994; Majchrzak and Gasser 1991, 1992). These COT models enable the researcher to examine the potential impact of general management strategies (Gasser and Majchrzak 1994; Carley and Svoboda 1996), or enable the manager to examine the organizational implications of specific management decisions (Levitt et al. 1994).

Several multipurpose computational-models of organization have been developed including well-known models such as the Garbage Can Model (Cohen et al. 1972), Plural-Soar (Carley et al. 1992), Team-Soar (Kang et al. 1998), and ORGAHEAD (Lee and Carley 2004). In a review of the state of computational modeling (Ashworth and Carley 2004, 2007), 29 specific organization theory computer simulations were found to have been introduced between 1989 and 2003; the authors also made a point that the richness of the models has also increased over those years. More recently, the CONSTRUCT model has been used extensively for theory generation and testing, notably in realms looking at the impact of communications occurring through diverse mEdia. CONSTRUCT provides a vigorous model of organization that has its

roots in symbolic interactionism (Blumer 1969), structural interactionism (Stryker 1980), and structural differentiation theory (Blau 1970). These core-theories are combined into a computational theory called constructuralism (Carley 1991), which is embodied in the CONSTRUCT model. The model recognizes that people interact within a dynamic social-based organizational network and are characteristically information-seeking agents. They interact to exchange information and purposefully may seek out others who have information that they do not yet hold. Others seeking their information, or knowledge are also seeking them out. This interaction dynamic is played out innumerable times in any organization. When this dynamic is coupled with the organization-membership changes (hiring and firing) in an organization, this emerging micro-interaction dynamic is manifested in complex organization-level dynamics and outcomes.

Computational organizational theorists often address issues of organizational design, organizational learning, and organizational adaptation. Consider the design question: organizations, through their design, are expected to be able to overcome the cognitive, physical, temporal, and institutional limitations of individual agency. Research has shown that there is no single organizational design that yields the optimal performance under all conditions, yet it has shown, that for a particular task and under particular conditions, there is a set of optimal designs. Organizational performance itself is dynamic, even under the same design (Cohen 1986). Thus, the determination of which organizational design is best depends on a plethora of factors, which interact in complex nonlinear ways to effect performance. Such factors include the task(s) being performed; intelligence, cognitive capabilities, skills, or training; available resources; quality and quantity of information; volatility of the environment; legal or political constraints on organizational design; the type of outcome desired (e.g., efficiency, effectiveness, accuracy, or minimal costs). The organization's design is considered to be capable of being intentionally changed in order to improve its performance. Consequently, computational models focused on design should be an invaluable decision aid to managers who are interested in comparing and contrasting different types of organizations. Researchers are thus providing guidelines for when to use which design, and developing computational tools for enabling managers to do just-in-time design.

Organizational learning, adaptation and change are one of the areas where COT continues to provide invaluable knowledge and understandable promise. In most organizations, multiple types of learning appear to co-exist and interact in complex ways. Organizational learning has been characterized in terms of the search for knowledge (Levinthal and March 1981), constraint based optimization (Carley and Svoboda 1996), and aggregation of individual learning (Carley 1992). In organizational learning, one major challenge is to link multiple models of organizational learning together and to see how they inform each other. It is necessary to understand how organizational networks evolve and how an evolved organizational design can be characterized as being statistically different from an initial design. Such issues of measurement are subjects of continued research within the field of COT.

Research Opportunities

The focus of COT is evolving. Past research has focused on representations of natural or human organizations. Increasingly, researchers are using COT methods to study organizations that are also composed of artificial agents, or combinations of both human and artificial agents. Human organizations, and artificial systems in general, often show intelligence and a set of capabilities that are distinct from the intelligence and capabilities of the membership within them. These systems can exhibit organization, intentional adaptation, and can display non-random and repeated patterns and processes of action, communication, knowledge, and memory regardless of whether or not the agents are human. By improving our understanding of the behavior of artificial worlds in general, researchers may discover whether there are general principles of organizing that transcend the type of agent in the organization. Artificial or virtual organizations are appearing and being used to do certain tasks such as scheduling or robotic control. One of the issues is how to structure inter-agent coordination and communications. Should organizations of humans and artificial agents be designed in the same way? Do artificial agents need to communicate the same type of information, as do

humans to be effective? Modeling the interactivity of humans and artificial agents should enable us to answer these questions.

COT will move theories of organizations beyond empirical description to predictive modeling. By focusing on the components (such as agent, structure, task, and resources), the networks of connections among these components (such as the communication structure or the resource access structure), and the processes by which they are altered (such as routines, learning, adaptation), a more dynamic and coherent view of the organization as an embedded, complex, adaptive system of human and automated agents with greater predictive ability will emerge (Carley and Prietula 1994). Attending to these factors will necessarily increase the complexity and veridicality of the models, as well as increasing the difficulty in building and validating the models. The resulting models, however, will be capable of addressing the concerns of both the theoretician and the practitioner, and yield greater predictive ability and practical guidance. COT thus has the potential to generate a better theoretical understanding of organizations, better tools for designing and reengineering organizations in real-time, and better tools for teaching people how teams, groups, and organizations function.

See

► Organization

References

Ashworth, M., & Carley, K. M. (2004). Toward unified organization theory: Perspectives on the state of computational modeling. *Proceedings of the NAACSOS 2004 Conference*, Pittsburgh, PA.

Ashworth, M., & Carley, K. M. (2007). Can tools help unify organization theory? Perspectives on the state of computational modeling. *Computational and Mathematical Organization Theory, 13*(1), 89–111.

Baligh, H. H., Burton, R. M., & Obel, B. (1990). Devising expert systems in organization theory: The organizational consultant. In M. Masuch (Ed.), *Organization, management, and expert systems*. Berlin: Walter De Gruyer.

Baum, J., & Oliver, C. (1991). Institutional linkages and organizational mortality. *Administrative Science Quarterly, 36*, 187–218.

Blau, P. M. (1970). A formal theory of differentiation in organizations. *American Sociological Review, 35*(2), 201–218.

Blumer, H. (1969). *Symbolic interactionism: Perspective and method*. Englewood Cliffs, NJ: Prentice-Hall.

Bond, A., & Gasser, L. (Eds.). (1988). *Readings in distributed artificial intelligence*. San Mateo, CA: Kaufmann.

Burt, R. (1992). *Structural holes: The social structure of competition*. Boston: Harvard University Press.

Burton, R. M., & Obel, B. (1996). Organization. In S. I. Gass & C. M. Harris (Eds.), *Encyclopedia of operations research and management science*. Norwood, MA: Kluwer Academic Publishers.

Carley, K. M. (1991). A theory of group stability. *American Sociological Review, 56*(3), 331–354.

Carley, K. M. (1992). Organizational learning and personnel turnover. *Organization Science, 3*(1), 20–46.

Carley, K. M. (1995). Computational and mathematical organization theory: Perspective and directions. *Computational and Mathematical Organization Theory, 1*(1), 39–56.

Carley, K. M., Kjaer-Hansen, J., Prietula, M., & Newell, A. (1992). Plural-soar: A prolegomenon to artificial agents and organizational behavior. In M. Masuch & M. Warglien (Eds.), *Distributed intelligence: Applications in human organizations* (pp. 87–118). Amsterdam: Elsevier Science.

Carley, K. M., & Newell, A. (1994). The nature of the social agent. *Journal of Mathematical Sociology, 19*(4), 221–262.

Carley, K. M., & Prietula, M. J. (Eds.). (1994). *Computational organization theory*. Hillsdale, IN: Lawrence Erlbaum Associates.

Carley, K. M., & Svoboda, D. M. (1996). Modeling organizational adaptation as a simulated annealing process. *Sociological Methods and Research, 25*, 138–168.

Cohen, M. D. (1986). Artificial intelligence and the dynamic performance of organizational designs. In J. G. March & R. Weissinger-Baylon (Eds.), *Ambiguity and command: Organizational perspectives on military decision making* (pp. 53–70). Marshfield, MA: Pitman.

Cohen, M. D., March, J. G., & Olsen, J. P. (1972). A garbage can model of organizational choice. *Administrative Science Quarterly, 17*, 1–25.

Cyert, R., & March, J. G. (1963). *A behavioral theory of the firm*. Englewood Cliffs, NJ: Prentice-Hall.

Decker, K. (1996). TAEMS: A framework for environment centered analysis and design of coordination mechanisms. In G. M. P. O'Hare & N. R. Jennings (Eds.), *Foundations of distributed artificial intelligence*. New York: John Wiley.

Durfee, E. H., & Montgomery, T. A. (1991). Coordination as distributed search in a hierarchical behavior space. *IEEE Transactions on Systems, Man, and Cybernetics, 21*, 1363–1378.

Galbraith, J. (1973). *Designing complex organizations*. Reading, MA: Addison-Wesley.

Gasser, L., & Huhns, M. N. (Eds.). (1989). *Distributed artificial intelligence* (Vol. 2). New York: Morgan Kaufmann.

Gasser, L., & Majchrzak, A. (1992). HITOP-A: Coordination, infrastructure, and enterprise integration. *Proceedings of the First International Conference on Enterprise Integration* (pp. 373–378). Hilton Head, SC: MIT Press.

Gasser, L., & Majchrzak, A. (1994). ACTION integrates manufacturing strategy, design, and planning. In P. Kidd & W. Karwowski (Eds.), *Ergonomics of hybrid automated systems IV* (pp. 133–136). Amsterdam: IOS Press.

Gilbert, N., & Doran, J. (Eds.). (1994). *Simulating societies: The computer simulation of social phenomena*. London: UCL Press.
Granovetter, M. (1985). Economic action and social structure: The problem of embeddedness. *The American Journal of Sociology, 91*, 481–510.
Hannan, M. T., & Freeman, J. (1977). The population ecology of organizations. *The American Journal of Sociology, 82*, 929–964.
Hannan, M. T., & Freeman, J. (1989). *Organizational ecology*. Cambridge, MA: Harvard University Press.
Kang, M., Waisel, L. B., & Wallace, W. A. (1998). Team-soar: A model for team decision making. In M. Prietula, K. Carley, & L. Glasser (Eds.), *Simulating organizations: Computational models of institutions and groups* (pp. 23–45). Menlo Park, CA: AAAI Press/The MIT Press.
Kaufer, D. S., & Carley, K. M. (1993). *Communication at a distance: The effect of print on socio-cultural organization and change*. Hillsdale, IN: Lawrence Erlbaum Associates.
Krackhardt, D. (1987). Cognitive social structures. *Social Networks, 9*, 109–134.
Lee, J-S., & Carley, K. M. (2004). *OrgAhead: A computational model of organizational learning and decision making [Version 2.1.5]* (Technical Report CMU-ISRI-04-117), Carnegie Mellon University, School of Computer Science, Institute for Software Research International.
Lesser, D. D., & Corkill, D. D. (1988). Functionally accurate, cooperative distributed systems. In A. H. Bond & L. Gasser (Eds.), *Readings in distributed artificial intelligence*. San Mateo, CA: Morgan Kaufmann.
Levinthal, D., & March, J. G. (1981). A model of adaptive organizational search. *Journal of Economic Behavior and Organization, 2*, 307–333.
Levitt, R. E., Cohen, G. P., Kunz, J. C., Nass, C. I., Christiansen, T., & Jin, Y. (1994). The Virtual Design Team: Simulating how organization structure and information processing tools affect team performance. In K. M. Carley & M. J. Prietula (Eds.), *Computational organization theory* (pp. 1–18). Hillsdale, IN: Erlbaum.
Majchrzak, A., & Gasser, L. (1991). On using artificial intelligence to integrate the design of organizational and process change in US manufacturing. *Artificial Intelligence and Society, 5*, 321–338.
Majchrzak, A., & Gasser, L. (1992). HITOP-A: A tool to facilitate interdisciplinary manufacturing systems design. *International Journal of Human Factors in Manufacturing, 2*(3), 255–276.
Malone, T. W. (1986). Modeling coordination in organizations and markets. *Management Science, 33*, 1317–1332.
March, J., & Simon, H. (1958). *Organizations*. New York: John Wiley.
Nersessian, N. J. (1992). How do scientists think? Capturing the dynamics of conceptual change in science. In R. N. Giere (Ed.), *Cognitive models of science* (Vol. 15). Minneapolis, MN: Minnesota Press.
Pfeffer, J., & Salancik, G. R. (1978). *The external control of organizations: A resource dependence perspective*. New York: Harper and Row.
Powell, W. W., & DiMaggio, P. J. (1991). *The new institutionalism in organizational analysis*. Chicago: University of Chicago Press.
Prietula, M. J., Carley, K. M., & Gasser, L. (Eds.). (1998). *Simulating organizations: Computational models of institutions and groups*. Menlo Park, CA: AAAI Press/The MIT Press.
Salancik, G. R., & Pfeffer, J. (1978). A social information professing approach to job attitudes and task design. *Administrative Science Quarterly, 23*, 224–253.
Salanick, G. R., & Leblebici, H. (1998). Variety and form in organizing transactions: A generative grammar of organization. *Research in the Sociology of Organizations, 6*, 1–31.
Simon, H. A. (1947). *Administrative behavior*. New York: Free Press.
Stryker, S. (1980). *Symbolic interactionism: A social structure version*. Menlo Park, CA: Benjamin/Cummings Publishing.
Stuart, T. E., & Podolny, J. M. (1996). Local search and the evolution of technological capabilities. *Strategic Management Journal, 17*, 21–38.
Thompson, J. D. (1967). *Organizations in action*. New York: McGraw-Hill.
Waisel, L., Wallace, W. A., & Willemain, T. (1998). Using diagrammatic reasoning in mathematical modeling: The sketches of expert modelers. *Proceedings of the AAAI 1997 Fall Symposium on Reasoning with Diagrammatic Representations II*. Menlo Park, CA: AAAI Press.
Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications*. New York: Cambridge University Press.
Wasserman, S., & Galaskiewicz, J. (Eds.). (1994). *Advances in social network analysis: Research in the social and behavioral sciences*. Thousand Oaks, CA: Sage.
Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35.
Zhiang, L., & Carley, K. (1995). DYCORP: A computational framework for examining organizational performance under dynamic conditions. *Journal of Mathematical Sociology, 20* (2–3), 193–218.
Zweben, M., & Fox, M. S. (Eds.). (1994). *Intelligent scheduling*. San Mateo, CA: Morgan Kaufmann.

Computational Probability

Broadly defined, computational probability is the computer-based analysis of stochastic models with a special focus on algorithmic development and computational efficacy. The computer and information revolution has made it easy for stochastic modelers to build more realistic models even if they are large and seemingly complex. Computational probability is not just concerned with questions raised by the numerical computation of existing analytic solutions and the exploitation of standard probabilistic properties. It is the additional concern of the probabilist, however, to ensure that the solutions

obtained are in the best and most natural form for numerical computation. Before the advent of modern computing, much effort was directed at obtaining insight into the behavior of formal models, while avoiding computation. On the other hand, the early difficulty of computation has allowed the development of a large number of formal solutions from which limited qualitative conclusions may be drawn, and whose appropriateness for algorithmic implementation has not been seriously considered. Ease of computation has now made it feasible to have the best of all worlds: computation is now possible for classical models heretofore not completely solved, while complex algorithms can be developed for providing often needed insights on stochastic behavior.

See

- ▶ Applied Probability
- ▶ Computer Science and Operations Research Interfaces
- ▶ Matrix-Analytic Stochastic Models
- ▶ Phase-Type Probability Distributions
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Drew, J., Evans, D., Glen, A., & Leemis, L. (2008). *Computational probability: Algorithms and applications in the mathematical sciences*. New York: Springer.

Computer Science and Operations Research Interfaces

John W. Chinneck[1] and Ramesh Sharda[2]
[1]Carleton University, Ottawa, Ontario, Canada
[2]Oklahoma State University, Stillwater, OK, USA

Introduction

Operations research (OR) and computer science (CS) grew up together. George Dantzig relates how his early work on linear programming (LP), the archetypal OR method, led to the funding of the development of the first electronic computers during the 1940s (Dantzig 1988, 2002). In the early years of commercial computing, a large fraction of all computing effort was devoted to linear programming. Rapid development of applications for linear programming and then for the many other OR methods followed quickly thereafter, and such developments still continue. From that same starting point, CS and computer engineering developed on parallel tracks, with the disciplines continuing to interact while developing their own separate traditions and foci of study. There has been a renewed exploration of the many areas of overlap, with the development of much improved hybrid methods for solving difficult problems.

Evidence of ongoing interest in the OR/CS interface is easy to find. There are a number of academic journals devoted to the interface, including *The INFORMS Journal on Computing, Computers and Operations Research, Computers and Industrial Engineering, Computational Optimization and Applications*, and *Mathematical Programming Computation*. The INFORMS Computing Society, a subgroup of INFORMS, the largest OR professional group, is devoted to the study of the interfaces between OR and CS.

OR can be viewed as a collection of methodologies for solving common problems related to operating organizations and designing systems. Computers are essential in using these techniques to solve problems of industrial scale. Computers carry out the numerous calculations involved in most OR methods and provide database functions to manage the very large volumes of data that are input and output. There are several important interfaces between OR and the discipline of CS; some of the main interfaces are reviewed below.

Computer Hardware

The essential interface of OR with CS and computer engineering is the computer itself. Computer hardware has seen numerous changes since the 1940s: mainframes, supercomputers, inexpensive personal computers, with additional major changes that include grids, clouds, and inexpensive multi-core machines. This has affected OR in terms of the methods used and the scale of the problems solved. OR methods have been adapted to solve extremely challenging problems of

very large scale by taking advantage of inexpensive and massively parallel computer architectures. Capabilities such as pipelining, vectorization, and superscalar computations have been employed in implementations of the simplex method, as well as interior point methods for LP. Algorithms have also been developed to exploit multiple as well as massively parallel processors: see the summaries by Zenios (1989) and Eckstein (1993). Thain et al. (2005) address distributed computing, which makes use of massively parallel computing resources that are heterogeneous and physically distributed, and subject to interruption by other uses that have higher priority. The CONDOR software used for this purpose is primarily directed toward high-throughput computing. As described in a special cluster of papers in the *INFORMS Journal on Computing* (*Volume 21, Issue 3, 2009*) on high-throughput optimization, the CONDOR software is a major enabler of large-scale optimization because it facilitates flexible access to a large pool of computers. A second major theme in the special cluster is the parallelization of tree search of various kinds.

Computers themselves, especially computer chips composed of Very Large Scale Integrated (VLSI) circuits, are extremely complex to design. There are many difficult optimization problems to solve during the design process. Some examples: What is the best way to arrange the devices on the chip to pack the maximum number of devices into the smallest area? How should the connecting wires be routed to minimize the total length of wiring? Which technologies should be employed for each of the devices? Here, not surprisingly, OR optimization techniques find many applications. The OR techniques of queueing analysis and simulation are also widely used to investigate the behavior of the chips prior to their production and the behavior of the entire computer system. For example, buffering delays related to queueing for memory or CPU access can be estimated. The survey by Chinneck et al. (2005) describes the many applications of OR in Computer-Aided Design (CAD) of VLSI chips.

Other useful developments in computer hardware that generally contribute to speedier computations also improve the speed of OR-related computations. These developments include cache memory and superscalar computation. Since the LP matrix computations involve working with sparse matrices, use of cache memory allows faster access and manipulation of matrix elements. Similarly, superscalar architectures, as well as vectorization facilities of the new computer, allow vectored calculations. LP codes such as Gurobi and IBM's CPLEX are examples of codes that have exploited the recent developments in computer architecture quite well.

OR has benefitted greatly from advances in computer design that originate in CS and computer engineering. Larger and more complex problems can now be solved. At the same time, the advances in computer hardware would likely not have been possible without the use of OR techniques in generating the designs. The fields are mutually reinforcing.

Software: Algorithms

Perhaps the widest area of overlap between OR with CS is software, particularly algorithms. While CS has a general interest in all algorithms, OR constitutes a particularly important subset of algorithms that have immediate practical applications. Interestingly, the two disciplines have often approached problems of mutual interest in completely different ways. This is particularly apparent in the field of combinatorial optimization where OR has traditionally taken a more mathematical approach while CS has taken a purely algorithmic approach as in constraint programming. The two approaches have begun to merge into a stronger hybrid. For example, concepts from constraint programming have been incorporated into branch-and-bound-based implementations of mixed-integer linear programming solvers. Hooker (2007) presents an excellent exposition of this theme.

The OR repertoire has been considerably expanded through the adoption of optimization techniques that arise from the CS algorithmic tradition, instead of the traditional OR mathematical tradition. Many of these are interesting heuristics that may not provide solution guarantees, but which can be effective in practice for certain classes of very difficult problems. Examples include Genetic Algorithms (Goldberg 1989), ant colonies, particle swarms, and other evolutionary algorithms. Partly as a consequence of exposure to these CS-originated methods, OR now develops CS-flavored methods, for example, scatter search and path relinking (Glover et al. 2000). Powell (2010) merges AI and OR to solve high-dimensional

stochastic optimization problems. Both OR and CS are keenly interested in artificial intelligence, knowledge and data management, and machine learning. Here the mixing of the two traditions is common. For example, an AI technique for robot route planning may use the solution of a number of standard OR problems as steps in a larger planning algorithm.

Software: User Interfaces

Both OR and CS face issues related to user interaction with complex objects. In OR, the objects are typically mathematical models, which may consist of functions representing the objective and the constraints in an optimization application, or the relationships representing interacting objects and their governing probability distributions in the case of simulation. This sort of modeling is a particular strength of OR because it allows the power of various algorithms and analysis techniques to be brought to bear. The problem is making a large amount of complex information and data comprehensible to the user. New graphical user interfaces (drop-down menus, hierarchically expandable-contractable structures, buttons, etc.) have been rapidly adopted in commercial OR implementations. Many graphical user interfaces are being developed to aid in formulation (Chari and Sen 1997; Androulakis and Vrahatis 1996). Jones (1998) provides an excellent overview of the use of graphics and visualization technologies in modeling and solutions.

In addition, spreadsheets have become a ubiquitous paradigm for managing models and associated data, as well as tools for delivery and presentation of results. Many spreadsheets include linear and nonlinear programming algorithms as a part of the standard function set. The spreadsheets are changing the way OR analysts prepare, manage, and deliver the models. Lijima (1996) discusses an automatic model building approach.

Software: Data Structures and Databases

New data structures developed in CS are routinely used in OR algorithms. As any serious OR algorithm developer knows very well, learning about sparse matrix approaches such as linked lists, arrays, orthogonal lists, etc. is key in implementing an algorithm. As an example, Adler et al. (1989) focus on the data structures employed in their implementation of interior point methods.

The developments in data structures and databases have helped OR in modeling and algorithmics. But OR has been a key player in designing distributed databases. OR models and their solutions are important in designs of such databases. Information storage and retrieval research has also been the beneficiary of OR algorithms for query optimization. Kraft (1985) provides a good survey of this interface between OR and CS. OR approaches (e.g., mixed-integer programming) are also used in artificial intelligence. Specific examples include the use of mixed-integer programming (MIP) in automated theorem proving. A similar example is a graph theory–based approach for partitioning knowledge bases (Srikanth 1995).

Areas of Mutual Interest

Combinatorial optimization is an area of great mutual interest for both OR and CS. Here the many algorithmic tools in both communities are brought to bear. For example, the iconic traveling salesman problem (TSP), so easy to state but so difficult to solve, has been the subject of much research in both communities. OR has used approaches such as branch-and-bound and heuristics, while CS has attempted solutions using heuristics and tree search algorithms similar to branch-and-bound. The artificial intelligence and neural network communities have also focused on solving the TSP. Other heuristics approaches, such as genetic algorithms, simulated annealing, and tabu search, are being employed by both OR and CS specialists to solve combinatorial optimization problems. Computer scientists are using logic programming to solve routing and scheduling problems. These combinatorial problems have also been the focus of much research in the OR community. The TSP belongs to the larger class of routing problems, which continue to be of great interest to both communities. Potvin (2009) outlines the many methods and combinations thereof that have been applied to routing problems by both communities.

As noted above, computer design is the subject of research in both communities. This includes hardware

design, database design, and operating system design. An example given by Greenberg (1988) includes the use of random walk theory to analyze various storage allocation approaches, an important issue in operating systems. Telecommunications systems are also vital to modern information processing and the timely delivery of OR models and solutions. Real-time data access is a key in successful implementation of many models and that is possible only because of advances in telecommunications. As complex systems in their own right, telecommunications problems are the targets of much OR research: network design and routing, location analysis, etc. Decisions in telecommunications networks are based on OR approaches using queueing theory, Markov analysis, simulations, and MIP models.

Of course, the tremendous growth of the Internet has resulted in a complete transformation of OR/MS model development, solution, and delivery. It has also profoundly impacted the OR/MS profession in terms of education and professional communication through conferences and journals. Bhargava and Krishnan (1998) discuss this important interface.

Another example of the impact of CS on OR is the field of computational probability. Researchers continue to work on developing improved numerical techniques for solving large systems of equations appearing in stochastic models (Albin and Harris 1987). Simulation research and practice has also been a beneficiary of CS advances. One example is the use of artificial intelligence techniques in design and interpretation of simulations. Advances in parallel processing have led to active research in parallel simulation to speed up the computations (Fujimoto 1993).

Concluding Remarks

The objective was to illustrate the vibrancy of the symbiosis between OR and CS. As a final example, consider these areas covered in the *INFORMS Journal on Computing*: Computational Probability and Analysis, Constraint Programming and Optimization, Design and Analysis of Algorithms, Heuristic Search and Learning, Knowledge and Data Management, Modeling: Methods and Analysis, Simulation, Telecommunications, and Electronic Commerce. These areas are also covered in CS journals. As an article in *Computer World* (Betts 1993) noted, OR/MS needs corporate data for its algorithms and needs the algorithms used in strategic information systems to make a real impact. On the other hand, information systems (IS) groups need OR to build smart applications. Betts calls the individuals with significant OR/MS and CS/IS skills the new Efficiency Einsteins, a term that indeed appropriately describes the individuals trained in this interface.

See

- ▶ Algebraic Modeling Languages for Optimization
- ▶ Artificial Intelligence
- ▶ Combinatorics
- ▶ Computational Probability
- ▶ Constraint Programming
- ▶ Heuristics
- ▶ Information Systems and Database Design in OR/MS
- ▶ Integer and Combinatorial Optimization
- ▶ Integer-programming Problem
- ▶ Knowledge Management
- ▶ Linear Programming
- ▶ Nonlinear Programming
- ▶ Parallel Computing
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Simulation Optimization
- ▶ Telecommunication Networks
- ▶ Traveling Salesman Problem
- ▶ Vehicle Routing
- ▶ Visualization

References

Adler, L., Karmarkar, N., Resende, M. D. G., & Beiga, G. (1989). Data structures and programming techniques for the implementation of Karmarkar's algorithm. *ORSA Journal on Computing, 1*, 84–106.

Albin, S. L. & Harris, C. M. (1987). Statistical and computational problems in probability modeling. *Annals of Operations Research*, *8/9*.

Androulakis, G. S., & Vrahatis, M. N. (1996). OPTAC: A portable software package for analyzing and comparing

optimization methods by visualization. *Journal of Computational and Applied Mathematics, 72*(1), 41–62.
Betts, M. (1993, March 22). Efficiency Einsteins. *ComputerWorld*, pp. 63–65.
Bhargava, H., & Krishnan, R. (1998). The World Wide Web: Opportunities for operations research and management science. *INFORMS Journal on Computing, 10*(4), 359–383.
Bisschop, J. J., & Fourer, R. (1996). New constructs for the description of combinatorial optimization problems in algebraic modeling languages. *Computational Optimization and Applications, 6*(1), 83–116.
Chari, K., & Sen, T. K. (1997). An integrated modeling system for structured modeling using model graphs. *INFORMS Journal on Computing, 9*(4), 397–416.
Chinneck, J. W., Nakhla, M., & Zhang, Q. J. (2005). Computer-aided design for electrical and computer engineering. In H. J. Greenberg (Ed.), *Tutorials on emerging methodologies and applications in operations research* (pp. 6-1 to 6-44). Springer Science + Business Media.
Choobineh, J. (1991). SQLMP: A data sublanguage for representation and formulation of linear mathematical models. *ORSA Journal on Computing, 3*, 358–375.
Dantzig, G. (1988, August). Impact of linear programming on computer development. *OR/MS Today*, pp. 12–17.
Dantzig, G. (2002). Linear programming. *Operations Research, 50*, 42–47.
Eckstein, J. (1993). Large-scale parallel computing, optimization, and operations research: A survey. *ORSA/CSTS Newsletter, 14*(2), 11–12, 25–28.
Fourer, R. (1983). Modeling languages versus matrix generators for linear programming. *ACM Transactions on Mathematical Software, 9*, 143–183.
Fujimoto, R. M. (1993). Parallel discrete event simulation: Will the field survive? *ORSA Journal on Computing, 5*, 213–230.
Geoffrion, A. M. (1996). Structured modeling: Survey and future research directions. *Interactive Transactions of ORMS, 1*(3).
Glover, F., Laguna, M., Marti, R., & Womer, K. (2000). Fundamentals of scatter search and path relinking. *Control and Cybernetics, 39*, 653–684.
Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading: Addison-Wesley Publishing.
Greenberg, H. J. (1988). Interfaces between operations research and computer science. *OR/MS Today, 15*, 5.
Greenberg, H. J. (1992). Intelligent analysis support for linear programs. *Computers and Chemical Engineering, 16*, 659–674.
Hooker, J. N. (2007). *Integrated methods for optimization*. New York: Springer.
Jones, C. V. (1998). Visualization and modeling. *Interactive Transactions of ORMS, 2*(1).
Kraft, D. H. (1985). Advances in information retrieval: Where is that/* & % record? *Advances in Computers, 24*, 277–318.
Krishnan, R. (1993). Model management: Survey, future research directions, and a bibliography. *ORSA/CSTS Newsletter, 14*(1), 1–16.
Lijima, J. (1996). Automatic model building and solving for optimization problems. *Decision Support Systems, 18* (3&4), 293–300.
Nemhauser, G. L. (1994). The age of optimization: Solving large scale real-world problems. *Operations Research, 42*, 5–13.
Potvin, J.-Y. (2009). State-of-the art review: Evolutionary algorithms for vehicle routing. *INFORMS Journal on Computing, 21*, 518–548.
Powell, W. B. (2010). Merging AI and OR to solve high-dimensional stochastic optimization problems using approximate dynamic programming. *INFORMS Journal on Computing, 22*, 2–17.
Sharda, R. (1993). *Linear and discrete optimization modeling and optimization software: An industry resource guide*. Atlanta, GA: Lionheart Publishing.
Srikanth, R. (1995). A graph theory-based approach for partitioning knowledge bases. *INFORMS Journal on Computing, 7*, 286–297.
Thain, D., Tannenbaum, T., & Livny, M. (2005). Distributed computing in practice: The condor experience. *Concurrency and Compution: Practice and Experience, 17*, 323–356.
Zenios, S. (1989). Parallel numerical optimization: Current status and an annotated bibliography. *ORSA Journal on Computing, 1*, 20–43.

Concave Function

A function that is never below its linear interpolation. Mathematically, a function $f(x)$ is concave over a convex set S, if for any two points, x_1 and x_2 in S and for any $0 \leq \alpha \leq 1$, $f[\alpha x_1 + (1 - \alpha)x_2] \geq \alpha f(x_1) + (1 - \alpha) f(x_2)$.

Conclusion

A portion of a rule composed of series of one or more actions that the inference engine can carry out if a rule's premise can be established to be true.

See

- ▶ Artificial Intelligence
- ▶ Expert Systems

Condition Number

- ▶ Numerical Analysis

Conditional Value-at-Risk (CVaR)

Gaia Serraino[1] and Stanislav Uryasev[2]
[1]American Optimal Decisions, Gainesville, FL, USA
[2]University of Florida, Gainesville, FL, USA

Introduction

Conditional Value-at-Risk (CVaR), introduced by Rockafellar and Uryasev (2000), is a popular tool for managing risk. CVaR approximately (or exactly, under certain conditions) equals the average of some percentage of the worst case loss scenarios. CVaR risk measure is similar to the Value-at-Risk (VaR) risk measure which is a percentile of a loss distribution. VaR is heavily used in various engineering applications, including financial ones. VaR risk constraints are equivalent to the so called chance constraints on probabilities of losses. Some risk communities prefer VaR, others prefer chance (or probabilistic) functions. There is a close correspondence between CVaR and VaR: with the same confidence level, VaR is a lower bound for CVaR. Rockafellar and Uryasev (2000, 2002) showed that CVaR is superior to VaR in optimization applications. The problem of choice between VaR and CVaR, especially in financial risk management, has been quite popular in academic literature. Reasons affecting the choice between VaR and CVaR are based on the differences in mathematical properties, stability of statistical estimation, simplicity of optimization procedures, acceptance by regulators, etc.

Definition of VaR and CVaR

Let X be a random variable with the cumulative distribution function $F_X(z) = P\{X \le z\}$. X may have meaning of loss or gain. In what follows, X has meaning of loss and this impacts the sign of functions in the definition of VaR and CVaR. Figure 1 presents the graphical representation of VaR and CVaR.

Definition 1: Value-at-Risk. Value-at-Risk (VaR) of X with confidence level $\alpha \in]0,1[$ is

$$\mathrm{VaR}_\alpha(X) = \min\{z | \boldsymbol{F_X}(z) \ge \alpha\}. \tag{1}$$

By definition, $\mathrm{VaR}_\alpha(X)$ is a lower α-percentile of the random variable X. Value-at-Risk is commonly used in many engineering areas involving uncertainties, such as military, nuclear, material, air and space, finance, etc. For instance, finance regulations like Basel I and Basel II, use VaR-deviation measuring the width of daily loss distribution of a portfolio.

For normally distributed random variables, VaR is proportional to the standard deviation. If $X \sim N(\mu, \sigma^2)$ and $F_X(z)$ is the cumulative distribution function of X, then (see Rockafellar and Uryasev 2000),

$$\mathrm{VaR}_\alpha(X) = F_X^{-1}(\alpha) = \mu + k(\alpha)\sigma, \tag{2}$$

where $k(\alpha) = \sqrt{2}\mathrm{erf}^{-1}(2\alpha - 1)$ and $\mathrm{erf}(\boldsymbol{z}) = \frac{2}{\sqrt{\pi}}\int_0^z \boldsymbol{e}^{-t^2}\boldsymbol{dt}$.

Ease and intuitiveness of VaR is counterbalanced by its mathematical properties. As a function of the confidence level, for discrete distributions $\mathrm{VaR}_\alpha(X)$ is a non-convex, discontinuous function. For discussion of numerical difficulties of VaR optimization see, for example, Rockafellar (2007), and Rockafellar and Uryasev (2000).

Definition 2: Conditional Value-at-Risk. An alternative percentile measure of risk is the Conditional Value-at-Risk (CVaR). For random variables with continuous distribution functions, $\mathrm{CVaR}_\alpha(X)$ equals the conditional expectation of X subject to $X \ge \mathrm{VaR}_\alpha(X)$. This definition is the basis for the name of Conditional Value-at-Risk. The term Conditional Value-at-Risk has been introduced by Rockafellar and Uryasev (2000). The general definition of CVaR for random variables with possibly discontinuous distribution function is as follows (see Rockafellar and Uryasev 2002).

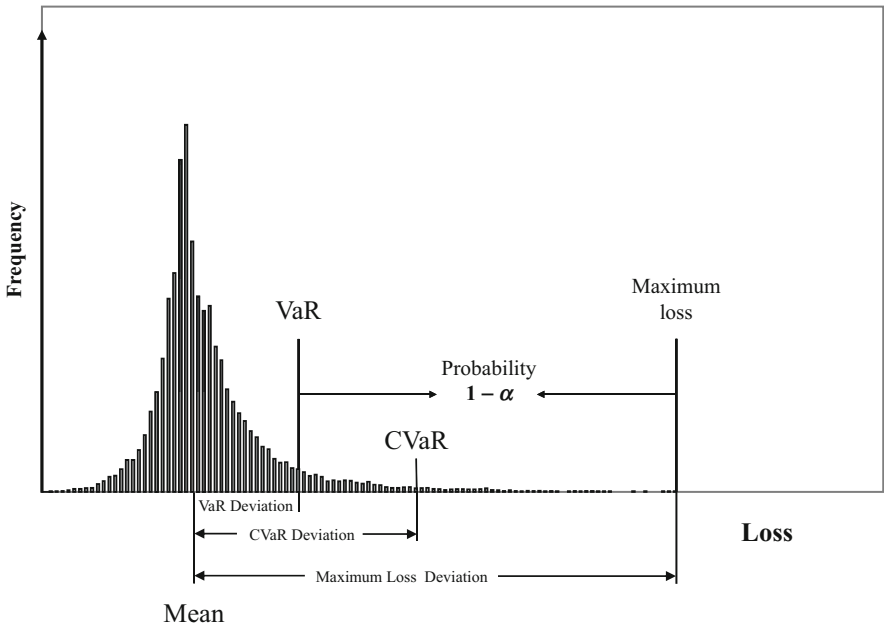
Conditional Value-at-Risk (CVaR) of X with confidence level $\alpha \in]0,1[$ is the mean of the generalized α-tail distribution:

$$\mathrm{CVaR}_\alpha(X) = \int_{-\infty}^{\infty} z dF_X^\alpha(z), \tag{3}$$

where

$$F_X^\alpha(z) = \begin{cases} 0, & \text{when } z < \mathrm{VaR}_\alpha(X), \\ \frac{F_X(z)-\alpha}{1-\alpha}, & \text{when } z \ge \mathrm{VaR}_\alpha(X). \end{cases}$$

Contrary to popular belief, in the general case, $\mathrm{CVaR}_\alpha(X)$ is not equal to an average of outcomes



Conditional Value-at-Risk (CVaR), Fig. 1 Risk Functions. Graphical Representation of VaR, VaR Deviation, CVaR, CVaR Deviation, Max Loss, Max Loss Deviation

greater than $\mathrm{VaR}_\alpha(X)$. For general distributions, one may need to split a probability atom. For example, when the distribution is modeled by scenarios, CVaR may be obtained by averaging a fractional number of scenarios. To explain this idea in more detail, alternative definitions of CVaR are presented in the following. Let $\mathrm{CVaR}^+_\alpha(X)$, called upper CVaR, be the conditional expectation of X subject to $X > \mathrm{VaR}_\alpha(X)$.

$$\mathrm{CVaR}^+_\alpha(X) = E[X \,|\, X > \mathrm{VaR}_\alpha(X)].$$

$\mathrm{CVaR}_\alpha(X)$ can be alternatively defined as the weighted average of $\mathrm{VaR}_\alpha(X)$ and $\mathrm{CVaR}^+_\alpha(X)$, as follows. If $F_X(\mathrm{VaR}_\alpha(X))<1$, so there is a chance of a loss greater than $\mathrm{VaR}_\alpha(X)$, then

$$\mathrm{CVaR}_\alpha(X) = \lambda_\alpha(X)\,\mathrm{VaR}_\alpha(X) + (1-\lambda_\alpha(X))\,\mathrm{CVaR}^+_\alpha(X), \tag{4}$$

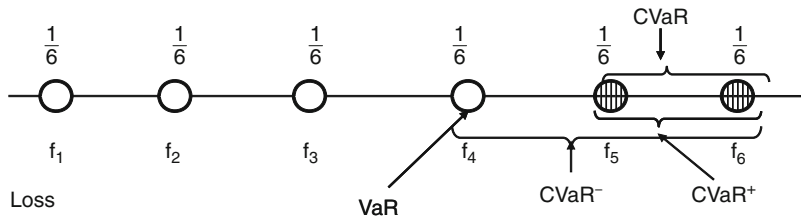
$$\text{where } \lambda_\alpha(X) = \frac{F_X(\mathrm{VaR}_\alpha(X)) - \alpha}{1-\alpha}, \tag{5}$$

whereas if $F_X(\mathrm{VaR}_\alpha(X)) = 1$, so that $\mathrm{VaR}_\alpha(X)$ is the highest loss that can occur, then

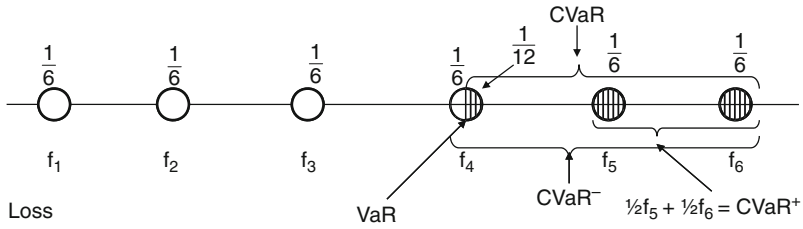
$$\mathrm{CVaR}_\alpha(x) = \mathrm{VaR}_\alpha(x)\ . \tag{6}$$

Definition (4) demonstrates that CVaR is not defined as a conditional expectation. The function $\mathrm{CVaR}^-_\alpha(X) = E[X \,|\, X \geq \mathrm{VaR}_\alpha(X)]$, called "lower CVaR", coincides with $\mathrm{CVaR}_\alpha(X)$ for continuous distributions; however, for general distributions it is discontinuous with respect to α and not convex. The construction of CVaR_α as a weighted average of VaR_α and $\mathrm{CVaR}^+_\alpha(X)$ is a major innovation. Neither VaR nor $\mathrm{CVaR}^+_\alpha(X)$ behaves well as a measure of risk for general loss distributions (both are discontinuous functions), but CVaR is a very attractive function. It is continuous with respect to α and jointly convex in (X, α). The unusual feature in the definition of CVaR is that VaR atom can be split. If $F_X(x)$ has a vertical discontinuity gap, then there is an interval of confidence level α having the same VaR. The lower and upper endpoints of that interval are $\alpha^- = F_X(\mathrm{VaR}^-_\alpha(X))$ and $\alpha^+ = F_X(\mathrm{VaR}_\alpha(X))$ where $F_X(\mathrm{VaR}^-_\alpha(X)) = P\{X < \mathrm{VaR}_\alpha(X)\}$. When $F_X(\mathrm{VaR}^-_\alpha(X)) < \alpha < F_X(\mathrm{VaR}_\alpha(X)) < 1$ the atom $\mathrm{VaR}_\alpha(X)$ having total probability $\alpha^+ - \alpha^-$ is split by the confidence level α in two pieces with probabilities $\alpha^+ - \alpha$ and $\alpha - \alpha^-$. Equation 4 highlights this splitting.

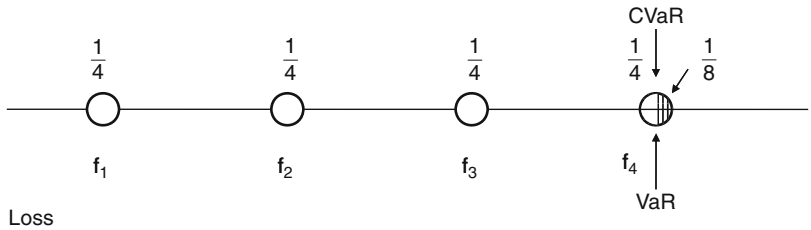
CVaR definition is illustrated further with the following examples, in which 6 equally likely scenarios have losses $f_1 \ldots f_6$. Let $\alpha = \frac{2}{3}$, see Fig. 2.



Conditional Value-at-Risk (CVaR), Fig. 2 CVaR Example 1. Computation of CVaR when α does not spilt the atom



Conditional Value-at-Risk (CVaR), Fig. 3 CVaR Example 2. Computation of CVaR when α splits the atom



Conditional Value-at-Risk (CVaR), Fig. 4 CVaR Example 3. Computation of CVaR when α splits the last atom

In this case α does not split any probability atom. Then $\mathrm{VaR}_\alpha(X) < \mathrm{CVaR}_\alpha^-(X) < \mathrm{CVaR}_\alpha(X) = \mathrm{CVaR}_\alpha^+(X)$, $\lambda_\alpha(X) = \frac{F_X(\mathrm{VaR}_\alpha(X)) - \alpha}{1-\alpha} = 0$ and $\mathrm{CVaR}_\alpha(X) = \mathrm{CVaR}_\alpha^+(X) = \frac{1}{2}f_5 + \frac{1}{2}f_6$, where f_5, f_6 are losses number five and six.

Let now $\alpha = \frac{7}{12}$, see Fig. 3. In this case α does split the $\mathrm{VaR}_\alpha(X)$ atom, $\lambda_\alpha(X) = \frac{F_X(\mathrm{VaR}_\alpha(X)) - \alpha}{1-\alpha} > 0$ and $\mathrm{CVaR}_\alpha(X)$ is given by:

$$\begin{aligned}\mathrm{CVaR}_\alpha(X) &= \frac{1}{5}\mathrm{VaR}_\alpha(X) + \frac{4}{5}\mathrm{CVaR}_\alpha^+(X) \\ &= \frac{1}{5}f_4 + \frac{2}{5}f_5 + \frac{2}{5}f_6 \ .\end{aligned}$$

In the last case, there are four equally likely scenarios and $\alpha = \frac{7}{8}$ splits the last atom; see Fig. 4. Now $\mathrm{VaR}_\alpha(X) = \mathrm{CVaR}_\alpha^-(X) = \mathrm{CVaR}_\alpha(X)$, upper CVaR, $\mathrm{CVaR}_\alpha^+(X)$ is not defined, $\lambda_\alpha(X) = \frac{F_X(\mathrm{VaR}_\alpha(X)) - \alpha}{1-\alpha} > 0$ and $\mathrm{CVaR}_\alpha(X) = \mathrm{VaR}(X) = f_4$. Portfolio Safeguard package (see American Optimal Decisions 2009), defines CVaR function for discrete distributions equivalently to (4) through the lower CVaR and upper CVaR. Suppose that $\mathrm{VaR}_\alpha(X)$ atom having total probability $\alpha^+ - \alpha^-$ is split by the confidence level α in two pieces with probabilities $\alpha^+ - \alpha$ and $\alpha - \alpha^-$. Then,

$$\begin{aligned}\mathrm{CVaR}_\alpha(X) = &\frac{\alpha^+ - \alpha}{\alpha^+ - \alpha^-}\,\frac{1-\alpha^-}{1-\alpha}\,\mathrm{CVaR}_\alpha^-(X) \\ &+ \frac{\alpha - \alpha^-}{\alpha^+ - \alpha^-}\,\frac{1-\alpha^+}{1-\alpha}\,\mathrm{CVaR}_\alpha^+(X),\end{aligned} \tag{7}$$

$$\begin{aligned}\text{where } \mathrm{CVaR}_\alpha^-(X) &= E[X \,|\, X \geq \mathrm{VaR}_\alpha(X)], \\ \mathrm{CVaR}_\alpha^+(X) &= E[X \,|\, X > \mathrm{VaR}_\alpha(X)]\,.\end{aligned} \tag{8}$$

Pflug (2000) followed a different approach and suggested to define CVaR via an optimization problem which he borrowed from Rockafellar and Uryasev (2000)

$$\begin{aligned}\mathrm{CVaR}_\alpha(X) = &\\ &\min_C \left\{ C + \frac{1}{1-\alpha}\,E[X - C]^+ \right\}, \\ &\text{where } [t]^+ = \max\{0, t\}.\end{aligned} \tag{9}$$

One more equivalent representation of CVaR was given by Acerbi (2002), who showed that CVaR is equal to "expected shortfall" defined by

$$\mathrm{CVaR}_{\alpha}(X) = \frac{1}{\alpha}\int_0^{\alpha} \mathrm{VaR}_{\beta}(X)d\beta\,.$$

For normally distributed random variables, CVaR deviation is proportional to the standard deviation. If $X \sim N(\mu, \sigma^2)$, then (see Rockafellar and Uryasev 2000),

$$\begin{aligned}\mathrm{CVaR}_{\alpha}(X) &= E[X \mid X \geq \mathrm{VaR}_{\alpha}(X)] \\ &= \mu + k_1(\alpha)\sigma,\end{aligned} \tag{10}$$

where

$$k_1(\alpha) = \left(\sqrt{2\pi} exp\left(\mathrm{erf}^{-1}(2\alpha - 1)\right)^2 (1-\alpha)\right)^{-1}$$

and $\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}}\int_0^z e^{-t^2}dt$.

CVaR Optimization

CVaR optimization has been researched in Rockafellar and Uryasev (2000) and Uryasev (2000). Nowadays VaR has achieved the high status of being written into industry regulations (for instance, in regulations for financial companies). It is difficult to optimize VaR numerically when losses are not normally distributed. Only recently VaR optimization was included in commercial packages such as Portfolio Safeguard (see American Optimal Decisions 2009). As a tool in optimization modeling, CVaR has superior properties in many respects. CVaR optimization is consistent with VaR optimization and yield the same results for normal or elliptical distributions (see definition of elliptical distribution in (see definition of elliptical distribution in Embrechts et al. (2001)); for models with such distributions, working with VaR, CVaR or minimum variance (Markowitz 1952) is equivalent (see Rockafellar and Uryasev 2000). Most importantly, CVaR can be expressed by a minimization formula suggested by Rockafellar and Uryasev (2000). This formula can be incorporated into the optimization problem with respect to decision variables $x \in X \in \Re^n$ that are designed to minimize risk or shape it within bounds. Significant shortcuts are thereby achieved while preserving the crucial problem features like convexity. Let the random loss function $f(x, y)$ depends upon the decision vector x and a random vector y of risk factors. For instance, $f(x, y) = -(y_1x_1 + y_2x_2)$ is the negative return of a portfolio involving two instruments. Here x_1, x_2 are positions and y_1, y_2 are rates of returns of two instruments in the portfolio. The main idea in Rockafellar and Uryasev (2000) is to define a function that can be used instead of CVaR:

$$F_{\alpha}(x, \zeta) = \zeta + \frac{1}{1-\alpha}E\{[f(x, y) - \zeta]^+\}\,. \tag{11}$$

The authors proved that:

1. $F_{\alpha}(x, \zeta)$ is convex w.r.t. α,
2. $\mathrm{VaR}_{\alpha}(x)$ is a minimum point of function $F_{\alpha}(x, \zeta)$ w.r.t. ζ,
3. Minimizing $F_{\alpha}(x, \zeta)$ w.r.t. ζ gives $\mathrm{CVaR}_{\alpha}(x)$:

$$\mathrm{CVaR}_{\alpha}(x) = \min_{\alpha} F_{\alpha}(x, \zeta)\,. \tag{12}$$

In optimization problems, CVaR can enter into the objective or constraints or both. A big advantage of CVaR over VaR in that context is the preservation of convexity, i.e., if $f(x, y)$ is convex in x than $\mathrm{CVaR}_{\alpha}(x)$ is convex in x. Moreover, if $f(x, y)$ is convex in x then the function $F_{\alpha}(x, \zeta)$ is convex in both x and ζ. This convexity is very valuable because minimizing $F_{\alpha}(x, \zeta)$ over $(x, \zeta) \in X \times \Re$, results in minimizing $\mathrm{CVaR}_{\alpha}(x)$

$$\min_{x \in X} \mathrm{CVaR}_{\alpha}(x) = \min_{(x,\zeta) \in X \times \Re} F_{\alpha}(x, \zeta)\,. \tag{13}$$

In addition, if (x^*, ζ^*) minimizes F_{α} over $X \times \Re$, then not only does x^* minimize $\mathrm{CVaR}_{\alpha}(x)$ over X but also

$$\mathrm{CVaR}_{\alpha}(x^*) = F_{\alpha}(x^*, \zeta^*)\,.$$

In risk management CVaR can be utilized to "shape" the risk in an optimization model. For that purpose several confidence levels can be specified. Rockafellar and Uryasev (2000) showed that for any selection of confidence levels α_i and loss tolerances ω_i, $i = 1, \ldots, l$, the problem:

$$\begin{aligned} &\min_{x \in X} \; g(x) \\ s.\,t. \quad &\mathrm{CVaR}_{\alpha_i}(x) \leq \omega_i, \quad i = 1, \ldots, l \end{aligned} \tag{14}$$

is equivalent to the problem:

$$\min_{x,\zeta_1,..,\zeta_l,\in X\times\Re\times...\Re} g(x) \quad s.\,t. \quad F_{\alpha_i}(x,\zeta_i) \le \omega_i, \quad i=1,\ldots,l. \tag{15}$$

When X and g are convex and $f(x,y)$ is convex in x, the optimization problems (13) and (14) are ones of convex programming and thus especially favorable for computation. When Y is a discrete probability space with elements y_k, $k=1,..,N$ having probabilities p_k, $k=1,..,N$:

$$F_{\alpha_i}(x,\zeta_i) = \zeta_i + \frac{1}{1-\alpha_i} \times \sum_{k=1}^{N} p_k[\,f(x,y_k)-\zeta_i]^+ \,. \tag{16}$$

The constraint $F_\alpha(x,\zeta)\le\omega$ can be replaced by a system of inequalities by introducing additional variables η_k :

$$\eta_k \ge 0, \quad f(x,y_k)-\zeta-\eta_k \le 0, \quad k=1,..,N, \tag{17}$$

$$\zeta + \frac{1}{1-\alpha}\sum_{k=1}^{N} p_k\eta_k \le \omega \,.$$

The minimization problem in (14) can be converted into the minimization of $g(x)$ with the constraints $F_{\alpha_i}(x,\zeta_i)\le\omega_i$ being replaced as presented in (17). When f is linear in x, constraints (17) are linear.

Risk Measures

Axiomatic investigation of risk measures was suggested by Artzner et al. (1999). Rockafellar (2007) defined a functional $\mathcal{R}:\mathcal{L}^2\to]-\infty,\infty]$ as a coherent risk measure in the extended sense if:

R1: $\mathcal{R}(\mathrm{C})=\mathrm{C}$ for all constant C,

R2: $\mathcal{R}((1-\lambda)X+\lambda X')\le(1-\lambda)\mathcal{R}(X)+\lambda R(X')$ for $\lambda\in]0,1[$ (convexity),

R3: $\mathcal{R}(X)\le\mathcal{R}(X')$ when $X\le X'$ (monotonicity),

R4: $\mathcal{R}(X)\le 0$ when $||X^k-X||_2\to 0$ with $\mathcal{R}(X^k)\le 0$ (closedness).

A functional $\mathcal{R}:\mathcal{L}^2\to]-\infty,\infty]$ is called a *coherent risk measure in the basic sense* if it satisfies axioms R1, R2, R3, R4 and additionally the axiom 4

R5: $\mathcal{R}(\lambda X)=\lambda\mathcal{R}(X)$ for $\lambda>0$ (positive homogeneity).

A functional $\mathcal{R}:\mathcal{L}^2\to]-\infty,\infty]$ is called an averse risk measure in the extended sense if it satisfies axioms R1, R2, R4 and

R6: $\mathcal{R}(X)>EX$ for all nonconstant X (aversity).

Aversity has the interpretation that the risk of loss in a nonconstant random variable X cannot be acceptable, i.e. $\mathcal{R}(X)<0$, unless $EX<0$.

A functional $\mathcal{R}:\mathcal{L}^2\to]-\infty,\infty]$ is called an averse risk measure in the basic sense if it satisfies R1, R2, R4, R6 and also R5.

Examples of coherent measures of risk are $\mathcal{R}(X)=\mu X=E[X]$ or $\mathcal{R}(X)=\sup X$. However, $R(X)=\mu(X)+\lambda\sigma(X)$ for some $\lambda>0$ is not a coherent measure of risk since it does not satisfies the monotonicity axiom R3.

$R(X)=\mathrm{VaR}_\alpha(X)$ is not a coherent nor an averse risk measure. The problem lies in the convexity axiom $R2$, which is equivalent to the combination of positive homogeneity and subadditivity, this last defined as $\mathcal{R}(X+X')\le\mathcal{R}(X)+\mathcal{R}(X')$. Although positive homogeneity is obeyed, the subadditivity is violated. The lack of coherency can destroy convexity; this can still be present if the distribution of the random variable X belongs to the log-concave class, but even then there are technical hurdles because the convexity of R is missing relative to the entire space $\mathcal{L}^2$. It has been proved, for example in Acerbi and Tasche (2002), Pug (2000), Rockafellar and Uryasev (2002), that for any probability level $\alpha\in]0,1[$, $\mathcal{R}(X)=\mathrm{CVaR}_\alpha(X)$ is a coherent measure of risk in the basic sense. $\mathrm{CVaR}_\alpha(X)$ is also an averse measure of risk for $\alpha\in]0,1]$ An averse measure of risk might not be coherent; a coherent measure might not be averse.

Deviation Measures

This section refers to Rockafellar (2007) and Rockafellar et al. (2006). A functional $\mathcal{D}:\mathcal{L}^2\to[0,\infty]$ is called a deviation measure in the extended sense if it satisfies

D1: $\mathcal{D}(C)=0$ for constant C, but $D(X)>0$ for nonconstant X,

D2: $\mathcal{D}((1-\lambda)X+\lambda X')\le(1-\lambda)\mathcal{D}(X)+\lambda D(X')$ for $\lambda\in]0,1[$ (convexity),

D3: $\mathcal{D}(X) \leq d$ when $||X^k - X||_2 \to 0$ with $\mathcal{D}(X^k) \leq$ d (closedness).

A functional is called a deviation measure in the basic sense when it satisfies axioms D1, D2, D3, and furthermore

D4: $\mathcal{D}(\lambda X) = \lambda \mathcal{D}(X)$ for $\lambda > 0$ (positive homogeneity).

A deviation measure in extended or basic sense is called a coherent measure in extended or basic sense if it additionally satisfies

D5: $\mathcal{D}(X) \leq \sup X - E[X]$ for all X (upper range boundedness).

An immediate example of a deviation measure in the basic sense is the standard deviation:

$$\sigma(X) = \left(E[X - EX]^2\right)^{1/2},$$

which satisfies axioms D1, D2, D3, D4, but not D5. I.e., standard deviation is not a coherent deviation measure. Here are more examples of deviation measures in the basic sense:

Standard semideviations

$$\sigma_+(X) = \left(E[\max\{X - EX, 0\}]^2\right)^{1/2},$$

$$\sigma_-(X) = \left(E[\max\{EX - X, 0\}]^2\right)^{1/2},$$

Mean Absolute Deviation

$$MAD(X) = E[|X - EX|].$$

Moreover it is possible to define the α-Value-at-Risk deviation measure and the α-Conditional Value-at-Risk deviation measure as:

$$\mathrm{VaR}_\alpha^\Delta(X) = \mathrm{VaR}_\alpha(X - EX) \tag{18}$$

and

$$\mathrm{CVaR}_\alpha^\Delta(X) = \mathrm{CVaR}_\alpha(X - EX)\ . \tag{19}$$

VaR deviation measure $\mathrm{VaR}_\alpha^\Delta(X)$ is not a deviation measure in the general or basic sense because the convexity axiom D2 is not satisfied. CVaR deviation measure $\mathrm{CVaR}_\alpha^\Delta(X)$ is a coherent deviation measure in the basic sense.

Risk Measures Versus Deviation Measures

Rockafellar et al. originally in Rockafellar et al. (2006), and then in Rockafellar (2007) obtained the following result:

Theorem 1. *A one-to-one correspondence between deviation measures D in the extended sense and averse risk measures $\mathcal{R}$ in the extended sense is expressed by the relations*

$$\mathcal{R}(X) = \mathcal{D}(X) + EX,$$

$$\mathcal{D}(X) = \mathcal{R}(X - EX),$$

additionally,

$$\mathcal{R}\ \textit{is coherent}\ \leftrightarrow\ \mathcal{D}\ \textit{is coherent}\ .$$

Moreover the positive homogeneity is preserved:

$$\mathcal{R}\ \textit{is positively homogeneous} \leftrightarrow\ \mathcal{D}\ \textit{is positively homogeneous}\ .$$

i.e., for an averse risk measures $\mathcal{R}$ in the basic sense and a deviation measures $\mathcal{D}$ in the basic sense the one-to-one correspondence is valid, and additionally, coherent $\mathcal{R}$ $\leftrightarrow$ coherent $\mathcal{D}$.

With this theorem it is obtained that for the standard deviation, $\sigma(X)$, which is a deviation measure in the basic sense, the counterpart is the standard risk $EX + \sigma(X)$, which is a risk averse measure in the basic sense. For CVaR deviation, $\mathrm{CVaR}_\alpha^\Delta(X)$, which is a coherent deviation measure in the basic sense, the counterpart is CVaR risk, $\mathrm{CVaR}_\alpha(X)$, which is a risk averse coherent measure in the basic sense.

Another coherent deviation measure in the basic sense is the so-called Mixed Deviation CVaR, quite promising for risk management purposes. Mixed Deviation CVaR is defined as:

$$Mixed - \mathrm{CVaR}_\alpha^\Delta(X) = \sum_{k=1}^{K} \lambda_k \mathrm{CVaR}_{\alpha_k}^\Delta(X)$$

for $\lambda_k \geq 0$, $\sum_{k=1}^K \lambda_k = 1$ and α_k in $]0, 1[$. The counterpart to the Mixed Deviation CVaR is the

Mixed CVaR, which is the coherent averse risk measure in the basic sense, defined by

$$Mixed - \mathrm{CVaR}_{\alpha}(X) = \sum_{k=1}^{K} \lambda_k \mathrm{CVaR}_{\alpha_k}(X) \ .$$

Generalized Regression Problem

In linear regression a random variable Y is approximated in terms of random variables $X_1, X_2, \ldots, X_n$ by an expression $c_0 + c_1X_1 + \ldots + c_nX_n$. The coefficients are chosen by minimizing mean square error:

$$\min_{c_0, c_1, .., c_n} E(Y - [c_0 + c_1X_1 + \ldots + c_nX_n])^2 \ . \qquad (20)$$

Mean square error minimization is equivalent to minimizing standard deviation with the unbiasedness constraint (see Rockafellar et al. 2002, 2008):

$$\begin{array}{ll} \min & \sigma(Y - [c_0 + c_1X_1 + \ldots + c_nX_n]) \\ s.\,t. & E[c_0 + c_1X_1 + \ldots + c_nX_n] = EY \ . \end{array} \qquad (21)$$

Rockafellar et al. (2002, 2008) considered a general axiomatic setting for error measures and corresponding deviation measures. They defined an error measure as a functional $\mathcal{E} : \mathcal{L}^2(\Omega) \to [0, \infty]$ satisfying the axioms

E1: $\mathcal{E}(0) = 0$, $\mathcal{E}(X) > 0$ for $X \neq 0$, $\mathcal{E}(C) < \infty$ for constant C

E2: $\mathcal{E}(\lambda X) = \lambda \mathcal{E}(X)$ for $\lambda > 0$ (positive homogeneity)

E3: $\mathcal{E}(X + X') \leq \mathcal{E}(X) + \mathcal{E}(X')$ for all X and X' (subadditivity)

E4: $\{X \in \mathcal{L}^2(\Omega) | \mathcal{E}, (X) \leq c\}$ is closed for all $c < \infty$ (lower semicontinuity)

For an error measure $\mathcal{E}$ the projected deviation measure $\mathcal{D}$ is defined by the equation, $\mathcal{D}(X) = \min_C \mathcal{E}(X - C)$, and the statistic, $S(X)$, is defined by $\mathcal{S}(X) = \arg\min_C \mathcal{E}(X - C)$. Their main finding is that the general regression problem:

$$\min_{c_0, c_1, .., c_n} \mathcal{E}(Y - [c_0 + c_1X_1 + \ldots + c_nX_n]) \qquad (22)$$

is equivalent to:

$$\min_{c_1, .., c_n} \mathcal{D}(Y - [c_1X_1 + \ldots + c_nX_n])$$
$$c_0 \in \mathcal{S}(Y - [c_1X_1 + \ldots + c_nX_n]) \ .$$

The equivalence of optimization problems (20) and (21) is a special case of this theorem This leads to the identification of a link between statistical work on percentile regression (see Koenker and Bassett 1978) and CVaR deviation measure: minimization of the Koenker and Bassett error measure is equivalent to minimization of CVaR deviation. Rockafellar et al. (2008) show that when the error measure is the Koenker and Basset function: $\mathcal{E}^{\alpha}_{KB}(X) = E[\max\{0, X\} + (\alpha^{-1} - 1)\max\{0, -X\}]$ the projected measure of deviation is: $\mathcal{D}(X) = \mathrm{CVaR}^{\Delta}_{\alpha}(X) = \mathrm{CVaR}_{\alpha}(X - EX)$ with the corresponding averse measure of risk and associated statistic given by

$$\mathcal{R}(X) = \mathrm{CVaR}_{\alpha}(X),$$

$$\mathcal{S}(X) = \mathrm{VaR}_{\alpha}(X) \ .$$

Then:

$$\min_{C \in \Re} (E[X - C]_+ + (\alpha^{-1} - 1)E[X - C]_-)$$
$$= \mathrm{CVaR}^{\Delta}_{\alpha}(X),$$

$$\arg\min_{C \in \Re} (E[X - C]_+ + (\alpha^{-1} - 1)E[X - C]_-)$$
$$= \mathrm{VaR}_{\alpha}(X) \ .$$

Similar result is available for the "mixed Koenker and Bassett error measure" and the corresponding mixed deviation CVaR (see Rockafellar et al. 2008).

Comparative Analysis of VaR and CVaR

VaR is a relatively simple risk management notion. Intuition behind α-percentile of a distributions is easily understood and VaR has a clear interpretation: how much it is possible to lose with certain confidence level. VaR is a single number measuring risk, defined by some specified confidence level, e.g., $\alpha = 0.95$.

Two distributions can be ranked by comparing their VaR's for the same confidence level. Specifying VaR for all confidence levels completely defines the distribution. In this sense, VaR is superior to the standard deviation. Unlike the standard deviation, VaR focuses on a specific part of the distribution specified by the confidence level. This is what is often needed, which made VaR popular in risk management, including finance, nuclear, air and space, material science, and various military applications. One of important properties of VaR is stability of estimation procedures. Since VaR disregards the tail, it is not affected by very high tail losses, which are usually difficult to measure. VaR is estimated with parametric models, for instance Covariance-VaR based on the normal distribution assumption is very well known in finance, with simulation models such as historical or Monte Carlo or by using approximations based on second order Taylor expansion.

VaR does not account for properties of the distribution beyond the confidence level. This implies that $\mathrm{VaR}_\alpha(X)$ may increase dramatically with a small increase in α. In order to adequately estimate risk in the tail, one may need to calculate several VaRs with different confidence levels. The fact that VaR disregards the tail of the distribution may lead to unintentional bearing of high risks. In financial setting, for instance, the strategy of "naked" shorting deep out-of-the-money options will result most of the time in receiving an option premium without any loss at expiration. However, there is a chance of a big adverse market movement leading to an extremely high loss. VaR cannot capture this risk. Risk control using VaR may lead to undesirable results for skewed distributions. VaR is a non-convex and discontinuous function for discrete distributions. For instance, in financial setting, VaR is a non-convex and discontinuous function w.r.t. portfolio positions when returns have discrete distributions. This makes VaR optimization a challenging computational problem. There are codes, such as Portfolio Safeguard (PSG), that can work with VaR functions very efficiently. Portfolio Safeguard can optimize VaR performance function and also shape distributions with multiple VaR constraints. For instance, in portfolio optimization it is possible to maximize expected return with several VaR constraints at different confidence levels.

CVaR has a clear engineering interpretation. It measures outcomes which hurt the most. For example, if L is a loss then the constraint $\mathrm{CVaR}_\alpha(L) \le \bar{L}$ ensures that the average of $(1-\alpha)\%$ highest losses does not exceed $\bar{L}$. Defining $\mathrm{CVaR}_\alpha(X)$ for all confidence levels α in $(0,1)$ completely specifies the distribution of X. In this sense it is superior to standard deviation. Conditional Value-at-Risk has several attractive mathematical properties. CVaR is a coherent risk measure. $\mathrm{CVaR}_\alpha(X)$ is continuous with respect to α. CVaR of a convex combination of random variables $\mathrm{CVaR}_\alpha(w_1X_1 + \ldots + w_nX_n)$ is a convex function with respect to $(w_1, \ldots, w_n)$. In financial settings, CVaR of a portfolio is a convex function of portfolio positions. CVaR optimization can be reduced to convex programming, in some cases to linear programming (i.e. for discrete distributions).

CVaR is more sensitive than VaR to estimation errors. If there is no good model for the tail of the distribution, CVaR value may be quite misleading; accuracy of CVaR estimation is heavily affected by accuracy of tail modelling. For instance, historical scenarios often do not provide enough information about tails, hence it is necessary to assume a certain model for the tail to be calibrated on historical data. In the absence of a good tail model, one should not count on CVaR. In financial settings, equally weighted portfolios may outperform CVaR-optimal portfolios out of sample when historical data have mean reverting characteristics. VaR and CVaR measure different parts of the distribution. Depending on what is needed, one may be preferred over the other. This topic can be illustrated with financial applications of VaR and CVaR, to examine which one of these measures is better for portfolio optimization. A trader may prefer VaR to CVaR, as he may like high uncontrolled risks; VaR is not as restrictive as CVaR with the same confidence level. Nothing dramatic happens to a trader in case of high losses. He will not pay losses from his pocket; if fired, he may move to some other company. A company owner will probably prefer CVaR; he has to cover large losses if they occur, hence he "really" needs to control tail events. A board of directors of a company may prefer to provide VaR based reports to shareholders and regulators since it is less than CVaR with the same confidence level. However,

CVaR may be used internally, thus creating asymmetry of information between different parties.

In financial optimization, VaR may be better for optimizing portfolios when good models for tails are not available. VaR disregards the hardest to measure events. CVaR may not perform well out of sample when portfolio optimization is run with poorly constructed set of scenarios. Historical data may not give right predictions of future tail events because of mean-reverting characteristics of assets. High returns typically are followed by low returns, hence CVaR based on history may be quite misleading in risk estimation. If a good model of tail is available, then CVaR can be accurately estimated and CVaR should be used. CVaR has superior mathematical properties and can be easily handled in optimization and statistics. When comparing stability of estimation of VaR and CVaR, appropriate confidence levels for VaR and CVaR must be chosen, avoiding comparison of VaR and CVaR for the same level of α, as they refer to different parts of the distribution (Sarykalin et al. 2008).

References

Acerbi, C. (2002). Spectral measures of risk: A coherent representation of subjective risk aversion. *Journal of Banking and Finance, 26*, 1505–1518.

Acerbi, C., & Tasche, D. (2002). On the coherence of expected shortfall. *Journal of Banking and Finance, 26*, 1487–1503.

American Optimal Decisions. (2009). Portfolio Safeguard (PSG).

Artzner, P., Delbaen, F., Eber, J. M., & Heath, D. (1999). Coherent measures of risk. *Mathematical Finance, 9*, 203–227.

Embrechts, P., Mc Neil, A. J., & Straumann, D. (2001). Correlation and dependency in risk management: Properties and pitfalls. In M. Dempster (Ed.), *Risk management: Value at risk and beyond*. Cambridge: Cambridge University Press.

Koenker, R., & Bassett, G. W. (1978). Regression quantiles. *Econometrica, 46*, 33–50.

Markowitz, H. M. (1952). Portfolio selection. *Journal of Finance, 7*(1), 77–91.

Pflug, G. C. (2000). Some remarks on the value-at-risk and the conditional value-at-risk. In S. P. Uryasev (Ed.) *Probabilistic constrainted optimization: Methodology and applications*. (pp. 278–287). Kluwer Academic Publishers.

Rockafellar, R. T. (2007). Coherent approaches to risk in optimization under uncertainty. In INFORMS (Ed.). *Tutorials in operations research*, (pp. 38–61).

Rockafellar, R. T., & Uryasev, S. P. (2000). Optimization of conditional value-at-risk. *Journal of Risk, 2*, 21–42.

Rockafellar, R. T., & Uryasev, S. P. (2002). Conditional value-at-risk for general loss distributions. *Journal of Banking and Finance, 26*, 1443–1471.

Rockafellar, R. T., Uryasev, S., & Zabarankin, M. (2002). Deviation measures in generalized linear regression, Research Report 2002–9, ISE Dept., University of Florida.

Rockafellar, R. T., Uryasev, S., & Zabarankin, M. (2008). Risk tuning with generalized linear regression. *Mathematics of Operations Research, 33*(3), 712–729.

Rockafellar, R. T., Uryasev, S., & Zabarankin, M. (2006). Generalized deviations in risk analysis. *Finance and Stochastics, 10*, 51–74.

Sarykalin, S., Serraino, G., & Uryasev, S. (2008). Value-at-risk vs conditional value-at-risk in risk management and optimization

Uryasev, S. (2000). Conditional value-at-risk: Optimization algorithms and applications, *Financial Engineering News,* 14, February,1-5.

Cone

A set which contains the ray generated by any of its points. Mathematically, a set S is a cone if the point x in S implies that αx is in S for all $\alpha \geq 0$.

Congestion System

Often used to be synonymous with queueing system because congestion refers to the inability of arriving customers to get immediate service, which is the reason behind doing queueing analyses.

See

► Queueing Theory

Conjoint Analysis

Situations are presented to subjects, with the features of the situations varied by experimental design. The subjects are asked to state their preferences among the situations, and the importance of each feature is assessed by statistical analysis.

See

► Forecasting

Conjugate Gradient Method

► Quadratic Programming

Connected Graph

A graph (or network) in which any two distinct nodes are connected by a path.

Conservation of Flow

(1) A set of flow-balance equations governing the flow of a commodity in a network that state that the difference between the amount of flow entering and leaving a node equals the supply or demand of the commodity at the node. (2) A set of equations that state that the limiting rates that units enter and leave a state or entity of a queueing system or related random process must be equal. The entities may be service facilities (stages), where the limiting number of units coming in must equal the limiting departing; balance at a state might mean, for example, that the rate at which a queueing system goes up to n customers equals the rate at which it goes down to n from above.

See

► Balance Equations
► Markov Chains
► Network Optimization
► Queueing Theory

Constrained Optimization Problem

A problem in which a function $f(X)$ is to be optimized (minimized or maximized), where the possible solutions X lie in a defined solution subspace S, which is usually determined by a set of linear and/or nonlinear constraints.

Constraint

An equation or inequality relating the variables in an optimization problem; a restriction on the permissible values of the decision variables of a given problem.

Constraint Programming

Irvin Lustig[1] and Jean-Francois Puget[2]
[1]IBM, Somers, NY, USA
[2]IBM, Valbonne, France

Introduction

Arising from research in the computer science community, constraint programming is a technique for solving optimization problems. It often is applied to difficult combinatorial optimization problems arising in configuration, sequencing, and scheduling. To apply constraint programming, users must write software that includes both a model of an optimization problem plus an algorithmic search procedure that indicates how to search for a solution.

Background

Constraint programming is often called constraint logic programming and originates in the artificial intelligence literature in the computer science community. Here, the word programming refers to computer programming. Knuth (1968) defines a computer program as "an expression of a computational method in a computer language." A computer program can be viewed as a plan of action of operations of the computer, and, hence, the common concept of a plan is shared with the origins of linear programming. With respect to constraint programming, it is a computer programming technique, with a name that is in the spirit of other programming techniques such as object-oriented programming, functional programming, and structured programming. Van Hentenryck (1999) wrote:

> The essence of constraint programming is a two-level architecture integrating a constraint and a programming component. The constraint component provides the basic

operations of the architecture and consists of a system reasoning about fundamental properties of constraint systems such as satisfiability and entailment. The constraint component is often called the constraint store, by analogy to the memory store of traditional programming languages. Operating around the constraint store is a programming-language component that specifies how to combine the basic operations, often in non-deterministic ways.

Hence, a constraint program is not a statement of a problem as in mathematical programming, but is rather a computer program that indicates a method for solving a particular problem. It is important to emphasize the two-level architecture of a constraint programming system. Because it is first and foremost a computer programming system, the system contains representations of programming variables, which are representations of memory cells in a computer that can be manipulated within the system. The first level of the constraint programming architecture allows users to state constraints over these programming variables. The second level of this architecture allows users to write a computer program that indicates how the variables should be modified so as to find values of the variables that satisfy the constraints.

The roots of constraint programming can be traced back to the work on constraint satisfaction problems in the 1970s, with the advent of arc consistency techniques (Mackworth 1977) on the one hand, and the language ALICE (Lauriere 1978) that was designed for stating and solving combinatorial problem on the other hand. In the 1980s, work in the logic programming community showed that the PROLOG language could be extended by replacing the fundamental logic programming algorithms with more powerful constraint solving algorithms. For instance, in 1980, PROLOG II used a constraint solver to solve equations and disequations on terms. This idea was further generalized in the constraint logic programming scheme and implemented in several languages (Colmerauer 1990; Jaffar and Lassez 1987; Van Hentenryck 1989). Van Hentenryck (1989) used the arc-consistency techniques developed in the constraint satisfaction problem (CSP) framework as the algorithm for the basic constraint solving. This was termed finite domain constraints.

In the 1990s, a rich area of research in constraint programming was the development of special purpose programming languages to allow people to apply the techniques of constraint programming to different classes of problems. Constraint logic programming was first proposed in the context of the programming language PROLOG, and there are many other specialized languages that have been developed that offer extended functionalities compared to traditional constraint logic programming systems. Some of these are implemented as libraries in mainstream languages, such as ILOG Solver C++ (Puget 1994) or Lisp (Puget 1992). Some others are special purpose languages, such as Oz (Smolka 1995) and Claire (Caseau and Laburthe 1995).

In the design of such languages, an axiom of their development is that they provide completeness with respect to being languages for doing computer programming. A recent innovative approach with respect to languages for constraint programming is in the design of the Optimization Programming Language (OPL) (Van Hentenryck 1999), where the language was designed with the purpose of making it easy to solve optimization problems by supporting constraint programming and mathematical programming techniques. Here, the completeness of the language for computer programming is not important. Instead, the language is designed to support the representation of optimization problems and includes the facilities to use an underlying constraint programming engine, with the ability to program a search strategy to find solutions to problems. The OPL language is not a complete programming language, but rather a language that is designed to solve optimization problems using either constraint programming or mathematical programming techniques. An advantage of OPL is that the same language is used to unify the representations of decision variables from traditional mathematical programming with programming variables from traditional constraint programming.

Constraint Satisfaction Problems

To understand the constraint programming framework, a formal definition of a constraint satisfaction problem is given next using the terminology of mathematical programming. Given a set of n decision variables $x_1, x_2, \ldots, x_n$, the set D_j of allowable values for each decision variable x_j, $j = 1, \ldots, n$, is called the

domain of the variable x_j. The domain of a decision variable can be any possible set, operating over any possible set of symbols. For example, the domain of a variable could be the even integers between 0 and 100, or the set of real numbers in the interval [1,100], or a set of people {Tom, John, Jim, Jack}. There is no restriction on the type of each decision variable, and, thus, decision variables can take on integer values, real values, set elements, or even subsets of sets.

Formally, a constraint $c(x_1, x_2,\ldots, x_n)$ is a mathematical relation, that is, a subset S of the set $D_1 \times D_2 \times \ldots \times D_n$, such that if $(x_1, x_2,\ldots,x_n) \varepsilon S$, then the constraint is said to be satisfied. Alternatively, a constraint can be defined as a mathematical function f: $D_1 \times D_2 \times \ldots D_n \rightarrow \{0,1\}$ such that $f(x_1, x_2,\ldots, x_n) = 1$ if and only if $c(x_1, x_2,\ldots, x_n)$ is satisfied. Using this functional notation, a constraint satisfaction problem (CSP) can be defined as follows:

Given n domains $D_1, D_2,\ldots, D_n$ and m constraints $f_1, f_2,\ldots, f_m$, find $x_1, x_2,\ldots, x_n$ such that

$$f_k(x_1, x_2, \ldots, x_n) = 1, \quad 1 \le k \le m$$
$$x_j \varepsilon D_j, \quad 1 \le j \le n$$

Note that this problem is only a feasibility problem, and that no objective function is defined. It is important to note here that the functions f_k do not necessarily have closed mathematical forms and can simply be defined by providing the set S described above. A solution to a CSP is simply a set of values of the variables such that the values are in the domains of the variables, and all of the constraints are satisfied.

Algorithms for Constraint Satisfaction

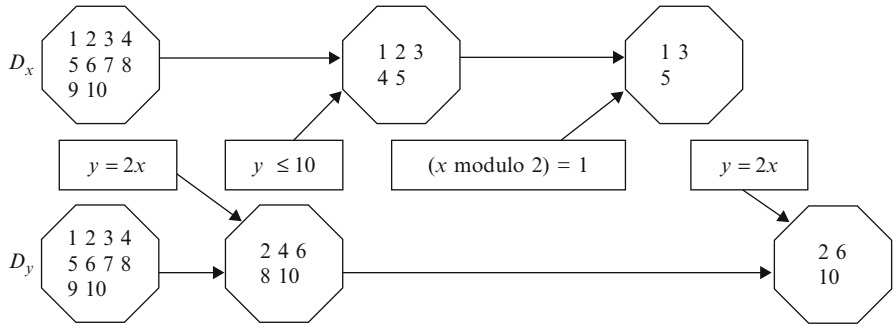
Up to now, there has been no discussion about the algorithm that a constraint programming system uses to determine solutions to constraint satisfaction problems. As mentioned earlier, a constraint programming system requires that the user programs a search strategy that indicates how the values of the variables should change so as to find values that satisfy the constraints. In OPL, there is a default search strategy that is used if the user does not program a search strategy. Most constraint programming systems require the user to program a search strategy. The first fundamental algorithm underlying a constraint programming system is given next, followed by a discussion of the methodologies used to program search.

Constraint Propagation and Domain Reduction

A constraint is defined as a mathematical function $f(x_1, x_2,\ldots, x_n)$ of the variables. Because constraint programming has its roots in computer programming, the variables can be viewed as programming language variables within a computer programming environment. Within this environment, assume there is an underlying mechanism that allows the domains of the variables to be maintained and updated. When a variable's domain is modified, the effects of this modification are then propagated to any constraint that interacts with that variable. For each constraint, a domain reduction algorithm is then programmed that modifies the domains of all the variables in that constraint, given the modification of one of the variables in that constraint. The domain reduction algorithm for a particular kind of constraint discovers inconsistencies among the domains of the variables in that constraint by removing values from the domains of the variables. If a particular variable's domain becomes empty, then it can be determined that the constraint cannot be satisfied, and an earlier choice can be undone.

This is best illustrated by the example in Fig. 1. Consider two variables x and y, where the domains of each variable are given as $D_x = \{1, 2, 3, 4, \ldots, 10\}$ and $D_y = \{1, 2, 3, 4, \ldots, 10\}$, and the single constraint $y = 2x$. For the variable y and this constraint, it is clear that y must be even and the domain of y can be changed to $D_y = \{2, 4, 6, 8, 10\}$. Now, considering the variable x, since $y \le 10$, it then follows that $x \le 5$, and the domain of x can be changed to $D_x = \{1, 2, 3, 4, 5\}$. Suppose that now a constraint is added of the form x (modulo 2) = 1. This is equivalent to the statement that x is odd. This reduces the domain of x to be $D_x = \{1, 3, 5\}$. Now, reconsidering the original constraint $y = 2x$, the values of 4 and 8 can be removed from the domain of y and obtain $D_y = \{2, 6, 10\}$.

A typical constraint programming system allows the programmer to take advantage of the existing



Constraint Programming, Fig. 1 Illustration of constraint propagation and domain reduction

propagators for built-in constraints that cause domain reductions, and to build one's own propagation and domain reduction schemes for user-defined constraints. Some systems, however, for example OPL built on top of ILOG Solver (ILOG 1999), are robust enough that large libraries of predefined constraints are provided as part of the constraint programming system, along with associated propagation and domain reduction algorithms, and it is often not necessary to create new constraints with specialized propagation and domain reduction algorithms.

Given a set of variables with their domains and a set of constraints on those variables, a constraint programming system will apply the constraint propagation and domain reduction algorithm in an iterative fashion to make the domains of each variable as small as possible, while making the entire system arc consistent. Given a constraint f_k as stated above and a variable x_j, a value $d \varepsilon D_j$ is consistent with f_k if there is at least one assignment of the variables such that $x_j = d$ and $f_k = 1$ with respect to that assignment. A constraint is then arc consistent if all of the values of all the variables involved in the constraint are consistent. A constraint system is arc consistent if all of the corresponding constraints are arc consistent. The term arc is used because the first CSPs were problems with constraints stated on pairs of variables, and this system could be viewed as a graph, with nodes corresponding to the variables and arcs corresponding to the constraints.

A number of algorithms have been developed to efficiently propagate constraints and reduce domains so as to create systems that are arc consistent. The predominant algorithm is called AC-5, developed by Van Hentenryck et al. (1992). This latter article unified the directions of the constraint satisfaction community and the logic programming community by introducing the concept of developing different algorithms for different constraints as implementations of the basic constraint propagation and domain reduction principle.

Programming Search

Given a CSP, the constraint propagation/domain reduction algorithm can be applied to reduce the domains of the variables so as to arrive at an arc consistent system. However, while this may determine if the CSP is infeasible, it does not necessarily find solutions of a CSP. To do this, one must program a search strategy. Traditionally, the search facilities provided by a constraint programming system have been based on depth-first search. The root node of the search tree contains the initial values of the variables. At each node, the user programs a *goal*, which is a strategy that breaks the problem into two (or more) parts, and decides which part should be evaluated first. A simple strategy might be to pick a variable and to try to set that variable to the different values in the variable's domain. This strategy creates a set of leaves in the search tree and creates what is called a *choice point*, with each leaf corresponding to a specific choice. The goal also orders the leaves among themselves within the choice point. In the next level of the tree, the results of the choice made at the leaf are propagated, and the domains are reduced locally in that part of the tree. This will either produce a smaller arc consistent system, or a proof that the choice made for this leaf is not possible. In this case, the system automatically backtracks to the parent and tries other leaves of that parent. The search, thus, proceeds in a depth-first

manner, until at a node low in the tree a solution is found, or until the entire tree is explored, in which case the CSP is found to be infeasible. The search strategy is enumerative, and, at each node, constraint propagation and domain reduction are used to help prune the search space.

A recent innovation in constraint programming systems is found in ILOG Solver 4.4 (ILOG 1999), where the idea of allowing the programmer to use other strategies beyond depth-first search is provided. Depth-first search has traditionally been used because in the context of computer programming, the issues regarding memory management are dramatically simplified. ILOG Solver 4.4 allows the programmer to use best first search (Nilsson 1971), limited discrepancy search (Harvey and Ginsberg 1995), depth-bounded discrepancy search (Walsh 1997), and interleaved depth-first search (Meseguer 1997). In ILOG Solver, the basic idea is that the user programs *node evaluators*, *search selectors*, and *search limits*. Node evaluators contain code that looks at each open node in the search tree and chooses one to explore next. Search selectors order the different choices within a node, and search limits allow the user to terminate the search after some global limit is reached (e.g., time, node count, etc.). With these basic constructs in place, it is then possible to easily program any search strategy that systematically searches the entire search space by choosing nodes to explore (i.e., programming node evaluators), dividing the search space at nodes (i.e., programming goals and creating choice points), and picking the choice to evaluate next within a specific node (i.e., programming search selectors). Constraint programming systems provide a framework for describing enumeration strategies for solving search problems in combinatorial optimization.

Constraint Programming and Branch and Bound

For those familiar with integer programming, the concept of search strategies should seem familiar. In fact, branch and bound, which is an enumerative search strategy, has been used to solve integer programs since the middle 1960s. Lawler and Wood (1966) present a survey, while the text by Garfinkel and Nemhauser (1972) describes branch and bound in the context of an enumerative procedure. In systems that have been developed for integer programming, users are often given the option of selecting a variable selection strategy and a node selection strategy. These are clearly equivalent to the descriptions of search electors and node evaluators described above.

There are two fundamental ways in which a constraint programming framework extends the basic branch and bound procedures. First, in a branch and bound procedure, two branches are created at each node after a variable x with a fractional value v has been chosen to branch on. The search space is then divided into two parts, by creating a choice point based on the two choices of $(x = \lfloor v \rfloor)$ and $(x \geq \lceil v \rceil)$. In the constraint programming framework, the choices that are created can be any set of constraints that divides the search space. For example, given two integer variables x_1 and x_2, a choice point could be created consisting of the three choices $(x_1 < x_2)$, $(x_1 > x_2)$, and $(x_1 = x_2)$.

The second way in which a constraint programming framework extends the basic branch and bound procedures is with respect to the variable selection strategy. In most branch and bound implementations, the variable selection strategy uses no knowledge about the problem to make the choice of variable to branch on. The integer program is treated in its matrix form, and different heuristics are used to choose the variable to branch on based on the solution of the linear programming relaxation that is solved at each node. In a constraint programming approach, the user specifies the branching strategy in terms of the formulation of the problem. Because a constraint program is a computer program, the decision variables of the problem can be treated as computer programming variables, and a strategy is programmed in the context of the problem formulation. Hence, to effectively apply constraint programming techniques, one uses problem-specific knowledge to help guide the search strategy so as to efficiently find a solution. In this way, a constraint programming system, when combined with a linear programming optimizer, can be viewed as a framework that allows users to program problem-specific branch and bound search strategies for solving mixed-integer programming problems. This capability has been available since 1996 by combining the products ILOG Solver for constraint programming and ILOG Planner for linear programming. Similar concepts also appeared in PROLOG III (Colmerauer 1990), CLP(R) (Jaffar and Lassez 1987), and CHIP (Dincbas et al. 1988).

Optimization in Constraint Programming

A constraint satisfaction problem was defined as a feasibility problem. With regard to optimization, constraint programming systems allow an objective function to be specified. Notationally, the objective function will be denoted as g: $D_1 \times D_2 \times \ldots \times D_n \rightarrow \Re$, so that at any feasible point to the CSP, the function $g(x_1, x_2, \ldots, x_n)$ can be evaluated, with the objective function to be minimized. A weakness of a constraint programming approach is that there is not necessarily a lower bound present when minimizing an objective function. This is unlike integer programming, where a lower bound exists due to the linear programming relaxation of the problem. Constraint programming systems offer two methods for optimizing problems, called standard and dichotomic search.

Standard and Dichotomic Search

The standard search procedure used is to first find a feasible solution to the CSP, while ignoring the objective function $g(x_1, x_2, v, x_n)$. Let $y_1, y_2, \ldots, y_n$ represent such a feasible point. The search space can then be pruned by adding the constraint $g(y_1, y_2, \ldots, y_n) > g(x_1, x_2, \ldots, x_n)$ to the system, and continuing the search. The constraint that is added specifies that any new feasible point must have a better objective value than the current point. As the search progresses, new points will have progressively better objective values. The procedure concludes until no feasible point is found. When this happens, the last feasible point can be taken as the optimal solution.

Dichotomic search depends on having a good lower bound L on the objective function $g(x_1, x_2, \ldots, x_n)$. Before optimizing the objective function, an initial feasible point is found, that determines an upper bound U on the objective function. A dichotomic search procedure is essentially a binary search on the objective function. The midpoint $M = (U + L)/2$ of the two bounds is computed, and a CSP is solved by taking the original constraints and adding the constraint $g(x_1, x_2, \ldots, x_n) < M$. If a new feasible point is found, then the upper bound is updated, and the search continues in the same way with a new midpoint M. If the system is found to be infeasible, then the lower bound is updated, and the search again continues with a new midpoint M. Dichotomic search is effective when the lower bound is strong, because the computation time to prove that a CSP is infeasible can often be large. The use of dichotomic search in cooperation with a linear programming solver might be effective if the linear programming representation can provide a good lower bound.

See

- Artificial Intelligence
- Branch and Bound
- Integer and Combinatorial Optimization
- Linear Programming

References

Abdennadher, S., & Frühwirth, T. (2003). *Essentials of constraint programming*. Heidelberg: Springer.

Apt, K. (2003). *Principles of constraint programming*. Cambridge, UK: University of Cambridge Press.

Caseau, Y. & Laburthe, F. (1995). *The Claire documentation* (LIENS report 96–15), *Ecole Normale Superieure*, Paris.

Colmerauer, A. (1990). An introduction to PROLOG III. *Communications of the ACM, 33*(7), 70–90.

Dincbas, M., Van Hentenryck, P., Simonis, H., Aggoun, A., Graf, T., & Berthier, F. (1988). The constraint logic programming language CHIP. *Proceedings of the International Conference on fifth generation computer systems*, Tokyo.

Garfinkel, R. S., & Nemhauser, G. L. (1972). *Integer programming*. New York: Wiley.

Harvey, W. D. & Ginsberg, M. L. (1995). Limited discrepancy search. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, Vol. 1. pp. 607–613.

ILOG (1999). *ILOG solver 4.4 users manual*. Gentilly, France: ILOG.

Jaffar, J., & Lassez, J.-L. (1987). Constraint logic programming. In *Conference Record of the Fourteenth Annual ACM Symposium on principles of programming languages*, Munich, pp. 111–119.

Knuth, D. E. (1968). *Fundamental algorithms, the art of computer programming* (2nd ed., Vol. 1). Reading, MA: Addison-Wesley.

Lauriere, J.-L. (1978). A language and a program for stating and solving combinatorial problems. *Artificial Intelligence, 10*, 29–127.

Lawler, E. L., & Wood, D. E. (1966). Branch-and-bound methods: A survey. *Operations Research, 14*, 699–719.

Mackworth, A. K. (1977). Consistency in networks of relations. *Artificial Intelligence, 8*, 99–118.

Meseguer, P. (1997). Interleaved depth-first search. In *Proceedings of the International Joint Conference on artificial intelligence (IJCAI)*, Vol. 2. pp. 1382–1387.

Nilsson, N. J. (1971). *Problem solving methods in artificial intelligence*. New York: McGraw-Hill.
Puget, J.-F. (1992). Pecos: A high level constraint programming language. *Proceedings of the 1st Singapore International Conference on intelligent systems*.
Puget, J.-F. (1994). A C++ implementation of CLP. *Proceedings of the 2nd Singapore International Conference on intelligent systems*. See also the current web site http://www.ilog.com/products/optimization/research/spicis94.pdf
Rossi, F., van Beek, P., & Walsh, T. (Eds.). (2006). *Handbook of constraint programming*. New York: Elsevier.
Smolka, G. (1995). The Oz programming model. In J. van Leeuwen (Ed.), *Computer science today*. Lecture notes in computer science (Vol. 1000, pp. 324–343). Springer-Verlag.
Van Hentenryck, P. (1989). *Constraint satisfaction in logic programming*. Cambridge, MA: MIT Press.
Van Hentenryck, P. (1999). *The OPL optimization programming language*. Cambridge, MA: MIT Press.
Van Hentenryck, P., Deville, Y., & Teng, C. M. (1992). A generic arc-consistency algorithm and its specializations. *Artificial Intelligence, 57*, 291.
Walsh, T. (1997). Depth-bounded discrepancy search. *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, Vol. 2. pp. 1388–1395.

Constraint Qualification

A condition imposed on the constraints of an optimization problem so that local minimum points will satisfy the Karush-Kuhn-Tucker conditions.

See

- ▶ Karush-Kuhn-Tucker (KKT) Conditions
- ▶ Nonlinear Programming

Construction Applications

Chad Perry
Queensland University of Technology,
Brisbane, Australia

Introduction

Due to their size and complexity, most construction projects would appear to offer a wide potential for MS/OR applications. For example, the standard critical path models of PERT, CPM and precedence diagrams are particularly successful in construction. However, apart from these models, MS/OR methods and models are not often used in construction. Schelle (1990, p. 111) summarizes, "In project management the large number of publications about operations research topics contrast to the small number of real applications."

This entry reviews three major areas of construction where MS/OR applications could occur — job estimation and tendering, project planning, and project management and control. Factors inhibiting the application of MS/OR in construction projects are discussed and possible future developments are canvassed.

Job Estimation and Tendering

Some MS/OR models have been applied to job estimation. Job estimation requires trade-offs between time and cost. Early MS/OR work assumed direct costs for each activity increased linearly with time, and therefore, used linear programming. But construction usually does not fit this assumption. Dynamic programming and integer linear programming have also been used, but the large number of variables and constraints of construction projects made them unworkable. Models based on heuristic and nonlinear curves have been found to be almost as accurate and more friendly for construction managers, and have been tried (Cusack 1985). In addition, the Line of Balance (LOB) model, originally developed for the U.S. Navy, is used to make trade-offs between alternative schedules, and a modified LOB model called Time Chainage is used in the U.K. for estimating schedules for construction of roads, bridges and other civil engineering projects (Wager and Pittard 1991).

Allied to job estimation is tendering, which must consider competitors' likely actions along with the bidder's decisions. It is a relatively more open and therefore more difficult system to model. Hence, although ARIMA and regression, plus other statistical and simulation models, have been developed to assist tendering, they have rarely been applied.

If tendering is considered from the selector's point of view, rather than a bidder's point of view, variables are not so uncertain because the selector will have certain information about all the bids.

Nevertheless, the complexity of construction projects again makes application of conventional MS/OR models difficult, especially as prior knowledge about bidders is an important choice factor. A hybrid model using linear programming, multiattribute utility, regression and expert systems seems appropriate here (Russell 1992).

Project Planning

While preparing a tender, construction managers must start to plan the project in more detail. The critical path models, integrated with cost control and reporting models, are widely used in construction for this purpose (Wager and Pittard 1991). Their application in complex construction projects has suggested theoretical extensions, for example, incorporating the stochastic relationship of cost with time. One such extension for the complex construction industry is a suite of PC programs, Construction Project Simulator (CPS), which incorporates productivity variability and external interferences to the construction process on site. It then produces bar charts, cost and resource schedules like the critical path models (Bennett and Ormerod 1984). However, most of these extensions have unrealistic data requirements and are rarely applied even if they are tried.

Modeling could be especially useful in planning tunnel construction projects. For example, Touran and Toshiyuki (1987) demonstrated a simulation model for tunnel construction and design. But model use is limited to very large projects.

Project planning usually involves more than cost minimization with constraints, for example, environmental considerations. Some MS/OR multi-objective models have provided assistance here. For example, Scott (1987) applied multi-objective valuation to roads construction, using a step-by-step procedure to evaluate all objectives, without having to assume all quantified data as being equally accurate and reliable.

Management and Control

After a project is planned, it must be managed and controlled. Linked with the project plan are straightforward accounting models. With increasing use of real time reporting, they allow closer management of costs. It is in this relatively stable field of managing and controlling the project after it has begun that conventional MS/OR models offer most promise, that is, at a tactical and relatively deterministic and repetitive level. For example, standard cost-minimization models could be applied to the management of construction equipment, to location and stocking of spare parts ware-houses, and to selecting material handling methods. In one of few actual MS/OR applications, Perry and Iliffe (1983) used a transhipment model to manage movement of sand during an airport construction project. Two other possible areas in where MS/OR models might be applied are multiple projects (where several projects are designed and built somewhat concurrently to minimize costs), and marketing.

In summary, although potential applications of MS/OR in construction appear at first glance to be plentiful, progress with actual MS/OR applications is slow. One reason for this is that risks in using unproven MS/OR models are high in commercial operations where claims resulting from mistakes can be taken to court. Moreover, each construction appears to be one-off, that is, the building is more or less different than previous ones of the constructor: at a different site with different subsurface conditions; involving different organizations and individuals with different goals; different weather; different material, labor requirements and shortages; different errors in estimates of time and cost; and different levels of interference from outside. Given this lack of standardization, MS/OR modeling has tended to move towards more general simulation models (which have large data requirements) or heuristic models. Still, MS/OR applications are few and although "computers are installed extensively throughout ... consultants and construction site offices ... their role appears to make the former manual processes more efficient rather than exploit the increased potential brought by the machine" (Brandon 1990, p. 285).

What does the future hold for MS/OR applications in the construction industry? A probable development is their increasing use in conjunction with user-friendly software on PCs. Research in the construction industry suggests that the key to successful implementation of research is a powerful intermediary like

construction managers. Developments in PC-based software such as simulations and expert systems, which assist rather than replace the experience-based knowledge of people like site managers, offer promise of more MS/OR applications, especially in the complex and expensive field of contractual disputes. These possibilities will be enhanced by interactive, three-dimensional graphical interfaces. In particular, expert systems should be used more frequently because they incorporate the existing knowledge of construction managers.

See

- ▶ Bidding Models
- ▶ CPM
- ▶ Engineering Applications
- ▶ Expert Systems
- ▶ Gantt Charts
- ▶ Linear Programming
- ▶ Multiobjective Programming
- ▶ PERT
- ▶ Project Management

References

Bennett, J., & Ormerod, R. N. (1984). Simulation applied to construction projects. *Construction Management and Economics, 2*, 225–263.

Brandon, P. S. (1990). The development of an expert system for the strategic planning of construction projects. *Construction Management and Economics, 8*, 285–300.

Cusack, M. M. (1985). A simplified approach to the planning and control of cost and project duration. *Construction Management and Economics, 3*, 183–198.

Gupta, V., Fisher, D., & Murtaza, M. (1996). A consortium sponsored knowledge-based system for managerial decision making in industrial construction. *Interfaces, 26*, 9–23, November/December.

Lewis, J. (2005). *Project planning, scheduling, & control: A hands-on guide to bringing projects in on time and on budget* (4th ed.). New York: McGraw-Hill Osborne Media.

Perry, C., & Iliffe, M. (1983). Earthmoving on construction sites. *Interfaces, 13*(1), 79–84.

Russell, J. S. (1992). Decision models for analysis and evaluation of construction contractors. *Construction Management and Economics, 10*, 185–202.

Schelle, H. (1990). Operations research and project management past, present and future. In H. Schelle & H. Reschke (Eds.), *Dimensions of project management*. Berlin: Springer-Verlag.

Scott, D. (1987). Multi-objective economic evaluation of minor roading projects. *Construction Management and Economics, 5*, 169–181.

Slowinski, R., & Weglarz, R. (Eds.). (1989). *Advances in project scheduling* (Studies in production and engineering economics, 9). Amsterdam: Elsevier.

Touran, A., & Toshiyuki, A. (1987). Simulation of tunnelling operations. *Construction Engineering and Management, 113*, 554–568.

Wager, D. M., & Pittard, S. J. (1991). *Using computers in project management*. Cambridge, UK: Construction Industry Computing Association.

Continuous-Time Markov Chain (CTMC)

A Markov process with a continuous parameter but countable state space. The stochastic process $\{X(t)\}$ has the property that, for all $s, t \geq 0$ and nonnegative integers i, j, and $x(u)$, $0 \leq u < \infty$,

$$\Pr\{X(t+s) = j | X(s) = i,\ X(u) = x(u),\ 0 \leq u < s\}$$

$$= \Pr\{X(t+s) = j | X(s) = i\}.$$

See

- ▶ Markov Chains
- ▶ Markov Processes

Control Charts

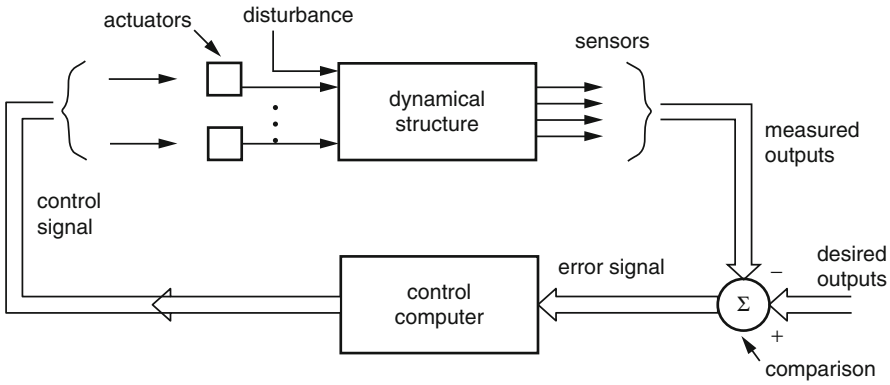
- ▶ Quality Control

Control Theory

Andre Z. Manitius
George Mason University, Fairfax, VA, USA

Introduction

Although the use of control theory is normally associated with applications in electrical and mechanical engineering, it shares much of its mathematical foundations with operations research and management science. These foundations include



Control Theory, Fig. 1 Closed loop multivariate system

differential and difference equations, stochastic processes, optimization, calculus of variations, and others.

In application, control theory is concerned with steering dynamical systems to achieve desired results. Both types of systems to be controlled and the goals of control include a wide variety of cases. Control theory is strongly related to control systems engineering, which is fundamental to many advanced technologies. In a broader sense, control theoretic concepts are applicable not just to technological systems, but also to dynamical systems encountered in biomedical, economic and social sciences. Control theory has also had a fundamental impact on many areas of applied mathematics and continues to be a rich source of research problems.

Systems to be controlled may be of various forms: they could be mechanical, electrical, chemical, thermal or other systems that exhibit dynamical behavior. Control of such systems requires that the system dynamics be well understood. This is usually accomplished by formulating and analyzing a mathematical model of the system. Physical properties of the system play an important role in establishing the mathematical model. However, once the model is established, the control theoretic considerations are independent of the exact physical nature of the system. Since different physical systems often have similar mathematical models, similar control principles are applicable to them. For example, a mechanical system of interconnected masses and springs is described by the same mathematical model as an electrical circuit of interconnected capacitors and inductors. From the control theoretic point of view, the two systems can be treated in the same way.

The control of a system is usually accomplished by providing an input signal which affects the system behavior. Physically, the input signal often changes the energy flow in the system, much like the pilot's commands change the thrust of the engines in the aircraft. The conversion of input signals into physical variables, such as the energy of the mass flow, is done by devices called actuators. System response is measured by various instruments, called sensors. The measurements, called output signals, are fed to a controller, which usually means a control computer. The controller determines the successive values of the input signals that are then passed on to the actuators. While the control computer hardware is the physical location where the control decisions are being made, the essence of the control is a control algorithm imbedded in the computer software. The development of control algorithms is often based on sophisticated mathematical theory of control and on specific models of systems under control, Fig. 1.

One of the key difficulties of control is the uncertainty about the system model and system outputs. The uncertainty has several origins. Mathematical models of systems under control are based on many simplifying assumptions and thus contain errors due to approximations. Properties or parameters of the system may change in unpredictable ways. Systems may be subject to unknown external inputs, such as, gusts of winds acting on the aircraft. Output signals provided by sensors contain sensor noise or communication channel noise. By its very nature, the control problem formulation usually includes uncertain parameters and signals. The task of control theory is to provide solutions which guarantee, whenever possible, good system performance in spite of the uncertainties.

Historical Development

The first systematic study of feedback control of steam engines by J. C. Maxwell appeared in 1868. In 1893, A.M. Lyapunov published a first paper on the stability of motion, but his work made an impact on the control theory literature only 55 years later. When the first electronic amplifiers appeared in the long-distance telephone lines after World War I, high-gain feedback coupled with high-order dynamics of amplifiers led to stability problems. In 1932, H. Nyquist provided a method of feedback stability analysis based on the frequency response. In the late 1930s, devices for controlling aircraft were introduced. World War II gave a big boost to the field of feedback control. Norbert Wiener's theory of filtering of stochastic processes, combined with the servomechanism theory, provided a unified framework for the design of control mechanisms in aircraft and ships and became what is known as classical control theory.

In the late 1950s and in the 1960s, an extensive development of control theory took place, coinciding with manned space flight and other aerospace applications, and with the advent of computers. Bellman's principle of optimality embedded in Dynamic Programming, Pontryagin's Maximum Principle of Optimal Control, and the Kalman Filter, were invented between 1956 and 1960. State-space methods of analysis, based on differential equations and matrix computations, have become the main tools of what was then named modern control theory. Control theory played a crucial role in the success of the Apollo moon-landing project in 1969. In the 1970s, substantial progress was made in the control of systems governed by partial differential equations, adaptive control and nonlinear control. The applications of control theory became very diverse, including complex material processing, bio-medical problems, and economic studies. In the 1980s, robust control theory was formulated and reached a significant level of maturity. Robust control theory has by now provided a synthesis of the classical and the modern (state-space based) control theory.

In general, research in control aims at studying the limits of performance of feedback control systems in some advanced applications. Computational tools of control have been coded in MATLAB software system and in similar software. Control hardware has been revolutionized by microprocessors and new sensor and actuator technologies, such as smart materials. Some tools of the intelligent control approach have been applied to on-board guidance and navigation systems. Anti-lock brake systems, computerized car engine control, and geographic positioning systems are a few examples of systems where the principles and tools of control theory are at work.

Mathematical Models for Control and the Identification Issue

The most commonly used mathematical control is the linear state-space model. This is a system of first-order, time-invariant, linear differential equations with inputs and outputs. Such a linear system can be written as:

$$\begin{cases} \frac{d}{dt}x(t) = \boldsymbol{A}x(t) + \boldsymbol{B}u(t) \\ y(t) = \boldsymbol{C}x(t) + \boldsymbol{D}u(t) \end{cases}$$

where $\boldsymbol{x}(t) =$ state vector, $\boldsymbol{u}(t) =$ control, $\boldsymbol{y}(t) =$ output, and $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$ are matrices of appropriate dimensions.

In practical applications, engineers often use scalar or matrix transfer functions. These are rational functions of the complex variable s arising in the Laplace transform or the variable z from the Z-transform, the latter being used for discrete-time systems. There are close relationships between state-space and transfer-function models.

In the past, many other systems have been analyzed in the control theory literature, such as nonlinear ordinary differential systems, differential equations with delay, integro-differential equations, linear and nonlinear partial differential equations, stochastic differential equations, both ordinary and partial, semigroup theory, discrete-event systems, queueing systems, Markov chains, Petri nets, neural network models, and others. In many cases, research on those systems has resulted in precise mathematical conditions under which the main paradigms of linear system theory extend to those systems.

Given an existing physical system, one of the most challenging tasks is the determination of the mathematical model for control. This is usually done in one of two ways: either the model equations are derived from physical laws and the few unknown parameters are estimated from input and output data,

or a general model form is assumed with all the parameters unknown, which then requires a more extensive parameter estimation and model validation procedure. In either case, the overall step of model determination from experimental data is called system identification. Well developed methods and computer algorithms exist to assist the control designer in this task.

The Main Ideas

Feedback is a scheme in which the control of the system is based on a concurrent measurement of the system's output. Usually, the system output is being compared to a given desired value of the output and the control is adjusted so as to steer the system output closer to the desired value. Feedback creates a directed loop linking the output to the input.

Complicated systems may have many feedback loops, either nested or intersecting one another. Feedback results in a change of a system's internal dynamics and system's input–output characteristics. A system with a properly designed feedback is capable of responding correctly to input commands even in the presence of uncertainties about the model and the external perturbations. An effective feedback reduces the effects of uncertainties, regardless of their origin. Feedback is also being used to improve stability margins, eliminate or attenuate some undesirable nonlinearities, or to shape system's bandwidth. Some systems cannot even function in a stable way without feedback. An example is the fly-by-wire fighter jet in which the feedback control loop keeps the aircraft in a stable flight envelope. The mechanism of feedback is well understood in case of linear systems. However, feedback mechanisms in nonlinear systems, especially those with many degrees of freedom, remain the subject of continued investigations. In a broader sense, the concept of feedback may be used to interpret various closed-loop interactions taking place in dynamical systems in physics, biology, economics, etc. (e.g., see Franklin et al. 2006; SIAM 1988).

Optimal Control

In many cases, the goal of control may be mathematically formulated as the optimization of a certain performance measure. The tools of optimization theory and calculus of variations have been applied to derive certain optimal control principles. For example, one of the fundamental results valid for a broad class of linear systems with a quadratic performance measure says that the optimal control is accomplished by a linear feedback based on the measurement of the internal state vector of the system. Parameters of that linear feedback are obtained by solving a quadratic equation called the Riccati equation. Another fundamental result says that the control of linear systems with bounded control function and the transition time as a performance measure is accomplished by using only the extremum values of control (a bang-bang control). Solution of optimal control problems often requires iterative numerical computations to find a control that yields the best performance.

Robust Control

Control methods have been developed to design feedback that minimizes the effect of uncertainty. Systems of this type are called robust. For example, one can design a feedback which minimizes the norm of the transfer function from unwanted disturbances to the output. Another design of that type makes the feedback system maximally insensitive to parameter variations. One of the key ideas in robust control is the use of norms in the Hardy function space H_∞, for both signals and operators (transfer functions). A close connection between the minimum H_∞ norm solutions and the solutions of certain systems of matrix Riccati equations has been discovered.

Robust control theory is well understood for linear time invariant systems, and some results have been obtained for nonlinear systems. A link has been discovered between the game-theoretic approach to control problems with uncertainty and the linear and nonlinear robust control.

Stochastic Control

Stochastic control theory involves the study of control and recursive estimation problems in which the uncertainty is modeled by random processes. One of the most significant achievements of the linear theory

was the discovery of Kalman filtering algorithms and the separation principle of the optimal stochastic control. The principle states that under certain conditions the solution of the optimal stochastic control problem combines the optimal deterministic state feedback and the optimal filter estimate of the state vector, which are obtained separately from each other.

For nonlinear systems, Markov diffusions have become the tool of analysis. Stochastic optimal control conditions lead to certain nonlinear second order partial differential equations which may have no smooth solutions satisfying appropriate initial and boundary conditions. Weak solutions and viscosity solutions have recently been used to describe solutions to such optimal control problems.

Adaptive Control

One possible remedy against the uncertainty about the system and external signals is the use of adaptive feedback mechanism. During the system operation under a regular feedback, input and output signals can be processed to produce increasingly accurate estimates of system parameters which in turn can be used to adjust the regular feedback loop. Alternatively, the step of estimating the original system can be bypassed in favor of a direct tuning of the feedback controller to minimize the error. The control system built this way contains two feedback loops, one regular but with adjustable parameters, and one that provides the adjustment mechanism. Adaptive systems are inherently nonlinear.

The main theoretical issue is the question of stability of the adaptive feedback loop; stable adaptive feedback laws for certain classes of nonlinear systems have been discovered. In contrast, bursting phenomena, oscillations, and chaos have also been found in certain simple adaptive systems. Research efforts include the finding of robust adaptive control laws and at solving stochastic adaptive control problems for systems governed by some partial differential equations.

Intelligent Control

The term intelligent control is meant to describe control which includes decision making in uncertain environments, learning, self-organization, evolution of the control laws based on adaptation to new data, and to changes in the environment. An intelligent controller may deal with situations that require deciding which variables should be controlled, which models should be used, and which control strategy should be applied at any particular stage of operation. In some situations, no precise mathematical model of the system may exist, with the only information about the process being descriptive.

Intelligent control is a blend of control theory with artificial intelligence. In contrast to mathematical control theory, that uses precisely formulated models and control laws, intelligent control relies in many cases on heuristic models and rules. It is an area of research with few established paradigms. Tools of intelligent control includes expert systems, fuzzy set theory and fuzzy control algorithms, and artificial neural networks. Examples of systems where the intelligent control may become effective are autonomous robots and vehicles, flexible manufacturing systems, and traffic control systems.

Concluding Remarks

Among the main control theory challenges are: feedback control laws for nonlinear systems with many degrees of freedom, including systems governed by nonlinear partial differential equations (e.g., control of fluid flow); adaptive and robust control of such systems; control of systems based on incomplete models with learning and intelligent decision making; and feedback mechanisms based on vision and other non-traditional sensory data (SIAM 1988).

See

- ▶ Artificial Intelligence
- ▶ Calculus of Variations
- ▶ Dynamic Programming
- ▶ Mathematical Programming
- ▶ Neural Networks
- ▶ Nonlinear Programming
- ▶ Unconstrained Optimization

References

Anderson, B. D. O., & Moore, J. B. (1990). *Optimal control*. Englewood Cliffs, NJ: Prentice Hall.

Astrom, K., & Wittenmark, B. (1989). *Adaptive control*. Reading, MA: Addison-Wesley.

Fleming, W., & Soner, M. (1994). *Controlled markov processes and viscosity solutions*. New York: Springer Verlag.

Franklin, G. F., Powell, J. D., & Emami-Naeni, A. (2006). *Feedback control of dynamic systems* (5th ed.). New Jersey: Pearson Prentice Hall.

Green, M., & Limebeer, D. J. N. (1995). *Linear robust control*. Englewood Cliffs, NJ: Prentice Hall.

Lin, C. F. (1994). *Advanced control systems design*. Englewood Cliffs, NJ: Prentice Hall.

SIAM. (1988). *Future directions in control theory: A mathematical perspective, SIAM reports on issues in the mathematical*. Philadelphia: Sciences.

Sontag, E. D. (1998). *Mathematical control theory: Deterministic finite dimensional systems* (2nd ed.). New York: Springer Verlag.

Stoorvogel, A. (1992). *The H_∞ control problem*. London: Prentice Hall International.

Zabczyk, J. (1992). *Mathematical control theory: An introduction*. Boston: Birkhauser.

Control Variates

In stochastic or Monte Carlo simulation, a variance reduction technique whereby a simulated random variable with known expectation (the control variate) is used to construct a more precise estimator by combining it (usually linearly) with another more standard estimator.

See

- ▶ Monte Carlo Methods
- ▶ Monte Carlo Simulation
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Variance Reduction Techniques in Monte Carlo Methods

Controllable Variables

In a decision problem, variables whose values are determined by the decision process and/or decision maker. Such variables are also called decision variables.

See

- ▶ Decision Maker (DM)
- ▶ Decision Problem
- ▶ Mathematical Model

Convex Combination

A weighted average of points (vectors). A convex combination of the points $x_1, \ldots, x_k$ is a point of the form $x = \alpha_1 x_1 + \ldots \alpha_k x_k$, where $\alpha_1 \geq 0, \ldots, \alpha_k \geq 0$, and $\alpha_1 + \ldots \alpha_k = 1$.

Convex Cone

A cone that is also a convex set.

Convex Function

A function that is never above its linear interpolation. Mathematically, a function $f(x)$ is a convex over a convex set S, if or any two points x_1 and x_2 in S and for any $0 \leq \alpha \leq 1$,

$$f[\alpha x_1 + (1-\alpha)x_2] \leq \alpha f(x_1) + (1-\alpha) f(x_2).$$

Convex Hull

The smallest convex set containing a given set of points S. The convex hull of a given set S is the intersection of all convex sets containing S. The convex hull of a given set of points S is the set of all convex combinations of set of points from S. If the set S is a finite set of points in a finite-dimensional space, then the convex hull is a polyhedron.

See

- ▶ Convex Set
- ▶ Polyhedron

Convex Optimization

Yurii Nesterov
Université Catholique de Louvain (UCL),
Louvain-la-Neuve, Belgium

Introduction

Optimization problems arise naturally in many domains of Operations Research. Usually they come from some design or planning procedures facing to the limits on different resources (budget, raw materials, labor, time, etc.). The required amounts of these resources become the decision variables. It is convenient to represent them by a vector $x \in R^n$.

Very often, the results of the planning procedure can be characterized by certain functions $f_i(x)$, $i = 0, \ldots, m$, called the functional components of the problem. Choosing the most important characteristic, say f_0, as the objective function, the following is the standard formulation of the constrained optimization problem:

$$f^* = \min_{x \in Q} \{ f_0(x) : \; a_i \le f_i(x) \le b_i, i = 1, \ldots, m \}, \tag{1}$$

where $Q \subseteq R^n$ is a basic feasible set, and $[a_i, b_i] \subset R$ are the target intervals for different characteristics of the decision variables.

Clearly, the formulation (1) is very natural and very important for many areas of human activity. However, it can be shown that in its general form (1) this problem is numerically unsolvable. It is one of the most important results of the *Complexity Theory* for optimization problems, developed in Nemirovski and Yudin (1983). This theory studies the abilities of numerical methods in computing an approximate solution of optimization problems. Since the possibility of computing an exact solution is extremely rare in Nonlinear Numerical Analysis, the methods are treated as iterative procedures, which generate an answer by collecting some information on the particular problem instance.

The computational efforts of such methods are measured by the number of calls of oracle, the special unit which can compute the values and differential characteristics of functional components f_i at the requested point $x \in R^n$. It is assumed that the oracle is a Black Box, meaning that no information on its structure and intermediate computational results is available. At the same time, no bounds are introduced for the computational cost of iteration and for the volume of required memory. Nevertheless, it appears that the worst-case complexity bound for generating an approximate global solution to problem (1) with accuracy $\varepsilon > 0$ is of the order

$$O\left(\frac{1}{\varepsilon^n}\right) \tag{2}$$

calls of oracle. In this lower bound, ε represents the accuracy in estimating the optimal value of the objective function, and functional components of (1) are assumed to be Lipschitz continuous. Worst-case bound means that for each method from a reasonably wide class of optimization procedures there exists a very bad problem from our problem class, for which the number of calls is at least (2). Note that this bound destroys any hope for developing a reliable method for approximating a global solution to the general problem (1). Indeed, taking very moderate values for the parameters, say $\varepsilon = 0.01$ and $n = 20$, we get a computational cost which is intractable for any computer now and in a foreseen future.

Note that the main reason for the disastrous bound (2) is the ambitious intention to approach a global solution of the general problem (1). By stepping back and reducing the goal to finding a stationary point, for the unconstrained minimization problem

$$f^* = \min_{x \in R^n} f(x), \tag{3}$$

the stationary points satisfy the Fermat condition

$$\nabla f(x) = 0, \tag{4}$$

where ∇f denotes the gradient of function f. Thus, the goal now is to find $\bar{x} \in R^n$ with

$$\| \nabla f(\bar{x}) \| \le \varepsilon. \tag{5}$$

For function with Lipschitz continuous gradient,

$$\| \nabla f(x) - \nabla f(y) \| \le L \| x - y \|, \quad x, y \in R^n, \tag{6}$$

apply the simplest Gradient Method,

$$x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k), \quad k \geq 0. \tag{7}$$

Then, after k iterations of the scheme,

$$\min_{0\leq i\leq k} \| \nabla f(x_i)\|^2 \leq \frac{L(f(x_0) - f^*)}{2(k+1)}.$$

Thus, the goal (5) can be reached in

$$O\left(\frac{1}{\varepsilon^2}\right) \tag{8}$$

iterations of method (7). As compared with (2), the estimate (8) does not depend on n. Thus, even for very large problems the goal (5) is reachable.

Since we are able to approach efficiently the stationary points of problem (3), the natural question is as follows:

What is the largest class of functions, for which the stationarity condition (4) *is a suffcient characterization of the global soution to* (3)*?*

Denoting this class of functions by $\mathcal{F}$, we could ask also for two other natural properties:

- If $f_i \in \mathcal{F}$ and $\alpha_i \geq 0$, then $\sum\limits_i \alpha_i f_i \in \mathcal{F}$.
- Any linear function belongs to $\mathcal{F}$.

Then, it can be shown, Section 2.1.1 in Nesterov (2004), that a differentiable function f belongs to $\mathcal{F}$ if and only if

$$f(y) \geq f(x) + \langle \nabla f(x), y - x\rangle, \quad x, y \in R^n, \tag{9}$$

where $\langle x, y\rangle \stackrel{\text{def}}{=} \sum\limits_{i=1}^{n} x^{(i)} y^{(i)}$. This is a definition of differentiable convex function on R^n.

This notion can be extended onto nondifferentiable functions defined onto the convex sets. A set $Q \subset R^n$ is called convex if

$$\begin{aligned} &x, y \in Q \quad \Rightarrow \\ &\quad x_\alpha \stackrel{\text{def}}{=} \alpha x + (1-\alpha) y \in Q, \ \alpha \in [0,1]. \end{aligned} \tag{10}$$

Function f is called convex if its epigraph is a convex set. That is

$$\begin{aligned} f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y), \\ x, y \in Q, \ \alpha \in [0,1]. \end{aligned} \tag{11}$$

Function f is called concave if $-f$ is convex.

Despite to the absence of usual differentiability and continuity, convex function possesses many interesting properties, especially at the interior points of its domain. At these points it is locally Lipschitz continuous and differentiable along any direction. Moreover, at any point x of its domain $\mathrm{dom} f$, there exists a special *set* of differential characteristics of this function called a subdifferential $\partial f(x) \subset R^n$. It is defined as follows:

$$f(y) \geq f(x) + \langle g_x, y - x\rangle, \quad x, y \in \mathrm{dom} f, \ g_x \in \partial f(x). \tag{12}$$

Subdifferential is a closed convex set, which is bounded for any interior point of $\mathrm{dom} f$. For differentiable functions, $\partial f(x) \equiv \{\nabla f(x)\}$. Convexity of functions is preserved by some natural operations (summation, multiplication by a positive constant, taking a maximum, etc.). All these operations are supported by corresponding operations with subdifferentials. Thus, in principal, the differential characteristics of convex functions are computable. Convex sets and convex functions are extensively studied in a special mathematical discipline called Convex Analysis, see Rockafellar (1970); Hiriart-Urruty and Lemarechal (1993).

The notion of convexity plays a central role in Operations Research and Optimization Theory. Using convex objects, we can write down the convex optimization problem:

$$f^* = \min_{x\in Q} \{ f_0(x) : \ f_i(x) \leq 0, \ i = 1, \ldots, m \}, \tag{13}$$

where Q is a closed convex set and all functions f_i, $i = 0, \ldots, m$, are convex. We will see below that this problem is generically tractable. It can be efficiently solved by different optimization methods.

Besides the convex optimization problems, there are two other important problem classes with convex structure.

Saddle point problems. In this setting, we need to find a solution of the following problem:

$$\min_{x\in Q_x} \max_{y\in Q_y} f(x, y), \tag{14}$$

where Q_x and Q_y are closed convex sets, and function $f(x, y)$ is convex in x and concave in y. For example,

we can write in this form a two-person zero-sum game. Note that optimization problem (13) is a particular case of problem (14):

$$f^* = \min_{x\in Q}\max_{y\in R^m_+}\left\{f_0(x) + \sum_{i=1}^m y^{(i)} f_i(x)\right\}.$$

For the saddle point problem (14), define a pair of primal-dual problems. Denote

$$f(x) = \max_{y\in Q_y} f(x,y), \quad \phi(y) = \min_{x\in Q_x} f(x,y).$$

Note that $f(x) \geq \phi(y)$ for all $x \in Q_x$, $y \in Q_y$. At the same time, under very mild assumptions there is zero duality gap:

$$f^* = \min_{x\in Q_x} f(x) = \max_{y\in Q_y} \phi(y).$$

Variational inequalities. Variational inequality problem (VI) is posed as follows:

$$\text{Find } x^* \in Q : \langle V(x^*), x - x^*\rangle \geq 0 \quad \forall x \in Q, \quad (15)$$

where Q is a closed convex set and $V : R^n \to R^n$. Point x^* is called the strong solution of VI. If $x_* \in Q$ and

$$\langle V(x), x - x_*\rangle \geq 0 \quad \forall x \in Q, \quad (16)$$

then x_* is called the weak solution of VI. For continuous operators, the sets of weak and strong solutions coincide. Not that the numerical schemes can approach only the set of weak solutions X_*, independently on existence of the strong ones. By definition, X_* is a closed convex set (may be empty).

The problem (15) has convex structure when the operator V is monotone:

$$\langle V(x) - V(y), x - y\rangle \geq 0 \quad \forall x, y \in Q. \quad (17)$$

Variational inequality problem with monotone operator is the most general (and most difficult) problem with convex structure. It includes, as a particular case, the saddle point problem (14).

An important example of monotone VI is the problem of finding the Nash equilibrium of a game with m players. Let $Q_i \subseteq R^{n_i}$, $i = 1, \ldots, m$, be the closed convex sets containing the feasible decision vectors of corresponding players. Assume that each player i has his own utility function $f_i(x_1, \ldots, x_m)$, which is convex in $x_i \in Q_i$, and jointly concave in all other variables $x_j \in Q_j$, $j \neq i$. The Nash equilibrium $x^* = (x_1^*, \ldots, x_m^*)$ is defined as follows:

$$x_i^* = \arg\min_{x_i\in Q_i} f_i(x_1^*, \ldots, x_i, \ldots, x_m^*), \quad i = 1, \ldots, m.$$

It can be shown that this point is a solution of corresponding VI with operator

$$V(x) = (\nabla_{x_1} f_1(x), \ldots, \nabla_{x_m} f_m(x)),$$
$$x = (x_1, \ldots, x_m) \in \prod_{i=1}^m Q_i.$$

This operator is monotone if function $\sum_{i=1}^m f_i(x)$ is convex.

From the modelling point of view, convexity is often a very natural property. Condition (10) implies that with two feasible decisions x and y, all intermediate variants x_α are feasible. Clearly, this assumption enormously facilitates the decision-making process. It is realized for a long time already, that even if the number of variables is relatively small, the problems with nonconvex or discrete feasible sets can be extremely difficult for human beings, e.g., "To be, or not to be?" Shakespeare (1602). For numerical methods, convexity is also a very favorable property.

Black-Box Optimization Methods

Nonsmooth Optimization

For explaining the main ideas of Black-Box optimization schemes, consider the simplest formulation of convex optimization problem,

$$\min_{x\in Q} f(x), \quad (18)$$

where $Q \subseteq R^n$ is a closed convex set, and f is a convex function defined on R^n. Black-box optimization methods approach the optimal solution of this problem by analyzing the answers of the oracle $(f(x_i), g_i \in \partial f(x_i))$ computed at the test points $\{x_i\}_{i=0}^\infty$. The simplest optimization strategy is

implemented in the (primal) Subgradient Method (Polyak 1967; Shor 1985):

$$x_{k+1} = \pi_Q(x_k - h_k g_k), \quad k \geq 0, \tag{19}$$

where $\pi_Q(x)$ is the Euclidean projection of x onto the convex set Q, and the apriori chosen step sized $\{h_k\}$ satisfy conditions

$$h_k > 0, \quad h_k \to 0, \quad \sum_{k=0}^{\infty} h_k = \infty.$$

Assuming that the subgradients of f are bounded on Q by constant M, and that $\| x_0 - x^* \| \leq R$, we can derive the optimal step size strategy for N-step process:

$$h_k = \frac{R}{M\sqrt{N+1}}, \quad k = 0, \ldots, N.$$

Then $\min\limits_{0 \leq k \leq N} f(x_k) - f^* \leq \frac{MR}{\sqrt{N+1}}$. Thus, in order to compute ε-solution of our problem, we need $\frac{M^2R^2}{\varepsilon^2}$ of calls of oracle. In accordance with Complexity Theory (Nemirovski and Yudin 1983), this efficiency estimate cannot be improved by the Black Box Methods working in a high-dimensional spaces (number of iterations never exceeds the dimension).

Unfortunately, the practical performance of the subgradient method coincides with its theoretical estimate, which is quite pessimistic. Therefore it is important to have numerical schemes which can accelerate on the particular problem instances. Note that by the Black Box Concept and inequality (12), after analyzing N test points, the full knowledge about the objective function is concentrated in the following inequality:

$$f(y) \geq f_N(x) \stackrel{\text{def}}{=} \max_{0 \leq i \leq N} [\, f(x_i) + \langle g_i, x - x_i \rangle], \quad x \in R^n. \tag{20}$$

Piece-wise linear function f_N is called the full model of our problem. It gives, for example, a computable lower bound for the optimal value of our problem:

$$f^* \geq f_N^* \stackrel{\text{def}}{=} \min_{x \in Q} f_N(x).$$

Note that $\hat{f}_N^* \stackrel{\text{def}}{=} \min\limits_{0 \leq k \leq N} f(x_k)$ gives us an upper bound for f^*. The models $f_N(x)$ are employed in so-called Bundle Methods, see Hiriart-Urruty and Lemarechal (1993). The most popular variant is the Level Method (Lemarechal et al. 1995):

$$Q_k = \{x \in Q : \; f_k(x) \leq \tfrac{1}{2}(\hat{f}_k^* + f_k^*)\},$$

$$x_{k+1} = \pi_{Q_k}(x_k), \quad k \geq 0.$$

The efficiency estimate for this method is the same as for the subragient scheme (the optimal one). However, its practical behavior usually is much better. Level Method can be also used for solving saddle point problems and variational inequalities.

For problem of moderate dimension, Complexity Theory provides us with lower complexity bound $O(n \ln \frac{1}{\varepsilon})$. Note that it has very weak dependence on ε. The methods which efficiency estimates depend polynomially on dimension and the logarithm of accuracy are called polynomial-time schemes.

In optimization, the methods which approach the above lower bound are based on idea of cutting planes. In accordance with (12), the optimal solution x^* satisfies the following condition:

$$\langle g_x, x - x^* \rangle \geq 0.$$

Therefore, after N iterations we know that

$$x^* \in \mathcal{L}_N \stackrel{\text{def}}{=} \{x \in Q : \; \langle g_k, x_k - x \rangle \geq 0, \; k = 0, \ldots, N\}.$$

The localization sets $\mathcal{L}_k$ can be used in different ways. The most straightforward strategy is implemented in the Method of Centers of Gravity (Newman 1960, Levin 1965):

$$x_{k+1} = \text{center\_of\_gravity}(\mathcal{L}_k), \quad k \geq 0.$$

It can be shown that this method has the optimal rate of convergence. However, its iteration is extremely expensive. An implementable version of this method is the famous Ellipsoid Method (Nemirovski and Yudin, 1983). It updates the outer

ellipsoidal approximations for the sets $\mathcal{L}_k$. For problem (18), the scheme is very simple:

$$\textbf{Initial settings}: R \geq \| x_0 - x^* \|, H_0 = R^2 \cdot I.$$

$$k\textbf{th iteration}: x_{k+1} = x_k - \frac{H_k g_k}{(n+1)\langle H_k g_k, g_k \rangle^{1/2}},$$

$$H_{k+1} = \frac{n^2}{n^2-1}\left(H_k - \frac{2H_k g_k g_k^T H_k}{(n+1)\langle H_k g_k, g_k \rangle}\right).$$

However, its efficiency estimate is not optimal: $O(n^2 \ln \frac{1}{\varepsilon})$. At this moment there exist several optimization methods with optimal efficiency estimate and reasonably small complexity of each iteration, see Nesterov (2004).

Smooth Optimization

Assume now that the objective function in problem (18) has Lipschitz-continuous gradient:

$$\| \nabla f(x) - \nabla f(y) \| \leq L \| x - y \|, \quad x, y \in Q. \quad (21)$$

The simplest scheme for solving this problem is the Primal Gradient Method:

$$x_{k+1} = \pi_Q\left(x_k - \frac{1}{L}\nabla f(x_k)\right), \quad k \geq 0.$$

Its rate of convergence is as follows: $f(x_k) - f^* \leq \frac{LR^2}{k+1}$. Thus, this scheme can compute ϵ-solution in $O(\frac{LR^2}{\varepsilon})$ iteration. Another possibility is to use Dual Gradient Method:

$$v_{k+1} = \arg\min_{v \in Q}\left\{\sum_{i=0}^{k}[f(v_i) + \langle \nabla f(v_i), v - v_i \rangle] + \frac{L}{2}\| v - v_0 \|^2\right\}, \quad k \geq 0.$$

Defining $x_k = \pi_Q\left(v_k - \frac{1}{L}\nabla f(v_k)\right)$, we get $\sum_{i=0}^{k}(f(x_k) - f^*) \leq \frac{L}{2}\| v_0 - x^* \|^2$. Finally, combining these two ideas, leads to the Fast Gradient Method (FGM):

$$v_k = \arg\min_{v \in Q}\left\{\sum_{i=0}^{k-1}\frac{i+1}{2}[f(y_i) + \langle \nabla f(y_i), v - y_i \rangle] + \frac{L}{2}\| v - x_0 \|^2\right\},$$

$$y_k = \frac{2}{k+2}v_k + \frac{k}{k+1}x_k, \quad x_{k+1} = \pi_Q(y_k - \frac{1}{L}\nabla f(y_k)), \quad k \geq 0. \quad (22)$$

It can be shown that $f(x_k) - f^* \leq \frac{2L\|x_0 - x^*\|^2}{(k+1)(k+2)}$, see Nesterov (2005). Thus, this method computes an ε-solution to problem (18) in $O(\frac{L^{1/2}R}{\varepsilon^{1/2}})$ iterations. Under assumption (21), this rate of convergence is optimal, Nemirovski and Yudin (1983). The first FGM was proposed in Nesterov (1983).

Second-Order Methods

If the second derivative of objective function is available, we can apply to problem (18) the second order schemes. Unfortunately, in this situation the classical Newton method does not allow the worst-case global complexity analysis. In order to get the full theoretical justification, we need to apply cubic regularization, Nesterov and Polyak (2006). Namely, let us assume that

$$\| \nabla^2 f(x) - \nabla^2 f(y) \| \leq K \| x - y \|, \quad x, y \in Q. \quad (23)$$

For problem (18) and (23), consider the following method:

$$x_{k+1} = \arg\min_{x \in Q}\left\{f(x_k) + \langle \nabla f(x_k), x - x_k \rangle + \frac{1}{2}\langle \nabla^2 f(x_k)(x - x_k), x - x_k \rangle + \frac{K}{6}\| x - x_k \|^3\right\}.$$

It converges as $f(x_k) - f^* \leq \frac{27K\|x_0 - x^*\|^2}{2(k+3)^2}$, see Nesterov and Polyak (2006). Using the ideas of Fast Gradient Methods, it can be accelerated up to the rate $f(x_k) - f^* \leq \frac{14K\|x_0 - x^*\|^2}{k(k+1)(k+2)}$, $k \geq 1$, see Nesterov (2008).

Structural Optimization

For Convex Optimization, black-box Complexity Theory has a hidden drawback. Indeed, in order to

apply the corresponding schemes, we need to be sure that our problem is convex (otherwise, the methods do not work). However, the only reliable way for checking convexity is the examination of its structure. If the function is constructed from convex elements by appropriate operations, we conclude that it is convex. Thus, the structure is visible at the preparatory stage. However, later it is ignored by numerical schemes.

Several systematic ways of using the structure of nonlinear convex optimization problems have been developed. We give two most important examples.

Polynomial-Time Interior-Point Methods

This theory is based on the notion of self-concordant function (Nesterov and Nemirovski, 1994).

Definition 1. *Let f be a closed convex function with open domain. It is called self-concordant* (sc) if $f \in C^3(\mathrm{dom} f)$ and

$$D^3 f(x)[h,h,h] \le 2\left(D^2 f(x)[h,h]\right)^{3/2}, \quad x \in \mathrm{dom} f,\ h \in R^n,$$

where $D^k f(x)[h,\ldots,h]$ denotes kth differential of f at x along direction h.

The central role in the analysis of sc-functions play the local norms defined by Hessians:

$$\| h \|_x = \langle \nabla^2 f(x)h, h\rangle^{1/2}, \quad \| s \|_x^* = \langle s, [\nabla^2 f(x)]^{-1} s\rangle^{1/2},$$
$$x \in \mathrm{dom}\, f,\ s,h \in R^n.$$

Define the Dikin ellipsoid $W_r(x) = \{y : \| y - x \|_x \le r\}$. Then $W_r(x) \subset \mathrm{dom} f$ for any $x \in \mathrm{dom}\, f$ and $r \in [0,1)$. Inside the Dikin ellipsoid, all Hessians are proportional. This feature facilitates the convergence analysis of the damped Newton method

$$x_0 \in \mathrm{dom} f, \quad x_{k+1} = x_k - \frac{[\nabla^2 f(x_k)]^{-1}\nabla f(x_k)}{1 + \| \nabla f(x_k) \|_{x_k}^*}, \quad k \ge 0.$$

It can be proved that all iterations of this scheme are feasible. They either decrease the value of f by an absolute constant, or converge quadratically. The region of quadratic convergence of this scheme is described by inequality $\| \nabla f(x) \|_x^* < \frac{1}{2}$. An important characteristic of the set $\mathrm{dom} f$ is the analytic center $x_f^* = \arg\min_x f(x)$. Its existence is equivalent to boundedness of f from below. Its uniqueness implies nondegeneracy of the Hessian at any feasible point.

An important class of sc-functions is formed by self-concordant barriers (scb) defined by inequality

$$\langle \nabla f(x), h\rangle^2 \le v\langle \nabla^2 f(x)h, h\rangle, \quad x \in \mathrm{dom} f,\ h \in R^n.$$

The value v is called the parameter of scb. Using such a barrier, we can solve the standard optimization problem

$$\min_{x \in Q} \langle c, x\rangle, \quad Q = \mathrm{Cl}\,(\mathrm{dom} f). \tag{24}$$

For that, we form the central trajectory $x^*(t) = \arg\min_x \{t\langle c, x\rangle + f(x)\}$, $t \ge 0$, and follow it by the Newton method. This can be done approximately by updating the points in the Euclidean neighborhood of the central path $\{x : \| tc + \nabla f(x) \|_x^* \le \frac{1}{4}\}$ using a predictor-corrector scheme. It can find an ε-solution of problem (24) in $O(v^{1/2}\ln\frac{v}{\varepsilon})$ iterations.

It can be proved that for any convex set in R^n there exists a scb with the parameter proportional to n. However, for its computation it is necessary to evaluate n-dimensional volumes. Therefore, in practice scb are constructed by analyzing the structure of functional components. Important examples of scb are as follows:

$$Q = \{y \in R^m : \langle a_i, y\rangle \le b_i,\ i = 1,\ldots,m\},$$
$$f(x) = -\sum_{i=1}^m \ln(b_i - \langle a_i, x\rangle), \quad v = m,$$
$$Q = \{X = X^T \in R^{n\times n} : X \succeq 0\},$$
$$f(X) = -\ln\det X, \quad v = n.$$

The most efficient interior-point methods are constructed for optimization problems in conic form. They allow infeasible start, long steps and eventual local quadratic convergence. In practice, the number of iterations of such schemes is often proportional to $\ln v$.

Smoothing Technique

The idea of this approach consists in approximating the nosmooth function by a smooth one, which can be efficiently minimized by FGM (22). It appears that

for functions with explicit max-representation this can be done in a systematic way (Nesterov, 2005). Assume that

$$f(x) = \max_{u \in Q_d} \{\langle Ax, u\rangle - \phi(u)\},$$

where Q_d is a convex set and ϕ is a convex function. Let us choose a prox-function d of the set Q_d (it is strongly convex with parameter one and attains its minimum at the center u_0 with $d(u_0) = 0$). Define

$$f_\mu(x) = \max_{u \in Q_d} \{\langle Ax, u\rangle - \phi(u) - \mu d(u)\}, \quad \mu \geq 0. \quad (25)$$

Then $f(x) \geq f_\mu(x) \geq f(x) - \mu D$ with $D = \max_{u \in Q_d} d(u)$. On the other hand, by Danskin Theorem, f_μ has Lipschitz continuous gradient with constant $L_\mu = \frac{1}{\mu} \| A \|^2$, where

$$\| A \| = \max_{x,u} \{\langle Ax, u\rangle : \| x \| = 1, \| u \| = 1\}$$

(norms for x and u are different). Thus, choosing $\mu = \Omega(\varepsilon)$, an ε-solution of problem (18) can be found by minimizing f_μ over Q by a fast gradient scheme. It will need at most $O(\frac{1}{\varepsilon})$ iterations instead of $O(\frac{1}{\varepsilon^2})$ iterations for a black-box method. This difference is due to the change of the structure of the oracle.

Of course, the smoothing technique is applicable only if the computation (25) can be done in a closed form. One of the most important examples is as follows:

$$f(x) = \max_{1 \leq i \leq n} x^{(i)}, \quad Q_d = \Delta_n = \{u \geq 0 : \sum_{i=1}^n u^{(i)} = 1\},$$
$$A = I, \quad \phi(u) = 0,$$

$$\| u \| = \sum_{i=1}^n |u^{(i)}|, \quad d(u) = \sum_{i=1}^n u^{(i)} \ln u^{(i)},$$
$$f_\mu(x) = \mu \ln \left(\frac{1}{n} \sum_{i=1}^n e^{x^{(i)}/\mu} \right).$$

See

- ▶ Global Optimization
- ▶ Interior-Point Methods for Conic-Linear Optimization

References

Hiriart-Urruty, J., & Lemarechal, C. (1993). *Convex analysis and minimization algorithms*. Berlin: Springer-Verlag.

Lemarechal, C., Nemirovski, A., & Nesterov, Y. (1995). New variants of bundle methods. *Mathematical Programming, 69*, 111–147.

Levin, A. (1965). One algorithm for minimization of convex functions. *Soviet Mathematics-Doklady, 160*(6), 1244–1247 (in Russian).

Nemirovski, A., & Yudin, D. (1983). *Problem complexity and method efficiency in optimization*. New York: John Wiley & Sons.

Nesterov, Y. (2004). *Introductory lectures on convex optimization*. Boston, MA: Kluwer.

Nesterov, Y. (2005). Smooth minimization of non-smooth functions. *Mathematical Programming (A), 103*(1), 127–152.

Nesterov, Y. (2008). Accelerating the cubic regularization of Newton's method on convex problems. *Mathematical Programming, 112*(1), 159–181.

Nesterov, Y., & Nemirovski, A. (1994). *Interior point polynomial methods in convex programming: Theory and applications*. Philadelphia: SIAM.

Nesterov, Y., & Polyak, B. (2006). Cubic regularization of Newton's method and its global performance. *Mathematical Programming, 108*(1), 177–205.

Nesterov, Y. (1983). A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. *Doklady AN SSSR* (translated as Soviet Math. Dokl.), **269** (3), 543–547.

Nesterov, Y. (2007). Gradient methods for minimizing composite functions. CORE DP 2007/76. Accepted by *Mathematical Programming*.

Newman, D. (1960). Location of the maximum of unimodal surfaces. *Journal of the ACM, 12*(3), 395–398.

Polyak, B. (1967). One general method for solving extremal problems. *Soviet Mathematics-Doklady, 174*(1), 33–36.

Rockafellar, R. (1970). *Convex analysis*. New York: Princeton University Press.

Shakespeare, W. (1602). The tragedie of Hamlet, prince of Denmarke.

Shor, N. (1985). Minimization methods for non-differentiable functions. Springer Ser. in Comp. Mathem. 3, Berlin: Springer-Verlag.

Convex Polyhedron

A set of points defined by the intersection of a finite number of linear equations and/or inequalities.

See

- ▶ Polyhedron

Convex Set

A set of points that contains the line segment connecting any two of its point. Mathematically, the set S is convex if for all $0 \leq \alpha \leq 1$ and for all x_1 and x_2 in S, the point $\alpha x_1 + (1-\alpha)x_2$ is also in S.

Convexity Rows

The constraints in the decomposition algorithm master problems that require solutions to be convex combinations of the extreme points of the subproblems.

See

- Dantzig-Wolfe Decomposition Algorithm

Convex-Programming Problem

A programming problem with convex objective function and convex inequality constraints. It is typically written as

$$\begin{aligned} &\text{Minimize } f(x) \\ &\text{subject to } g_i(x) \leq 0, \quad i = 1, ..., m, \end{aligned}$$

where the functions $f(x)$ and $g_i(x)$ are convex functions defined on Euclidean n-space.

See

- Convex Optimization
- Mathematical-Programming Problem
- Nonlinear Programming

CONWIP

CONstant WIP (Work in Process), corresponding to a pull-type production control system in which the number of parts in the system is kept fixed.

See

- Kanban
- Pull System

References

Spearman, M., Woodruff, D., & Hopp, W. (1990). CONWIP: A pull alternative to kanban. *International Journal of Production Research, 28*, 879–894.

Copula

A probability distribution function used to describe the dependence between random variables, which allows the joint cumulative distribution function CDF to be expressed in terms of the marginal CDFs and the copula. This representation enables the estimation of the marginals and the dependent behavior to be decoupled and the generation of the dependent random variables via the inverse transform method. Specifically, if $X_i \sim F_i$, $i = 1, \ldots, n$, where F_i are the marginal CDFs, the copula function is a mapping C: $[0,1]^n \rightarrow [0,1]$ given by (Nelsen 2010)

$$C(u_l, \ldots, u_n) = P(F(X_l) \leq u_l, \ldots, F(X_n) \leq u_n).$$

Thus, if the joint uniform random numbers $(U_1, \ldots, U_n)$ are generated according to C, the set of dependent random variates $(X_1, \ldots, X_n)$ can be generated by applying the corresponding inverse transform method to each component, i.e., $X_i = F_i^{-1}(U_i)$. The most well-known families of copulas are the Gaussian and the Archimedean.

See

- Inverse Transform Method
- Monte Carlo Methods
- Simulation of Stochastic Discrete-Event Systems
- Stochastic Input Model Selection

References

Nelsen, R. B. (2010). *An introduction to copulas* (2nd ed.). New York: Springer.

Corner Point

▶ Extreme Point

Corporate Strategy

Arnoldo C. Hax[1] and Nicolas S. Majluf[2]
[1]Massachusetts Institute of Technology, Cambridge, MA, USA
[2]Pontificia Universidad Católica de Chile, Santiago, Chile

Introduction

A formal strategic planning process distinguishes three perspectives: corporate, business, and functional. These perspectives are different both in term of the nature of the decisions they address, as well as the organizational units and managers involved in formulating and implementing the corresponding action programs generated by the strategy formation process.

The corporate level deals with the tasks that cannot be delegated downward in the organization, because they need the broadest possible scope — involving the whole firm — to be properly addressed. The business level faces those decisions that are critical to establish a sustainable competitive advantage, leading toward superior economic returns in the industry where the business competes. The functional level attempts to develop and nurture the core competencies of the firm, the capabilities that are the sources of the competitive advantages.

This article deals exclusively with corporate strategic tasks (Hax and Majluf 1996). There are three different imperatives — leadership, economic, and managerial — that are useful to characterize these tasks, depending on whether the concern is with shaping the vision of the firm, extracting the highest profitability levels, or assuring proper coordination and managerial capabilities.

The Leadership Imperative

This imperative is commonly associated with the person of the CEO, who is expected to define a vision for the firm, and communicate it in a way that generates contagious enthusiasm.

The CEO's vision provides a sense of purpose to the organization, poses a significant but yet attainable challenge, and draws the basic direction to the pursuit of that challenge. Successful organizations invariable seem to have competent leaders who are able to define and transmit a creative vision, one that generates a spirit of success. In other words, success breeds success.

Hamel and Prahalad (1989) argued that the vision of the firm should carry with it an obsession that they refer to as Strategic Intent. It implies a sizable stretch for the organization that requires leveraging resources to reach seemingly unattainable goals.

Much has been written and said about leadership including the controversy on nature or nurture — whether leaders are born or made — and on the existence of common characteristics to describe successful leaders (Schein 1992; Kotter 1988). This literature is not reviewed here. Instead the concentration is on the economic and managerial imperatives of the corporate strategic tasks. Nonetheless, the set of corporate tasks that deal with the economic and managerial imperatives are the critical instruments to imprint the vision of the firm. The leadership capabilities are expressed and made tangible through the tasks that are discussed herein (Pfeffer 1992).

The Economic Imperative

This imperative is concerned with creating value at the corporate level. The acid test is whether the businesses of the firm are benefitting from being together, or if they would be better off as separate and autonomous units. From this point of view, the essence of corporate strategy is to assure that the value of the whole firm is bigger than the sum of the contributions of its businesses as independent units.

The economic imperative involves three central issues: the definition of the businesses of the firm; the identification and exploitation of interrelationships across those businesses, and the coordination of the business activities that allow sharing assets and skills (Porter 1987; Pearson 1989).

There are eight corporate tasks that are associated with the economic imperative of corporate strategy. The first one is the Environmental Scan at the Corporate Level, which allow us to start the reflection

of the firm's competitive position by a thorough understanding of the external forces that it is facing. One of the principal objectives of strategy is to seek a proper alignment between the firm and its environment. Therefore, it seems logical to start the corporate strategic planning process with a rigorous examination of the external environment.

The seven additional tasks imply critical strategic decisions seeking the attainment of corporate competitive advantages. They are mission of the firm, business segmentation, horizontal strategy, vertical integration, corporate philosophy, strategic posture of the firm, and portfolio management. The essence of these tasks are discussed next.

1. Environmental Scan at the corporate level — Understanding the external forces impacting the firm: The Environmental Scan provides an assessment of the distinct business opportunities offered by the geographical regions in which the firm operates. It also examines the general trends of the various industrial sectors related to the portfolio of businesses of the corporation. Finally, it describes the favorable and unfavorable impacts to the firm from technological trends, supply of human resources, as well as political, social, and legal factors. The output of the Environmental Scan is the identification of key opportunities and threats resulting from the impact of external factors.
2. The mission of the firm — Choosing competitive domains and the way to compete: The mission of the firm defines the business scope — products, markets, and geographical locations — as well as the unique competencies that determine its capabilities. The level of aggregation used to express this mission statement is very broad, because of the need to encompass all the critical activities and capabilities of the corporation.

   The mission of the firm defines the overall portfolio of businesses. It selects the businesses in which the firm will enter or exit, as well as the discretionary allocation of tangible and intangible resources assigned to them. The selection of a business scope at the level of the firm is often very hard to reverse without incurring in significant or prohibitive costs. The development of unique competencies shape the corporate advantage, namely, the capabilities that will be transferred across the portfolio of businesses.

   The mission of the firm involves two of the most essential decisions of corporate strategy: selecting the businesses of the firm, and integrating the business strategies to create additional economic value. Mistakes in these two categories of decisions could be painful, because the stakes that are assigned to the resulting bets are very high indeed.
3. Business segmentation — Selecting planning and organizational focuses: The mission of the firm defines its business scope, namely the products and services it generates, the markets it serves, and the geographical locations in which it operates. The business segmentation defines the perspectives or dimensions that will be used to group these activities in a way that will be managed most effectively. It adds planning and organizational focuses which are central for both the strategic analysis and the implementation of the business strategies. This concept is of great importance in the conduct of a formal strategic planning process, since the resulting businesses are the most relevant units of analysis in that process.
4. Horizontal strategy — Pursuing synergistic linkages across business units: One could argue that horizontal strategies are the primary sources for corporate advantage of a diversified firm. It is through the detection and realization of the existing synergy across the various businesses that significant additional economic value can be created. The value chain is the basic framework that is used to detect opportunities for sharing resources and activities across businesses (Porter 1985). The resulting degree of linkages among businesses determines their relative autonomy and independence.

   The mission of the firm defines the business scope; business segmentation organizes the businesses into planning and managerial units; horizontal strategies determines their degree of interdependence. Consequently, these tasks are highly linked. Moreover, the mission of the firm also defines the current and future corporate core competencies, which are the basis that supports the relationship among the various businesses, and the role to be played by horizontal strategy.
5. Vertical integration — Defining the boundaries of the firm: Vertical integration determines the breadth

of the value chain, as well as the intensity of each of the activities performed internally by the firm. It specifies the firm's boundaries, and establishes the relationship of the firm with its primary outside constituencies — suppliers. distributors, and customers.

The major benefits of vertical integration are realized through: cost reductions from economies of scale and scope; creation of defensive market power against suppliers and clients; and creation of offensive market power to profit from new business opportunities. The main deterrents of vertical integration are: diseconomies of scale from increases in overhead and capital investments; loss of flexibility; and administrative penalties stemming from more complex managerial activities (Stuckey and White 1993; Harrigan 1985; Walker 1988; Teece 1987).

6. Corporate philosophy — Defining the relationship between the firm and its stakeholders: The corporate philosophy provides a unifying theme and a statement of basic principles for the organization. First, it addresses the relationship between the firm and its employees, customers, suppliers, communities, and shareholders. Second, it specifies broad objectives for the firm's growth and profitability. Third, it defines the basic corporate policies; and finally, it comments on issues of ethics, beliefs, and rules of personal and corporate conduct.

   The corporate philosophy is the task that is most closely related to the leadership imperative, insofar as bringing a capability to articulate key elements of the CEO's vision.

7. Strategic posture of the firm — identifying the strategic thrusts, and corporate performance objectives: The strategic posture of the firm is a set of pragmatic requirements developed at the corporate level to guide the formulation of corporate, business, and functional strategies. The strategic thrusts characterize the strategic agenda of the firm. They identify all of the key strategic issues, and signal the organizational units responsible to respond to them. The corporate performance objectives define the key indicators used to evaluate the managerial results, and assign numerical targets as an expression of the strategic intent of the firm. The strategic posture captures the outputs of all of the previous tasks and use them as challenges to be recognized and dealt with in terms of action-driven issues.
8. Portfolio management — Assigning priorities for resource allocation and identifying opportunities for diversification and divestment: Portfolio management and resource allocation have always been recognized as responsibilities that reside squarely at the corporate level. As noted above, the development of core competencies shared by the various businesses of the firm constitute a critical source of corporate advantage. Those competencies are borne from resources that the firm should be able to nurture and deploy effectively, including: physical assets, like plant and equipment; intangible assets, like highly-recognized brands; and capabilities, like skills associated with product design and development.

   The heart of an effective resource allocation process is the capacity to create economic value. Sometimes, this value emerges from internal activities of the firm, other times it is acquired from external sources through mergers, acquisitions, joint ventures, and other forms of alliances. Even, on occasions, value can be created by divesting businesses that are not earning their cost of capital, i.e., they are destroying instead of adding value to the firm. Portfolio management deals with all of these critical issues.

   In the 1980s, most developed economies faced periods of stagnation which have forced firms to implement drastic restructuring policies. Restructuring leads to the realignment of physical assets (including divestment), human resources, and organizational boundaries of the various businesses with the intent of reshaping their structure and performance. Restructuring decisions are also part of portfolio management (Donaldson 1994).

The Managerial Imperative

This imperative is the major determinant for a successful implementation of corporate strategy. It involves two additional important corporate tasks: the design of the firm managerial infrastructure, and the management of its key personnel.

9. Managerial infrastructure — Designing and adjusting the organizational structure, managerial processes,

and systems in consonance with the culture of the firm to facilitate the implementation of strategy: Organizational structure and administrative systems constitute the managerial infrastructure of the firm. An effective managerial infrastructure is critical for the successful implementation of the strategies of the firm. Its ultimate objective is the development of corporate values, managerial capabilities, organizational responsibilities, and managerial processes to create a self-sustaining set of rules that allow the decentralization of the activities of the firm.

The term organizational architecture is commonly used to designate the design efforts that produce an alignment between the environment, the organizational resources, the culture of the firm, and its strategy (Nadler et al. 1992).

10. Human resources management of key personnel — selection, development, appraisal, rewards, and promotion: Regardless how large a corporation is, it will be always managed by a few key individuals. Percy Barnevik, the CEO of Asea Brown-Boveri (ABB), a successful global company, stated that one of ABB's biggest priority and crucial bottleneck is to create global managers. He immediately added, however, that a global company does not need thousands of them. At ABB, five hundred out of a total of fifteen thousand managers are enough to make ABB work well (Taylor 1991).

    Tom MacAvoy, the former President of Corning Glass-Works, used to talk, in a rather colorful way, about the need for one hundred centurions to run an organization. These are huge corporations, with operations in over one hundred countries. When it comes to identify the key personnel they need, the numbers are surprisingly small; yet, the process of identifying, developing, promoting, rewarding, and retaining them, is one of the toughest challenges that an organization faces.

The Fundamental Elements in the Definition of Corporate Strategy

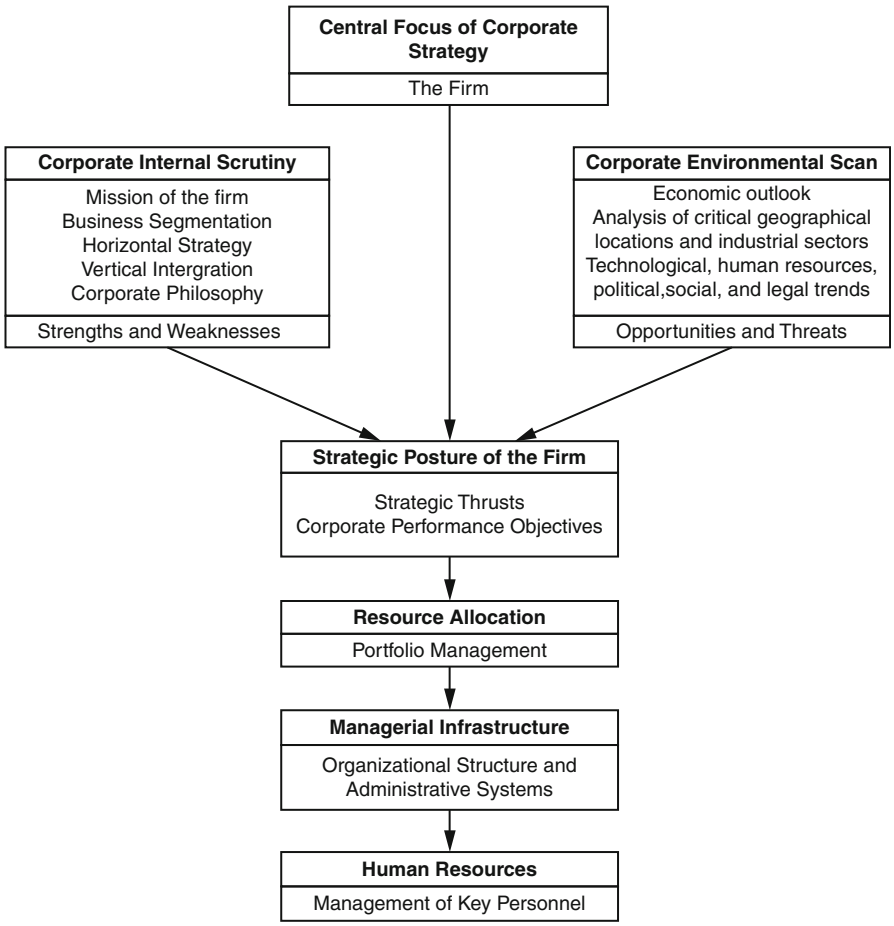
The corporate strategic tasks can be organized in a strategic planning framework, "The Fundamental Elements in the Definition of Corporate Strategy: The Ten Tasks" (Fig. 1).

The first element of the framework — The Central Focus of Corporate Strategy — consists in identifying the entity that is going to be part of the corporate strategic analysis. As opposed to the case of business strategy, where the unit of analysis is the Strategic Business Unit (SBU), corporate strategy can be applied at different levels in a large diversified organization. The amplest possible scope is the firm as a whole. There are circumstances, however, under which the scope of the analysis to a sector, group, or division of a given organization should be narrowed. These entities should encompass a number of different business units to be the subject of a meaningful corporate strategic analysis.

The next two elements of the framework are Corporate Environmental Scan and Corporate Internal Scrutiny. But, before addressing their collective tasks, it is important to note that throughout the corporate strategic analysis, existing conditions are contrasted with future ones. Thus, an underlying time frame is required to be spelled out at the beginning of the planning process.

In the case of the Corporate Environmental Scan, there are two different treatments of the future. When dealing with completely uncontrollable factors, there is a need to forecast their most likely trends to be able to understand their potential impacts. There are situations, however, in which the corporation would like to influence future events, especially when it can exercise some degree of control that will allow the future to be shaped to an advantage. By contrast, in all of the tasks that are part of the Internal Scrutiny, the future represents a state being directed at through a set of controllable decisions.

The Corporate Environmental Scan should be conducted first in the planning process, because it serves to frame the impacts resulting from the external environment. It has also the important role of transferring a common set of assumptions to the various businesses and functional managers of the firm, to serve as inputs in their own strategic planning efforts. It gives a sense of uniformity to the strategic planning thinking across all the key organizational units of the firm. This task culminates with the recognition of opportunities — the favorable impacts of the external environment which the corporation would like to seize — and threats — the unfavorable impacts which the corporation would like to neutralize.



Corporate Strategy, Fig. 1 The fundamental elements in the definition of corporate strategy: the ten tasks

The Corporate Internal Scrutiny captures the key actions and decisions the corporation has to address to gain a competitive position that is in line with the challenges generated by the external environment, and conducive to the development of a sustainable corporate advantage. This advantage is transferable to the various business units of the firm, and enhances its resources and capabilities. The tasks which are part of the Internal Scrutiny in our framework are:

- Mission of the Firm
- Business Segmentation
- Horizontal Strategy
- Vertical Integration
- Corporate Philosophy

In all of these decisions, the current state is contrasted with a desirable future one. The process then proceeds to define the challenges those changes generate for the formulation of corporate strategy. The Internal Scrutiny concludes with an overall statement of corporate strengths that the firm wishes to maintain and reinforce, as well as a statement of corporate weaknesses that the firm wishes to correct or eliminate.

The Corporate Environmental Scan and the Corporate Internal Scrutiny provide the basic inputs that will define the Strategic Posture of the firm. This task serves as a synthesis of the analysis conducted so far, and captures the strategic agenda of the firm. The strategic thrusts are a powerful expression of all of the issues that, from the perspective of the firm, need to be addressed to come out with an integrative strategy. The Corporate Performance Objectives define the key indicators that will be used to detect the operational and strategic effectiveness of the firm. The Strategic Posture is the essence of the formulation of the corporate strategy, and as such, it is a task that should receive the utmost attention. When properly conducted, the firm is able to frame the activities, responsibilities, and performance measurements that are critical for its superior strategic position.

The subsequent task, Resource Allocation and Portfolio Management, permits to backup the strategic actions implicit in the Strategic Posture of the firm with the necessary resources needed for their deployment. This leads to the realm of strategy implementation. These implementation efforts are reinforced strongly by the remaining two corporate tasks: Managerial Infrastructure and Human Resources Management of Key Personnel.

See

- ▶ Computational Organization Theory
- ▶ Organization

References

Donaldson, G. (1994). *Corporate restructuring, managing the change process from within*. Boston: Harvard Business School Press.

Hamel, G., & Prahalad, C. K. (1989). Strategic intent. *Harvard Business Review, 67*(3), 63–76.

Harrigan, K. R. (1985). *Strategic flexibility: A management guide for changing times*. Lexington, MA: Lexington Books.

Hax, A. C., & Majluf, N. S. (1996). *The strategy concept and process: A pragmatic approach* (2nd ed.). Englewood Cliffs, NJ: Prentice Hall.

Kotter, J. P. (1988). *The leadership factor*. New York: Free Press.

Nadler, D. A., Gerstein, M. S., Shaw, R. B., & Associates. (1992). *Organizational architecture: Designs for changing organizations*. San Francisco: Josses-Bass.

Pearson, A. E. (1989). Six basics for general managers. *Harvard Business Review, 67*(4), 94–101.

Pfeffer, J. (1992). *Managing with power: Politics and influence in organizations*. Boston: Harvard Business School Press.

Porter, M. E. (1985). *Competitive advantage*. New York: Free Press.

Porter, M. E. (1987). From competitive advantage to corporate strategy. *Harvard Business Review, 65*(3), 43–59.

Schein, E. E. (1992). *Organizational culture and leadership* (2nd ed.). San Francisco: Josses-Bass.

Stuckey, J., & White, D. (1993). When and when not to vertically integrate. *Sloan Management Review, 34*(3), 71–83.

Taylor, W. (1991). The logic of global business: An interview with ABB's Percy barnevik. *Harvard Business Review, 69*(2), 90–105.

Teece, D. J. (1987). Profiting from technological innovations: Implications for integration, collaboration, licensing, and public policy. In D. J. Teece (Ed.), *The competitive challenge: Strategies for industrial innovations and renewal*. Cambridge, MA: Ballinger Publishing.

Walker, G. (1988). Strategic sourcing, vertical integration and transaction costs. *Interfaces, 19*(3), 62–73.

Cost Analysis

Stephen J. Balut[1] and Thomas R. Gulledge[2]
[1]Institute for Defense Analyses, Alexandria, VA, USA
[2]George Mason University, Fairfax, VA, USA

Introduction

Cost analysis is the process of estimating the individual and comparative costs of alternative ways of accomplishing an objective. The goal is not to forecast precisely accurate costs, but rather to reveal the extent to which one alternative costs more or less than another. A cost analysis is often conducted in conjunction with an effectiveness analysis to aid in the selection of one alternative over others.

Evolution

Cost analysis emerged as part of a broader initiative in the late 1940s and early 1950s to apply economic principles to the decision making process of the U.S. Department of Defense (DoD). A confluence of events following World War II resulted in a dramatic and enduring change in the way resource allocation decisions were made in public organizations. The development and evolution of cost-effectiveness analysis and cost analysis occurred nearly simultaneously and are closely related. Both types of analysis make use of operations research methods.

Operations research was invented and applied mainly by civilian scientists in support of the war effort. From its inception, operations research sought to "use scientific methods to get the most out of available resources" (Quade 1971). Immediately following the war, many of these scientists were retained by the Military Departments to apply newly developed quantitative methods to aid defense decisions. The forerunners of the RAND Corporation, the Institute for Defense Analyses (IDA), and the Center for Naval Analyses (CNA) were formed during this period.

After the war, separation of military responsibilities between the U.S. Armed Services broke down as a consequence of the rapid development of military technology and the different character of the military

threat (Smale 1967). The Services began competing for missions and disputes were settled via approval of budgets for new weapon systems. Competing systems were considered on the basis of cost-effectiveness. When equally effective weapon systems were compared, those estimated to cost the least won funding approvals. The analytical procedure applied to such decisions was first named weapon systems analysis, later shortened to systems analysis. The first documented systems analysis was accomplished in 1949 by the RAND Corporation and compared the B-52 to a turbo-prop bomber. The use of dollar costs as a proxy for real costs changed the basic systems analysis question from "Which weapon system is best for the job?" to "Given a fixed budget, which weapon system is most cost effective?" (Smale 1967; Novick 1988).

The birth of cost analysis as a separate activity occurred in the early 1950s and is attributed to Novick (1988), a cost analyst with the RAND Corporation. Novick pioneered weapon system cost analysis and is referred to as the father of cost analysis. Novick and his group at RAND are attributed with development of the fundamental building blocks of cost analysis. These include separation of total costs into cost elements, separation of one-time and recurring costs, development of cost estimating relationships, and development of conceptual costs or order-of-magnitude estimates used to compare future system proposals. Novick's group went on to invent parametric cost estimating, incremental costing, and "Total Force Costing" (Novick 1988; Hough 1989).

In the early 1960s, the Department of Defense established and implemented a centralized resource allocation process called the Planning, Programming and Budgeting System (PPBS). Under this system, future defense resources were allocated to missions in a systematic, rational manner using cost-effectiveness as the decision criterion. In 1961, a Systems Analysis Office was established within the Office of the Secretary of Defense (OSD) to help implement this new resource allocation procedure. In 1965, a Cost Analysis Division was established within the office of the Assistant Secretary of Defense, Systems Analysis. With this act, cost analysis gained a primary role in the examination of alternative force structures at the OSD level. Also in 1965, the PPBS system was extended to all federal agencies by President Lyndon Johnson.

The next few decades brought initiatives that strengthened the cost analysis capabilities of the DoD. The military departments established cost analysis offices at headquarters and major commands and staffed them, at least in part, with people trained and experienced in the methods of operations research. The DoD initiated systematic collection of cost information from defense contractors to provide defense cost analysts with records of cost experiences on major weapon system acquisitions. These records formed the bases of estimates of the costs of proposed systems at acquisition milestone decision points, strengthened the DoD's position during contract negotiations, and provided for DoD tracking of negotiated costs. In 1971, Deputy Secretary of Defense Packard instituted defense acquisition reforms that included establishment of the DoD Cost Analysis Improvement Group (Hough 1989), the requirement for independent parametric estimates for new systems acquisitions, formalization of cost analysis reviews at milestone decision points, and requirements for the military departments to improve their cost-estimating capabilities. As part of the Packard reform, cost was elevated to a principal design parameter with implementation of the "Design to Cost" initiative (Hough 1989). Ten years later, in 1981, Deputy Secretary of Defense Carlucci placed further demands on the DoD's cost analysis capabilities. He instituted the practice of "Multi-Year Procurement" based on benefit/risk analyses, "Budget to Most Likely or Expected Cost," budgeting more realistically for inflation, the use of economic production rates, the requirement to forecast business base at defense contractors' plants, increased efforts to quantify cost risk and uncertainty, and provision of greater incentives on design-to-cost goals by tying award fees to actual costs achieved in production.

Throughout the 1970s and 1980s, the practice of cost analysis continued to expand mainly in the public sector. The US government's cost analysis organizations grew in size by drawing people skilled in engineering, economics, operations research, accounting, mathematics, statistics, business, and related fields. Several focused educational programs were initiated to support this budding profession at military universities, including the Air Force Institute of Technology, the Naval Postgraduate School, and the Defense Systems Management College.

The 1990s brought a surge of activity in cost analysis with institutionalization of a Cost and Operational Effectiveness Analysis (COEA) as an integral part of the defense acquisition process. COEAs are required to be conducted and presented to the Defense Acquisition Executive at major milestone in the acquisition of a major weapon system.

Around the turn of the century, the DoD established a preference for the use of evolutionary acquisition strategies (DoD D5000.01) that promise to speed the delivery of advanced capabilities to warfighters while also providing follow-on improvements in capabilities as planned technological advances are achieved. Adoption of this approach provides cost analysts with the challenge of estimating the costs of systems that embody ultimate capabilities that cannot be fully defined at the beginning of the acquisition program.

Methods

Cost analysis is a sequential process: first identification, then measurement, and finally evaluation of alternatives. This involves the structuring and analysis of resource alternatives in a full planning context. In the case of defense, the size of the U.S. defense budget limits the dollars available to provide for the national defense. Monies spent on one mission/capability/weapon system are not available to spend on another. "Therefore, properly constructed cost estimates and cost analyses are essential because an accurate assessment of the cost of individual programs is the first necessary step towards understanding the comparative benefits of alternative programs and capabilities" (Smale 1967).

Economic costs are benefits lost and are often referred to as alternative costs or opportunity costs (Fisher 1970). An estimate of the economic cost of one choice, decision or alternative, within this context, is an estimate of the benefits that could otherwise have been obtained by choosing the best of the remaining alternatives. When constructed in this way, costs have the same dimension as benefits, and direct comparison is possible.

The following cost analysis concepts are briefly described here: the Work Breakdown Structure (WBS), Estimating Relationships (ER), and Cost Progress Curves. The treatment is not comprehensive in any sense and is provided to give those completely unfamiliar with the methods of cost analysis an idea of what is involved.

Work Breakdown Structure — Cost analysts break complex systems down into pieces before attempting to estimate their costs. A notion fundamental to this process is the Work Breakdown Structure (WBS) (U.S. Air Force Material Command 1993). The basic concept of a WBS is to represent an aircraft system, for example, as a hierarchical tree composed of hardware, software, facilities, data, services, and other work tasks. This tree completely defines the product and the work to be accomplished. It relates elements of work to each other and to the end product. Cost analysts usually estimate total systems costs as the sum of the costs of the individual elements of the WBS.

Estimating Relationships — Another tool that is fundamental to cost analysis is the estimating relationship (ER). In a broad sense, estimating relationships are transformation devices which permit cost analysts to go from basic inputs (e.g. descriptive information for some future weapon system) to estimates of the cost of output-oriented packages of military capability (Fisher 1970). More specifically, ERs are analytic devices that relate various categories of cost (e.g. dollars or physical units) to explanatory variables referred to as cost drivers. While taking many different forms, ERs are usually mathematical functions derived from empirical data using statistical analyses.

Cost Progress Curves — The basic notion of a learning curve is that, as a work procedure (e.g. sequence of steps/activities) is repeated, the person performing the procedure normally becomes better or more efficient at performing the procedure. The reduction in time or cost to perform the procedure is commonly attributed to learning. Cost analysts, who are more interested in reductions in cost, refer to this phenomenon as cost progress rather than learning.

The theory of cost progress curves states that as the total quantity of units (e.g. aircraft, wings, or fuselages) produced doubles, the cost per unit declines by some constant percentage. Wright (1936) empirically demonstrated the principle (Asher 1956). The standard mathematical model is a power function that relates manufacturing labor hours required to

produce a particular unit to the cumulative number of units produced. The functional form is simply:

$$C = aQ^b$$

where C is the number of hours required to produce unit Q, a is the labor hours required to produce the first unit, and b is a parameter that measures the amount of cost progress reflected in the data used to estimate the model parameters. The form is a hyperbolic function that is linear in logarithmic space. The characteristic of linearity in logarithmic space and the ease of application account for the general acceptance and popularity of the cost progress curve among cost analysts. The cost progress curve is applied widely by defense cost analysts when estimating the costs of alternative force sizes and compositions.

Professional Organizations

As cost analysis evolved over the past few decades, a number of professional organizations were formed to further advance cost analysis and related professional activities. The Cost-Effectiveness Technical Section of the Operations Research Society of America (now the Institute for Operations Research and the Management Sciences–INFORMS) was formed in 1956 to provide for the exchange of experiences in conducting such analyses. This organization has since changed its name to the Military Application Section (MAS) of INFORMS.

The National Estimating Society (NES) was formed in 1978. This organization's focus was on cost estimating from the perspective of the private sector. The formation of the Institute of Cost Analysis (ICA) in 1981 was referred to as the most significant event of the decade for DoD cost analysts (Hough 1989). ICA was dedicated to the furtherance of cost analysis in the public and private sectors. Both ICA and NES established programs under which the technical competence of members were certified, leading to a designation of Certified Cost Analyst or Certified Cost Estimator. ICA and NES subsequently merged to form the Society of Cost Estimating and Analysis (SCEA). SCEA continues the certification process by conferring the "Certified Cost Estimator/Analyst" designation to those who pass a qualifying examination.

See

- ▶ Center for Naval Analyses
- ▶ Cost-Effectiveness Analysis
- ▶ RAND Corporation

References

Asher, H. (1956). *Cost-quantity relationships in the airframe industry, R-291*. Santa Monica, CA: The RAND Corporation.

DoD Directive 5000.01 (2003). *The defense acquisition system*. Washington, DC: Department of Defense.

Fisher, G. H. (1970). *Cost considerations in systems analysis, R-490-ASD*. Santa Monica, CA: The RAND Corporation.

Hough, P. G. (1989). *Birth of a profession: Four decades of military cost analysis*. Santa Monica, CA: The RAND Corporation.

Novick, D. (1988). *Beginnings of military cost analysis: 1950–1961, P-7425*. Santa Monica, CA: The RAND Corporation.

Quade, E. S. (1971). *A history of cost-effectiveness analysis, Paper P-4557*. Santa Monica, CA: The RAND Corporation.

Smale, G. F. (1967). *A commentary on defense management*. Washington, DC: Industrial College of the Armed Forces.

U.S. Air Force Materiel Command (1993). *Work breakdown structures for defense material items*. Military Standard 881B.

Wright, T. P. (1936). Factors affecting the cost of air-planes. *Journal of Aeronautical Sciences, 3*, 122–128.

Cost Coefficient

In a linear programming problem, the generic name given to the objective function coefficients.

Cost Range

- ▶ Ranging
- ▶ Sensitivity Analysis

Cost Row

The row in a simplex tableau that contains the reduced costs of the associated feasible bases.

See

- ▶ Simplex Method (Algorithm)

Cost Slope

The rate of cost change per unit of time duration of a project's work item.

See

► Network Planning

Cost Vector

In a linear-programming problem, a row vector $\boldsymbol{c}$ whose components are the objective function coefficients of the problem.

See

► Cost Row

Cost-Effectiveness Analysis

Norman Keith Womer
University of Missouri-St Louis, St. Louis, MO, USA

Introduction

Cost effectiveness analysis is a practical way of assessing the usefulness of public projects. The history of the subject can be traced to Dupuit's classic 1844 paper, "On the Measurement of the Utility of Public Works." The technique has been a mainstay of the Army Corps of Engineers since 1902. Recent variations of the technique have been labeled cost effectiveness analysis, cost benefit analysis, systems analysis, or merely analysis. It has been extensively applied to projects in defense, transportation, irrigation, waterways, and housing. Cost effectiveness analysis is required by law and regulation throughout the federal government to decide among certain alternative policies and projects. It has been recently required in federal regulations designed to protect human health, safety, or the environment. Despite this fact, the practice of cost effectiveness analysis is subject to criticism. Robert Dorfman (1996) declared, "Three prominent shortcomings of benefit-cost analysis as currently practiced are (1) it does not identify the population segments whom the proposed measure benefits or harms, (2) it attempts to reduce all comparisons to a single dimension, generally dollars and cents, and (3) it conceals the degree of inaccuracy or uncertainty in its estimates."

Cost effectiveness analysis (CEA) is the process of using theory, data, and models to examine a problem's relevant objectives and alternative means of achieving them. It is used to compare the costs, benefits, and risks of alternative solutions to a problem and to assist decision makers in choosing among them. The differences between cost effectiveness analysis and the discipline of operations research itself are subtle and, in some treatments, merely a matter of emphasis (see the discussion in Quade 1971). The convention adopted here is that operations research is a body of knowledge that includes all of the tools and methods that might be used in any study, while cost effectiveness analysis is a particular application of models and methods to a choice problem.

Sometimes CEA is portrayed as the combination of the difficult problem of measuring effectiveness with the rather mundane problem of cost estimation. In fact, cost measurement is an important issue. Cost effectiveness analysis provides a tool for effective resource allocation only when all the resource implications associated with each alternative — both direct and indirect — are included in the analysis. The opportunity cost of a proposed allocation of resources is the value of those resources in their best alternative use. The very concept of opportunity cost therefore requires knowledge of the goals and objectives, measures of effectiveness, the other alternatives and constraints of the organization. That is, to employ this basic concept of cost, a careful analysis of the problem must be accomplished.

Therefore, CEA must focus on the process of modeling both cost and effectiveness to develop relevant measures that shed light on the problem under study. Ultimately, CEA consists of methods for evaluating vectors of measures. In the process, CEA must grapple with issues like the scale of operations, risk, uncertainty, timing, and actions of other players.

The Role of Models

Figure 1, adapted from Quade (1971), portrays the elements of CEA. Models are used in CEA to aid in the evaluation of alternatives. These models often take the form of equations that relate the physical description of alternative systems to various impacts of their production and use. The models may concern the acquisition of the systems, their operation, or various circumstances associated with applying the system in an environment.

There are many assumptions in any analysis. One important class of assumptions that is often left unstated in CEAs concerns the behavior of key players in the process. Traditionally, CEAs have been based on rather mechanical models that relate a system's physical characteristics (e.g., weight and speed) to production cost. Any reference to behavior has often been confined to vague statements about efficiency. In fact, costs and benefits result only from actions. Thus, the motivation to act is an important part of modeling costs and benefits. Unfortunately, these behavioral assumptions are often not stated explicitly. Instead, they are frequently imbedded in detailed computer simulations that attempt to emulate the simultaneous operation of complex systems in realistic environments.

Incommensurable Impacts

The output of a suite of models may be a rather long list of measured system impacts. Some of the system impacts are measurable in units of effectiveness or costs, while others are external to our frame of reference. Generally, each of the impacts will be measurable in units that are unique to that impact, for example, number of lives lost, replacement cost of lost equipment, number of minutes of error free transmission accomplished, etc. Choice requires not only the objective consideration of the measurable impacts, but also the consideration of the often immeasurable externalities. As a result, it is important that the analyst carefully report both impact measures and their accuracy and those impacts that remain unmeasured. Choice also requires the explicit use of a criterion that evaluates the impacts and their relation to the choice problem at hand.

The Analyst and the Decision Maker

In doing analysis, the first and most important issue is to understand the decision maker's problem. Answering the question "What is the problem?" often requires understanding both the organization for which the analysis is performed and the physical system or structural change that is under study. The problem may be stated in different forms at different points in time and at different levels in the organization. Thus, understanding the problem requires understanding the objectives of the entire organization.

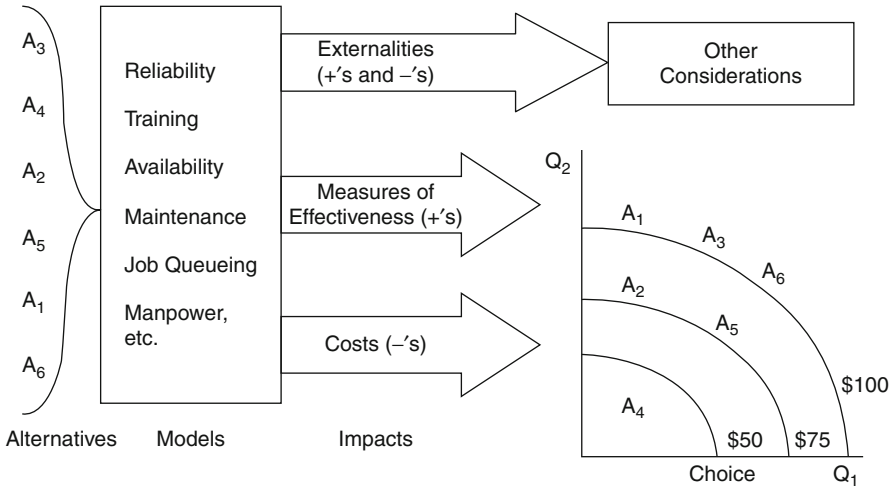
For example, consider the problem of analyzing the cost of a mission currently assigned to an aircraft system. What is the problem? Some candidates are:

- Should the existing system be replaced?
- What design should be chosen?
- Who should produce the system?
- How should the mission be performed?
- Is the mission affordable?

Often analysis is done with reference to one of these problems and then later the same study is applied to a different problem. Clearly, the alternatives, the risks, the objectives, and the cost are not independent of the problem being addressed.

Whose problem is this? It is the analyst who must choose the techniques, collect the data, model the processes, and measure the costs and outputs. It is the analyst who must justify the choices made in the particular context of the problem being addressed. Thus, it is the analyst who must be able to answer the question, "What is the Problem?"

If the role of the analyst is so large, what is left for the decision maker? The decision maker must also understand the problem and judge the value of the analysis. The decision maker must examine the completeness of the alternatives, evaluate the assumptions, examine the measurement of the impacts, and determine if risks are adequately addressed. All of these tasks are important. But the most important task of the decision maker is the task of evaluating the relative importance of the various positive and negative impacts. This includes not only the impacts that are internal to the organization but also the externalities. Evaluating the impacts also means dealing with their risks and uncertainties. The decision maker's values also include his or her attitudes toward risk. It is in this effort that the decision maker's role is uniquely different from that of the analyst.



Cost-Effectiveness Analysis, Fig. 1 The elements of cost-effectiveness analysis

Once the impacts have been evaluated, then choice can be merely a matter of adding them up and comparing the weighted impacts of each of the alternatives.

Criteria

Cost-benefit ratios — Cost effectiveness analysis often is implemented by classifying each impact of a system as either a cost or a benefit. Common units are then found for costs and for benefits and the discounted present value of each is calculated. Alternatives are compared by the ratio of these two measures.

Using a cost-benefit ratio to choose among alternatives presents several problems. Often, this approach leaves out relevant measures (i.e., treats them as the externalities depicted in Fig. 1) because those impacts cannot be evaluated in units that are comparable to the main impacts. Choosing units for the main impacts involves subjective decisions that trade-off relative measures of merit. For example, lives lost must be compared to visual pollution or environmental impact must be valued relative to economic loss. The person who determines common units for such diverse measures of merit is no longer playing the role of an analyst. That person is acting as the decision maker.

The alternative is to leave the various measures of merit uncombined. But a major problem occurs with ratio analysis when the analysis must consider multiple inputs and multiple outputs. Several ratios may be constructed but then it is not clear how these multiple ratios should be combined to determine the overall value of an alternative. Cost-benefit ratios provide the decision maker with little guidance on how to proceed in this case.

Another problem with ratio analysis is the constant returns to scale assumption that is implicit in calculating a cost-benefit ratio. By displaying the results in ratio form, the analyst implies that if the system is expanded or contracted the costs and the benefits both change proportionately. Unfortunately, the world is replete with examples of alternatives that violate such proportionality rules. The use of the incremental cost-effectiveness ratio recognizes this problem. Finally, ratio analysis does not lend itself to explicit treatments of risk and uncertainty, see Conigliani and Tancredi (2009).

Production functions — The production function approach to CEA can deal with variable returns to scale and with other nonlinearities in technology. Numerous estimates of costs and benefits for various alternatives at different scale levels are used to fit a nonlinear production function by regression. This technique can deal with several measures of input and can therefore overcome some of the difficulties of the cost-benefit ratio. Production functions can also incorporate risk described with random variables. But, the multiple regression production function also has some drawbacks. First, the use of regression tends to measure efficiency relative to average performance instead of best performance. That is, all the observations are pooled to fit the production function, a measure of average efficiency, then each alternative is compared to that average measure. Also, multiple regression requires that a single indicator for output

be used. Thus, multiple outputs must be combined into a single effectiveness indicator, similar to ratio analysis. This type of problem is especially severe in non-profit and governmental organizations where prices for outputs are unavailable or incomplete. Charnes and Cooper (1985) also criticized regression's lack of ability, "in identifying the underlying sources and amounts of inefficiencies."

Data envelopment analysis — Data envelopment analysis (DEA) provides an efficiency measure that offers some aid for the criterion problem. This linear-programming based measure has its origin in linear production theory. Golany (1988) pointed out that "DEA is quickly emerging as the leading method for efficient evaluation, in terms of both the number of research papers published and the number of applications to real world problems."

DEA is a procedure that has been designed specifically to measure relative efficiency in situations in which there are multiple measures of merit and there is no obvious objective way of aggregating measures of merit into a meaningful index of productive efficiency. Compared to regression, which averages the aggregate impact of a system, DEA is an extremal method. DEA calculates the efficiency of each alternative by comparing (via mathematical programming models) an alternative's measures of merit with the measures of merit of the other alternatives. Each alternative's measures of merit are weighed as favorably as possible. If the alternative is inefficient, DEA indicates which of its measures of merit imply its inefficiency. Also, DEA does not require the parametric specification of a production function; it derives an estimate of the production function directly from the observed data on elements of cost and effectiveness that are model outputs. DEA has been used to measure the productivity and efficiency of many organizations. It has been particularly useful for public sector organizations where market prices of outputs are not available. DEA has the potential to be extremely helpful in developing criteria in cost effectiveness analyses.

Advances — Contributions to the literature on CEA make explicit use of methods for analyzing risk and uncertainty, Conigliani and Tancredi, (2009); of dynamic models some using Markov models, Soares and Castro, (2010); and others using computable general equilibrium models, Löschel and Otto (2009).

Examples. Cost effectiveness analyses have been conducted in support of (and in opposition to) numerous significant national decisions. For example, the study of alternative delivery systems that resulted in the choice of the space shuttle, the series of studies on the Anti-Ballistic Missile, and the studies for and against the breakup of AT&T are classic studies that illustrate both the power and the fragility of this important concept.

See

- ▶ Cost Analysis
- ▶ Data Envelopment Analysis
- ▶ Measure of Effectiveness (MOE)
- ▶ Multi-Criteria Decision Making (MCDM)
- ▶ Opportunity Cost

References

Charnes, A., & Cooper, W. W. (1985). Preface to topics in data envelopment analysis. *Annals Operations Research, 2*, 59–94.

Charnes, A., Cooper, W. W., & Sueyoshi, T. (1988). A goal programming/constrained regression review of the bell system breakup. *Management Science, 34*, 1–26.

Conigliani, C., & Tancredi, A. (2009). A Bayesian model averaging approach for cost-effectiveness analyses. *Health Economics, 18*, 807–821.

Dorfman, R. (1996). Why benefit-cost analysis is widely disregarded and what to do about it. *Interfaces, 26*(5), 1–6.

Dupuit, J. (1844). *De la Mesure de l'utilité des travaux publics*. Reprinted in Jules Dupuit, De l'utilitéet de sa mesure, Torino, la Roforma sociale, 1933.

Evans, D. S., & Heckman, J. J. (1983). Natural monopoly. In D. S. Evans (Ed.), *Breaking up bell* (pp. 127–156). New York: North Holland.

Evans, D. S., & Heckman, J. J. (1988). Natural monopoly and the bell system: Response to Charnes, Cooper and Sueyoshi. *Management Science, 34*, 27–38.

Golany, B. (1988). An interactive MOLP procedure for the extension of DEA to effectiveness analysis. *Journal of the Operational Research Society, 39*, 725–734.

Gregory, W. H. (1973). NASA analyzes shuttle economics. *Aviation week and space technology*, Sept 24, 1973.

Heiss, K. P., & Morgenstern, O. (1971). *Factors for a decision on a new reusable space transportation system*. Memorandum for Dr. James C. Fletcher, Administrator NASA, Mathematica, Princeton, NJ.

Löschel, A., & Otto, V. M. (2009). Technological uncertainty and cost effectiveness of CO_2 emission reduction. *Energy Economics, 31*, S4–S17.

Operations Research Society of America. (1971). Guidelines for the practice of operations research. *Operations Research, 19*, 1123–1258.

Quade, E. S. (1964). *Analysis of military decisions*. Santa Monica, CA: United States Air Force Project Rand, R–387-PR.
Quade, E. S. (1971). *A history of cost-effectiveness*. Santa Monica, CA: United States Air Force Project Rand, P-4557.
Soares, M. O., & Castro, L. C. (2010). Simulation or cohort models? Continuous time simulation and discretized Markov models to estimate cost-effectiveness. *CHE Research Paper Centre for Health Economics*, Alcuin College, University of York, York, UK.
Sueyoshi, T. (1991). Estimation of stochastic frontier cost function using data envelopment analysis: An application to the AT&T divestiture. *Journal of the Operational Research Society, 42*, 463–477.

COV

▶ Coefficient of Variation

Covering Problem

▶ Set-Covering Problem

Coxian Distribution

A probability distribution whose Laplace-Stieltjes transform may be written as the quotient of two polynomials (i.e., a rational function). All Coxian distributions have a phase-type formulation which may include fictitious stages.

See

▶ Queueing Theory

CPM

Critical path method.

See

▶ Critical Path Method (CPM)
▶ Network Planning
▶ PERT
▶ Research and Development

CPP

▶ Chinese Postman Problem

Cramer's Rule

A formula for calculating the solution of a nonsingular system of linear equations. Cramer's rule states that the solution of the $(n \times n)$ nonsingular linear system $\boldsymbol{Ax} = \boldsymbol{b}$ is $x_i = det\ \boldsymbol{A_i(b)}/det\ \boldsymbol{A}$, $i = 1, \ldots, n$, where $det\ \boldsymbol{A}$ is the determinant of A, and $det\ \boldsymbol{A_i(b)}$ is the determinant of the matrix obtained by replacing the ith column of $\boldsymbol{A}$ by the right-hand side vector $\boldsymbol{b}$. This rule is inefficient for numerical computation and its main use is in theoretical analysis.

See

▶ Matrices and Matrix Algebra

Crash Cost

The estimated cost for a job (project) based on its crash time.

See

▶ Network Planning

Crash Time

The minimal time in which a job may be completed by expediting the work.

See

▶ Network Planning

Crew Scheduling

The determination of the temporal and special succession of the activities of staff personnel, as, for example, in an airlines, train, factory, etc. Such problems are often modeled as mathematical programs.

See

► Airline Industry Operations Research

Crime and Justice

Arnold Barnett[1], Jonathan P. Caulkins[2] and Michael D. Maltz[3]
[1]Massachusetts Institute of Technology, Cambridge, MA, USA
[2]Carnegie Mellon University, Pittsburgh, PA, USA
[3]University of Illinois at Chicago, Chicago, IL, USA

Introduction

Ever since the publications of President Johnson's Commission on Law Enforcement and Administration of Justice (Government Printing Office 1967a, b), OR/MS professionals have investigated just about all facets of the U.S. national, state, and local aspects of crime and justice. There results have had a major influence all out of proportion to their numbers; OR/MS scholars have transformed the way many decision makers think about problems of crime and punishment. Of particular importance is the research of Blumstein and Larson (1969) on the total criminal justice system.

The OR/MS contribution pervades quantitative discussions about crime and justice systems. It has generated a more precise and transparent description of the crime problem than had hitherto been available. It has achieved uneven but sometimes magnificent successes in both identifying and implementing crime-reduction strategies. And it has enhanced the scientific rigor with which criminal justice policy experiments are analyzed and interpreted.

It is not commonly known that some of the most frequently used tools of OR/MS were developed because of crime and justice problems. In the early 19th century, France began to amass statistics on the operation of the criminal justice system (Daston 1988), and the richness of these data led statisticians to devise new techniques to analyze them. Stigler (1986) describes how Simeon Denis Poisson developed the statistical distribution that bears his name – arguably the union label of the OR/MS profession – while modeling conviction rates in French courtrooms. Similarly, Hacking (1990) shows how Poisson developed the law of large numbers by modeling the reliability of jurors in criminal trials.

As noted, the application of OR to crime and justice began in the mid-1960s, when operations researchers and systems analysts on the President's Crime Commission directed their talents to the science and technology aspects of the criminal justice system (Government Printing Office 1967b). Since then, the application of OR/MS ideas in this area has burgeoned (Maltz 1994). Some of the more salient roles played by OR/MS in this field are discussed in this article, especially how OR/MS has been used in analyzing crime statistics, offender behavior, and criminal justice system dynamics. Also described here are how queueing models and optimization techniques have been applied in criminal justice contexts, how OR/MS has caused (some) criminologists to rethink some of their conclusions, the growing role of Geographic Information Systems (GIS) in criminal justice, and how OR/MS is pioneering the extension of quantitative analysis to model offenders who do not fit the traditional street offender mold.

Homicide

In discussing crime, it is natural to start with the most serious offense – murder. Led by the FBI, those assessing homicide patterns had thought it sufficient to consider annual murder rates, expressed in killings per 100,000 citizens per year. The calculated rates had a reassuring quality about them: if 50 per 100,000 citizens were murdered last year, then the other 99,950 were not murdered. Thus, after Detroit had precisely that murder rate in 1973, *The New York Times* reported that "If you live in Detroit, the odds are 2000–1 (i.e., 99,950–50) that you will not be killed by

one of your fellow citizens. Optimists searching for perspective in the city's murder statistics insist that these odds are pretty good."

But some OR/MS scholars raised a question: why measure homicide risk per year as opposed to (say) per day, per month, or per decade? Given that an urban resident has a lifetime danger of being murdered, that would seem the natural time frame over which to measure the risk. And at an annual risk of 1 in 2000, a person with a natural lifespan of 70 years would face a lifetime murder risk of 1 in 28 (!!). Refinements of this raw calculation leave its result virtually unchanged (Barnett et al. 1975; 1980; Barnett and Schwartz 1989).

The idea of estimating lifetime risk of murder has come into general use: detailed projections appeared in the 1981 FBI *Uniform Crime Reports*, and such forecasts have since been incorporated into the actuarial projections of the (U.S.) Centers for Disease Control. There is now widespread awareness that homicide is not a tragic, rare phenomenon, but instead a critical public health problem.

Parallel reasoning is likewise being applied to the lifetime risk of imprisonment. For example, even though the probability of being sent to prison on any given day is small, Bonczar and Beck (1997) estimate that 5.1% of all persons in the U.S. and 28.5% of black males will serve time in prison at some point in their lives.

Offender Behavior

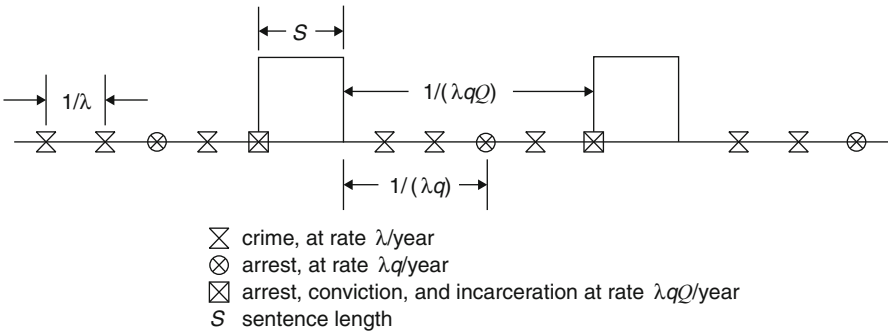
From the standpoint of public policy, it makes a great deal of difference whether existing crimes are committed by relatively few individuals who all offend frequently or by a large number who all offend rarely. OR/MS researchers have taken part in efforts to estimate the total number of offenders, some of whom may never be apprehended for their crimes (Greene and Stollmack 1981; Greene 1984). Of course, the offender population is highly diverse in terms of both frequency of criminal activity and types of crime committed (Chaiken and Chaiken 1982). A major OR/MS contribution to criminal justice has been in creating succinct models that can characterize both individual criminal behavior and the variation of that behavior across offenders.

Most offenders do not commit crimes according to some deterministic schedule. The exact nature of their crime-generation process is by and large unknown, but it is generally safe to say that the aggregate crime commission by a group of offenders can be modeled by the Poisson distribution. This distribution plays the same role in aggregating point processes that the normal distribution plays in aggregating continuous processes.

In highly influential work, Shinnar and Shinnar (1975) proposed a simple but insightful model of the crime and punishment process. The authors assumed that an active offender commits crimes at a Poisson rate λ per year over a career of length T years. If not arrested, the offender would commit on average λT career crimes. But things change if, like Shinnar and Shinnar, it is assumed that the offender's probability of arrest for each crime is q, that the probability of imprisonment given arrest is Q, and that the average sentence length per prison term is S. If career length T is long relative to sentence length S, then steady-state arguments imply that, under the revised scenario, the offender is free on average for only $1/\lambda qQ$ years between successive imprisonments (Fig. 1). Thus, because of detention, the offender is free and active only for fraction $(1/\lambda qQ)/(1/\lambda qQ + S)$ of the offender's career rather than for all of it. It follows that incapacitation has reduced the offender's total number of offenses by the proportion $S/(S + 1/\lambda qQ)$ compared to the number in a world free of punishment.

There are some gross simplifications in this model (for example, the career length T is assumed independent of the punishment policy in place). But it encapsulates in one equation the effects of all primary elements of the criminal justice system: the offender (via crime commission rate λ), the police (arrest probability q), the courts (chance of imprisonment given arrest, Q), and the correctional system (sentence length S). The model also provides guidance to those exploring empirical data (e.g., offenders' arrest, sentencing, and conviction records) about which quantities were especially worth trying to estimate.

OR/MS professionals like Blumstein and his colleagues worked to flesh out the description of the individual criminal career (Blumstein et al. 1986). They estimated key parameters like the proportion of citizens who participated at some time in criminal behavior, the frequency of crime-commission during the career, the degree to which offenders specialize by crime-type, and the duration of the criminal career.



Crime and Justice, Fig. 1 A (deterministic) criminal career

A simple model of the career might summarize it with four parameters: P, the fraction of individuals in a birth cohort who initiate criminal careers; A, the age of onset for the career; λ, the average annual crime commission rate while free and active, and ρ, the annual probability that the career ends. A macroscopic model could reflect diversity among offenders by assigning a population-wide distribution to each of these parameters, as does the model in Blumstein et al. (1993). (Other distributions about offense type would fill out the description; e.g., Chaiken and Chaiken 1982). Interestingly, offenders who differ greatly on some parameters may be quite similar on others. In a cohort of London multiple-offenders, for example, individuals appear to differ far more in their λ-values than their ρ -values (Barnett et al. 1987). Thus, their career lengths may diverge far less than do intensities of activity during their careers.

A common problem in criminal justice policy analysis has been estimating the effect of enhanced sentences on both crime rates and criminal justice spending. For example, Greenwood et al. (1994) used these Poisson models of criminal offending to predict the consequences of full-scale implementation of California's "Three-Strikes and You're Out" law, and Greenwood et al. (1999) use the model to help identify ways in which actual implementation deviated from that bench-mark.

As is shown in Fig. 1, many offenders continue to commit offenses despite their having been in correctional institutions. But not all do, and the extent of recidivism (commission of additional offenses) is an important concern in criminal justice research. OR/MS researchers have devised, calibrated, and tested probabilistic models that assess the likelihood that given offenders with given past records will again commit crimes within particular future time periods (Stollmack and Harris 1974; Harris et al. 1981; Maltz 1984; Ellerman et al. 1992). These flexible and mathematically-rich techniques allow frequent updating of the prognoses for particular individuals.

The Criminal Justice System (CJS)

With mathematical models, OR/MS professionals described the idea that the CJS – composed of police, courts, and corrections — is, in fact, a system (Government Printing Office 1967b), within which policy shifts in one component generally have consequences for the others. An increase in arrests aimed at reducing crime, for example, can first clog the courts and then overcrowd jails and prisons which, in turn, may be required to reduce surging inmate populations by instituting early release programs for those incarcerated. One of the earliest models to incorporate such feedback effects was the Justice System Interactive Model (JUSSIM), (Belkin et al. 1972); subsequent efforts include Cassidy (1985) and Morgan (1985). JUSSIM has since been updated and software written for personal computers by the U.S. Department of Justice to permit its widespread use (Institute for Law and Justice 1991).

While it is hoped that the CJS provides a fair and cost-effective way to reduce crime, there is a continuing national debate about whether this goal can be achieved. The aims of the CJS are to deter potential offenders from committing crime, to incapacitate those who have been convicted by imprisoning them, and to rehabilitate past offenders so that they are harmless in the future. Of course, the system might induce undesirable changes in criminal behavior, such as brutalization under

which an offender released from prison is more violent than ever before. Statistical investigations by OR/MS researchers have tried to estimate various net effects of the CJS on crime levels (Blumstein et al. 1978, 1986), as well as to assess the realism of specific attempts to estimate such effects from aggregate data (Barnett 1981).

Over the past many years, the number of people under criminal justice supervision in the U.S. has grown dramatically. Although crime rates have fallen, they were rising during much of the build-up in incarceration and, even if increased incarceration were contributing substantially to declining crime rates, it is clearly an expensive way to suppress crime, in both budgetary and human terms. Hence, there has been considerable interest in creating systems models that embrace not only the CJS, but also broader sets of interventions in a manner that allows different classes of interventions to be compared. The goal is to determine whether spending more money on violence prevention or drug treatment programs, for example, might be a more cost-effective way to reduce crime than would be spending more money on prisons and jails.

These analyses show that violence prevention programs are a promising alternative (Greenwood et al. 1996) and that prevention interventions offer a broad array of benefits. They also, however, find that there is a great deal of uncertainty associated with estimates of prevention's cost-effectiveness, even though many, many individual prevention programs have been evaluated (Caulkins et al. 1999). Importantly, the systems framework identifies the sources of the uncertainty and highlights why past evaluations have not been more informative. Some of the reasons are pedestrian, such as never reporting program costs or a focus on showing statistical significance of effects rather than estimating their magnitude. Others are more insightful, such as the fact that traditional evaluations often only consider effects on program participants, even though indirect effects on those not actually in the program are, in some cases, larger in aggregate.

Queueing Models

While everyone recognizes that crime rates vary from neighborhood to neighborhood and by time of day, OR/MS analysts have built probabilistic models that allow exploration of the consequences of such heterogeneity, especially with respect to practical issues of police deployment and staffing of 911 emergency centers (Larson 1972; Kolesar et al. 1975; Chelst 1978). From such models and OR/MS insights into queueing theory have come an unpleasant realization: randomness in the arrival times of calls for service can cause surprisingly large delays in responding to them. Getting six calls randomly distributed over a one-hour period, for example, can yield much slower responses than getting six calls spaced exactly ten minutes apart.

Queueing theory has been applied and extended in developing improved allocation methods for police patrol resources. Such formulations as the hypercube queueing model (Larson 1974; Larson and Odoni 1981; Larson and Rich 1987) and RAND's Patrol Car Allocation Model (Chaiken and Dormont 1978) have depicted with great accuracy how particular police response strategies affect mean response times, workload imbalance across officers, and a host of other performance measures. The models, which are used by many U.S. cities to set police dispatching strategies, allow the user to vary the number of patrol cars and the deployment rules, and then to observe on a computer screen the performance statistics under each scenario. Other OR/MS developments allow the user to set priorities in responding to calls for service and to analyze sending multiple vehicles to incidents (Green and Kolesar 1984). A review of this work is provided in Swersey (1994).

Optimization

Optimization, one of the strongest OR/MS specialties, has played a relatively small role in the profession's contribution to criminal justice. For example, limited success has attended OR/MS efforts to suggest optimal punishment policies. Under particular assumptions about crime-commission processes and their sensitivity to the sentencing strategy in place, Blumstein and Nagin (1978) and Barnett and Lofaso (1986) have worked out optimal allocations of prison space. But the verification of such assumptions — let alone the estimation of key model parameters — has not gone far enough that such models are taken very seriously. Associated attempts to estimate how prison populations vary with changes

in demography and sentencing policy have yielded prison population forecasts that do not immediately demonstrate the practicality of the models (Blumstein et al. 1980; Barnett 1987).

Perhaps the most famous OR/MS proposal for optimal prison sentencing was Greenwood's selective incapacitation scheme in which heavy sentences would be imposed on offenders with at least four of seven high-risk characteristics (Chaiken and Rolph 1980). But data analyses revealed difficulties with implementing such policies (Chaiken and Chaiken 1982; Greenwood and Turner 1987), including a high rate of false positives (people incarcerated to prevent projected future crimes that would never have occurred were they free). These false positives raise controversies about sentencing by conjecture and yield smaller crime-reduction benefits in practice than the strategy can achieve in theory.

A Sense of Ambiguity

Sometimes OR/MS people have contributed to criminal justice research less by what they said than by what they didn't say. OR/MS scholars approach data with a sense of ambiguity: an awareness that a particular empirical pattern is often consistent with a broad range of possibilities. Thus, they have usefully called out "not so fast!" when the most obvious interpretation of certain data was being treated as the only viable one. Four examples of such rescue activities are described below.

One case concerns the Kansas City Preventive Patrol Experiment conducted in the early 1970s. When not responding to calls to service, patrol cars drive randomly through their districts; in theory, such preventive patrol reduce crimes because would-be offenders realize that, even if their victims cannot contact the police, a patrol car might reach the crime scene purely by chance. That theory was called into doubt after Kansas City, in a prearranged experiment, acted to increase preventive patrol sharply in some beats and virtually eliminate it in others. When neither beat-by-beat crime rates nor citizen perceptions about police presence changed visibly during the (unannounced) experiment, some people saw preventive patrol as having lost any rationale.

arson (1975), however, demonstrated with detailed calculations that actual conditions during the Kansas City experiment were quite different from the anticipated ones. Patrol cars from high-activity beats were spending much of their time responding to calls for service from low-activity ones, which had been deprived of all local police vehicles. The upshot was that there was a great deal of police-car movement — often with sirens screaming — in the districts supposedly without preventive patrol, and surprisingly little increase in patrol in the districts supposedly saturated with it. Perhaps, Larson argued, the reason crime rates and citizen perceptions did not change was that police activity itself had not meaningfully changed.

A second example concerned the relationship, well-known to criminologists, between arrests and age. The graph of arrests vs. age is unimodal, reaching a peak in the late teens and then dropping off steadily and sharply. Given this curve, some people argued that it was not cost-effective to give long sentences to offenders convicted at age 30; such offenders, it was contended, were already far less active than at their primes and were unlikely to do much harm even if left on the streets.

But Blumstein et al. (1982) pointed out that such an analysis was vulnerable to a variant of the well-known ecological fallacy: Even if arrests in the aggregate were dropping rapidly with age, it did not follow that individual offenders exhibited this pattern. Having studied longitudinal data about individual offenders, they found that the drop in arrests with age reflected not less activity per year among active offenders, but rather a growing fraction of offenders who had retired from criminal activity. Statistically, an individual convicted at age 30, presumably still active at that age, would be expected, if allowed to go free, to commit as many crimes over the next several years as someone several years younger.

While citizens in the U.S. were debating in the mid-1970s whether to restore the death penalty, several economists came forth with historically-based analyses that purported to weed out extraneous factors and estimate how each execution affects the overall homicide level. The model, whose findings were cited by the U.S. Supreme Court, purported to show that each execution deterred eight homicides. But Barnett (1981) wondered whether the econometric models being used had sufficient explanatory power to fulfill their ambitious goals. Arguing that homicide levels were subject to roughly Poisson-level statistical

noise, he proposed a test of how well the econometric models could forecast state-by-state homicide levels in the very data sets used to calibrate them. The test results indicated that the predictions from all the models suffered large systematic errors of unknown cause and that, indeed, the errors were far larger than any reasonable estimate of the size of the effect the models sought to measure. Thus, Barnett concluded, the analyses were not sensitive enough to answer the question that motivated them.

A final example of ambiguity arises from a study that concluded that juvenile detention acts to reduce delinquency. The study found that an average Illinois male youth sent to a reformatory, though not cured of criminal activity by his stay there, was arrested far fewer times per month after his release than just prior to his detention. The decline was interpreted as the post-release suppression effect on incarceration: getting tough works.

Maltz and Pollock (1980), however, saw another possibility, tied to the phenomenon called regression-to-the-mean. Even if a youth commits crime at a steady frequency and has an unchanging probability of arrest per offense, varying luck in the arrest lottery will cause his observed arrest rate to fluctuate from month to month. But the authorities are especially likely to send him to a reformatory after an upsurge of arrests — that is, at a peak of the fluctuating pattern. Thus, even if the reformatory has no effect on his underlying pattern of criminal behavior, his post-detention arrests would likely fall in frequency compared to his unluckily high predetention levels. Tierney (1983) proposed a revision in their analysis that modified its result, but the work still showed that the suppression effect was quite possibly just an illusion.

These four examples show one of the primary assets of OR/MS thinking as applied to a field so data rich as the CJS. Although data may exhibit certain aggregate patterns, these patterns need not illuminate what is happening at the more detailed level that is, quite often, the appropriate focus of policy analysis. OR/MS analysts should never forget the importance of studying a problem's molecular structure.

Geographical Analyses

Location is a key attribute of crime. Crime is the result of a convergence in time and space of criminals and victims, in the absence of guardians. Land use types, physical geography, the built environment, population characteristics, and police resource allocations and tactics are among the location-based factors influencing crime. Until recently, there was limited capacity for analysts to consider geographical factors explicitly. (Even a panel data set with city-level information is aggregated both in the sense that all events within a city are pooled and in the sense that information about the relative proximity of different cities is usually ignored). The advent and development of geographic information systems (GIS) has changed this situation. GIS is an information technology that geocodes information and processes it spatially in order to facilitate analysis.

Geocoding takes three forms. GIS uses 1) rule bases and scoring to match text of crime incident street addresses with street addresses that already have locations (latitude/longitude coordinates or their flat projections) to place points on maps, 2) global positioning system instruments to read world coordinates of a point, and 3) spatial overlay of map boundary layers (e.g., police precincts, patrol beats, or uniform grid cells) on crime incident points to classify the points' area membership.

Spatial processing includes 1) integrating crime incidents and other factors affecting crime through location, 2) spatial overlay of points and reapportionment of area-based statistics to a consistent crime space/time series (e.g., counts of persons aged 14 to 25 and monthly robberies of persons by spatial grid cell), 3) proximity determination using spatial queries (e.g., all drug arrests within 1000 feet of schools), and 4) connectivity of streets for routing.

The geocoded and processed information can be used for a wide variety of decision support roles. Crime maps for analysis or communication (Maltz et al. 1991; Brantingham and Brantingham 1998) are among the most familiar, but crime forecasting (Foster and Gorr 1986; Gorr and Olligschlaeger 1994; Olligschlaeger 1998) is growing in importance, particularly in conjunction with computer statistics (COMSTAT) police management systems that need counter-factuals against which actual performance can be compared. Likewise, GIS can provide the real time data and detailed information about street networks that is needed to implement effectively some of the OR models described above. Examples

of these applications include patrol resource allocation and the design of administrative boundaries to balance workloads (Koper 1995), as well as queueing applications for routing emergency response (Green 1984; Larson and Rich 1987; Larson and McKnew 1982).

Consensual Crime

Originally the modern application of OR/MS efforts in crime and justice focused on the actions and consequences of the typical street offender who robbed, burglarized, assaulted, or killed strangers. OR/MS methods such as process control charts (Anderson and Diaz 1996), data envelopment analysis (Thanassoulis 1995), and simulation (Larson et al. 1993) continue to be applied in innovative ways to address these problems. But other OR/MS methods have also been finding their way into analyses of consensual crimes including corruption and drug trafficking.

Corruption is widespread among societies and institutions and has stimulated a small but vibrant literature at the intersection of management science and economics. An early standard reference on corruption is Rose-Ackerman (1978) who studied the economics of the supply and demand of bribes. These ideas have been applied to analyses of tax evasion (Chander and Wilde 1992), the distribution of bribes within a hierarchical bureaucracy (Hillman and Katz 1987), and a range of other situations reviewed by Andvig (1991) and Shleifer and Vishny (1993).

Understandably, researchers often approach corruption in game-theoretic terms. Static analyses are common (e.g., Basu et al. 1992; Mookherjee and Png 1995; Marjit and Shi 1998), in part because of their relative tractability, but some of the most exciting developments have involved dynamic optimization. For example, Dawid and Feichtinger (1996a) dynamically extend Akerlof's (1980) model. They find that, unless corruption is the globally dominant strategy, the solution is like that described by a so-called Schelling diagram (Schelling 1973; cf. also Andvig 1991). There are two locally stable equilibria, one where everyone is corrupt and this corruption is accepted, and another where the whole population is honest and corruption is uniformly condemned. The only intermediate equilibrium is unstable.

Likewise, Lui (1986) considers the impact of exogenous corruption deterrence on the (stationary) level of corruption and views anti-corruption campaigns as efforts to shift from an unfavorable to a favorable equilibrium. Feichtinger and Wirl (1994) endogenize these episodes of crusades against corruption. Antoci and Sacco (1995) use replicator dynamics to describe the changing behavior of a population where each individual can decide in each period whether the individual will act honestly or be corrupt. Bicchieri and Rovelli (1995) model the exchange of bribes as a system in which there are two types of players who play a sequence of repeated prisoner's dilemma games with randomly chosen opponents. Wirl et al. (1997) consider interaction between a corrupt politician and an investigative journalist in a differential game and calculate the open-loop Nash equilibrium, which generates interesting insights into the non-cooperative dynamic interaction of crime and enforcement. [Dawid and Feichtinger (1996b) extend the analysis for a similar model to a feedback Nash equilibrium]. Bicchieri and Duffy (1997) demonstrate how corruption can become cyclical under the assumption that politicians, in order to be reelected, have to compensate voters through material incentives.

Whereas corruption has been a problem through the ages, illicit drugs have risen to prominence during the last half of the 20th century. But the OR/MS community and OR/MS tools have already played a prominent role in this new area. Some of the applications have looked specifically at the relationship between drug use and predatory crime (Powers et al. 1991), but many focus explicitly on the production, distribution, sale, and consumption of the drugs themselves. Interdiction activities have received particular attention (Caulkins et al. 1993; Washburn and Wood 1995), perhaps because of the prominent role of the military in that sphere. But production in source countries (Kennedy et al. 1993), domestic distribution networks (Caulkins 1997), managing local enforcement operations (Caulkins 1993a; Naik et al. 1996; Baveja et al. 1997), and drug testing (Lattimore et al. 1996; Meyer and Savory 1997; Kaushal et al. 1998) have also been active research areas.

Just as the Poisson model of offending has been a workhorse in the analysis of predatory offenders, Markov models of drug demand (Everingham et al. 1995) and models of drug markets provide the framework for systems analyses that compare the

effectiveness of different drug control interventions. Typical findings include, for example, that treating heavy users is cost-effective (Rydell et al. 1996), and that mandatory minimum sentences are not (Caulkins 1993b; Caulkins et al. 1997).

The finding that treatment is cost-effective illustrates the importance of choosing the right objective function. Treating heavy users performs miserably if the performance measure is proportion of people treated who are abstinent two years later, but it dominates other available interventions when the performance measure is kilograms of cocaine consumption averted per million dollars spent. True, most people quickly drop out of treatment. They make relapse rates look awful and they contribute nothing to the numerator of a measure such as consumption averted per million dollars. They also, however, do not contribute much to the denominator. The system simply cannot waste that much money on someone who only stays in the program for a few days. Furthermore, relapse measures completely ignore the benefits of reductions in use while someone is in treatment. It turns out that even if 100% of heavy users relapsed, treatment would still be a cost-effective way to reduce drug use just on the basis of the in-treatment effect.

Dynamic models that examine how policies should vary over the course of a drug epidemic are an area of particular interest. Systems dynamics models (Homer 1993) take a descriptive approach to this issue, but an optimal control framework can also yield interesting insights. For example, Tragler et al. (2000) show that detecting the onset of a drug epidemic quickly is valuable because total costs are much lower if control begins early. They also show that sharp price declines, such as those observed in the 1980s for cocaine in the U.S., do not necessarily imply a policy failure; indeed it can be optimal to have such declines. Likewise Behrens et al. (1999) show that it is rarely optimal to advocate greater spending on demand-side interventions generally. Both prevention and treatment can play an important role in drug control, but probably not at the same time. Their comparative advantages come at different stages of an epidemic.

Concluding Remarks

It is not easy to quantify the overall OR/MS contribution to public safety and crime control. OR/MS research has brought about a deeper understanding of the crime problem and how it affects and is affected by the criminal justice system. Crime, however, has such deep psychological, cultural, economic, and social roots that there are limits to what mathematical models can be expected to accomplish on their own. But, likewise, there are limits to what less quantitative perspectives can accomplish on their own. Crime and justice are truly multi-disciplinary problems that are best addressed by multi-disciplinary collaborations, with OR/MS being an integral part of that collaboration.

See

- ▶ Emergency Services
- ▶ Hypercube Queueing Model
- ▶ Program Evaluation
- ▶ Public Policy Analysis
- ▶ Queueing Theory

References

Akerlof, G. A. (1980). A theory of social custom, of which unemployment may be one consequence. *Quarterly Journal of Economics, 94*, 749–775.

Anderson, E. A., & Diaz, J. (1996). Using process control chart techniques to analyze crime rates in Houston, Texas. *Journal of the Operational Research Society, 47*(7), 871–882.

Andvig, J. C. (1991). The economics of corruption: A survey. *Studi Economici, 43*(1), 57–94.

Antoci, A., & Sacco, P. L. (1995). A public contracting evolutionary game with corruption. *Journal of Economics, 61*(2), 89–122.

Barnett, A. (1981). The deterrent effect of capital punishment: A test of some recent studies. *Operations Research, 29*, 346–370.

Barnett, A. (1987). Prison populations: A projection model. *Operations Research, 35*, 18–34.

Barnett, A., Blumstein, A., & Farrington, D. P. (1987). Probabilistic models of youthful criminal careers. *Criminology, 30*, 83–108.

Barnett, A., Essenfeld, E., & Kleitman, D. J. (1980). Urban homicide: Some recent developments. *Journal of Criminal Justice, 8*, 379–385.

Barnett, A., Kleitman, D. J., & Larson, R. C. (1975). On urban homicide: A statistical analysis. *Journal of Criminal Justice, 3*, 85–110.

Barnett, A., & Lofaso, A. J. (1986). On the optimal allocation of prison space. In A. J. Swersey & E. Ignall (Eds.), *Delivery of urban services* (TIMS series in the management sciences, Vol. 22, pp. 249–268). Amsterdam: Elsevier-North Holland.

Barnett, A., & Schwartz, E. (1989). Urban homicide: Still the same. *Journal of Quantitative Criminology, 5*, 83–100.

Basu, K., Bhattacharya, S., & Mishra, A. (1992). Notes on bribery and the control of corruption. *Journal of Public Economics, 48*, 349–359.

Baveja, A., Caulkins, J. P., Liu, W., Batta, R., & Karwan, M. H. (1997). When haste makes sense: Cracking down on street markets for illicit drugs. *Socio-Economic Planning Sciences, 31*, 293–306.

Behrens, D. A., Caulkins, J. P., Tragler, G., & Feichtinger, G. (1999). Optimal control of drug epidemics: Prevent and treat – But not at the same time. *Management Science, 46*, 333–347.

Belkin, J., Blumstein, A., Glass, W., & Lettre, M. (1972). JUSSIM: An interactive computer program and its uses in criminal justice planning. In G. Cooper (Ed.), *Proceedings of international symposium on criminal justice information and statistics systems* (pp. 467–477). Sacramento, CA: Project SEARCH.

Bicchieri, C., & Duffy, J. (1997). Corruption cycles. *Political Studies, 45*, 477–495.

Bicchieri, C., & Rovelli, C. (1995). Evolution and revolution: The dynamics of corruption. *Rationality and Society, 7*, 201–224.

Blumstein, A. (2002). Crime modeling. *Operations Research, 50*(1), 16–24.

Blumstein, A. (2007). An OR missionary's visits to the criminal justice system. *Operations Research, 55*(1), 14–23.

Blumstein, A., Canela-Cacho, J. A., & Cohen, J. (1993). Filtered sampling from populations with heterogeneous event frequencies. *Management Science, 39*, 886–899.

Blumstein, A., Cohen, J., & Hsieh, P. (1982). *The durations of adult criminal careers. Final report to national institute of justice*. Pittsburgh, PA: Carnegie-Mellon University.

Blumstein, A., Cohen, J., & Miller, H. (1980). Demographically disaggregated projections of prison populations. *Journal of Criminal Justice, 8*, 1–25.

Blumstein, A., Cohen, J., & Nagin, D. (Eds.). (1978). *Deterrence and incapacitation: Estimating the effects of criminal sanctions on crime rates*. Washington, DC: National Academy of Sciences.

Blumstein, A., Cohen, J., Roth, J. A., & Visher, C. (1986). *Criminal careers and "career criminals." vols. I and II*. Washington, DC: National Academy of Sciences.

Blumstein, A., & Larson, R. (1969). Models of a total criminal justice system. *Operations Research, 17*(2), 199–232.

Blumstein, A., & Nagin, D. (1978). On the optimum use of incarceration for crime control. *Operations Research, 26*, 383–405.

Bonczar, T. P., & Beck, A. J. (1997). *Lifetime likelihood of going to state or federal prison*. Washington, DC: National Institute of Justice.

Brantingham, P. L., & Brantingham, P. J. (1998). Mapping crime for analytic purposes: Location quotients, counts, and rates. In D. Weisburd & T. McEwen (Eds.), *Crime mapping, crime prevention, crime prevention studies 8*. New York: Criminal Justice Press.

Cassidy, R. G. (1985). Modelling a criminal justice system. In D. P. Farrington & R. Tarling (Eds.), *Prediction in criminology*. Albany, NY: State University of New York Press.

Caulkins, J. (1993a). Zero-tolerance policies: Do they inhibit or stimulate illicit drug consumption? *Management Science, 39*, 458–476.

Caulkins, J. (1993b). Local drug markets' response to focused police enforcement. *Operations Research, 41*, 843–863.

Caulkins, J. P. (1997). Modeling the domestic distribution network for illicit drugs. *Management Science, 43*, 1364–1371.

Caulkins, J. P., Crawford, G., & Reuter, P. (1993). Simulation of adaptive response: A model of drug interdiction. *Mathematical and Computer Modelling, 17*(2), 37–52.

Caulkins, J. P., Rydell, C. P., Everingham, S. S., Chiesa, J., & Bushway, S. (1999). *An ounce of prevention, a pound of uncertainty: The cost-effectiveness of school-based drug prevention program* (Technical report). Santa Monica, CA: RAND Corporation.

Caulkins, J. P., Rydell, C. P., Schwabe, W. L., and Chiesa, J. (1997). Mandatory minimum drug sentences: Throwing away the key or the taxpayers' money? (Report MR-827-DPRC). Santa Monica, CA: RAND Corporation.

Chaiken, J. M., & Chaiken, M. R. (1982). *Varieties of criminal behavior* (Report R-2814-NIJ). Santa Monica, CA: Rand Corporation.

Chaiken, J. M., & Dormont, P. (1978). A patrol car allocation model: Background, capabilities, and algorithms. *Management Science, 24*, 1280–1300.

Chaiken, J. M., & Rolph, J. (1980). Selective incapacitation strategies based on estimated crime rates. *Operations Research, 28*, 1259–1274.

Chander, P., & Wilde, L. (1992). Corruption in tax administration. *Journal of Public Economics, 49*, 333–349.

Chelst, K. (1978). An algorithm for deploying a crime-directed (tactical) patrol force. *Management Science, 24*, 1314–1327.

Cormican, K. J., Morton, D. P., & Wood, R. K. (1998). Stochastic network interdiction. *Operations Research, 46*, 184–197.

Daston, L. (1988). *Classical probability in the enlightenment*. Princeton, NJ: Princeton University Press.

Dawid, H., & Feichtinger, G. (1996a). On the persistence of corruption. *Journal of Economics, 64*(2), 177–193.

Dawid, H., & Feichtinger, G. (1996b). Optimal allocation of drug control efforts: A differential game analysis. *Journal of Optimization Theory and Applications, 91*, 279–297.

Ellerman, P., Sullo, P., & Tien, J. M. (1992). An alternative approach to modeling recidivism using quantile residual life functions. *Operations Research, 40*, 485–504.

Everingham, S., Rydell, C. P., & Caulkins, J. P. (1995). Cocaine consumption in the US: Estimating past trends and future scenarios. *Socio-Economic Planning Sciences, 29*, 305–314.

Feichtinger, G., & Wirl, F. (1994). On the stability and potential cyclicity of corruption in governments subject to popularity constraints. *Mathematical Social Sciences, 28*, 113–131.

Foster, S. A., & Gorr, W. L. (1986). An adaptive filter for estimating spatially-varying parameters: Application to modeling police hours spent in response to calls for service. *Management Science, 32*, 878–889.

Gorr, W. L., & Olligschlaeger, A. M. (1994). Weighted spatial adaptive filtering: Monte Carlo studies and application to illicit drug market modeling. *Geographical Analysis, 26*(1), 67–87.

Government Printing Office. (1967a). *The challenge of crime in a free society*. Washington, DC: President's Commission on Law Enforcement and Administration of Justice.

Government Printing Office. (1967b). *Task force report: Science and technology*. Washington, DC: President's Commission on Law Enforcement and Administration of Justice.

Green, L. (1984). A multiple dispatch queueing model of police patrol operations. *Management Science, 30*, 653–664.

Green, L., & Kolesar, P. (1984). A comparison of multiple dispatch and M/M/C priority queueing models of police patrol. *Management Science, 30*, 665–670.

Greene, M. A. (1984). Estimating the size of the criminal population using an open population approach. *Proceedings American Statistical Association*, *Survey Methods Research Section*, pp. 8–13.

Greene, M. A., & Stollmack, S. (1981). Estimating the number of criminals. In J. A. Fox (Ed.), *Models in quantitative criminology* (pp. 1–24). New York: Academic.

Greenwood, P. W., & Abrahamse, A. F. (1981). *Selective incapacitation* (Report R-2815-NIJ). Santa Monica, CA: Rand Corporation.

Greenwood, P. W., Everingham, S. S., Chen, E., Abrahamse, A. F., Merritt, N., & Chiesa, J. (1999). *Three strikes revisited*: *An early assessment of implementation and effects* (Technical report). Santa Monica, CA: RAND Corporation.

Greenwood, P. W., Model, K. E., Rydell, C. P., & Chiesa, J. (1996). *Diverting children from a life of crime*: *Measuring the costs and benefits* (Report MJ-699-UCB/RC/F). Santa Monica, CA: RAND Corporation.

Greenwood, P. W., Rydell, C. P., Abrahamse, A. F., Caulkins, J. P., Chiesa, J. R., Model, K. E., & Klein, S. P. (1994). *Three strikes and you're out*: *Estimated benefits and costs of California's new mandatory-sentencing law* (Report MR-509-RC). Santa Monica, CA: RAND Corporation.

Greenwood, P. W., & Turner, S. (1987). *Selective incapacitation revisited*: *Why the high-rate offenders are hard to predict* (Report R-3397-NIJ). Santa Monica, CA: Rand Corporation.

Hacking, I. (1990). *The taming of chance*. England: Cambridge University Press.

Harris, C. M., Kaylan, A. R., & Maltz, M. D. (1981). Recent advances in the statistics of recidivism measurement. In J. A. Fox (Ed.), *Models of quantitative criminology* (pp. 61–79). New York: Academic Press.

Hillman, A. L., & Katz, E. (1987). Hierarchical structure and the social costs of bribes and transfers. *Journal of Public Economics, 34*, 129–142.

Homer, J. B. (1993). A system dynamics model of national cocaine prevalence. *System Dynamics Review, 9*(1), 49–78.

Institute for Law and Justice. (1991). *CJSSIM: Criminal justice system simulation model: Software and user manual*. Alexandria, VA: Institute for Law and Justice.

Karoly, L. A., Greenwood, P. W., Everingham, S. S., Houbé, J., Kilburn, M. R., Rydell, C. P., Sanders, M., & Chiesa, J. (1998). *Investing in our children*: *What we know and don't know About the costs and benefits of early childhood*, *interventions* (MR-898-TCWF). Santa Monica, CA: RAND Corporation.

Kaushal, C., Baker, J. R., & Lattimore, P. K. (1998). A decision support system for partial drug testing. *Decision Support Systems, 23*, 241–257.

Kennedy, M., Reuter, P., & Riley, K. J. (1993). A simple economic model of cocaine production. *Mathematical and Computer Modelling, 12*(2), 19–36.

Kolesar, P. J., Rider, K. L., Crabill, T. B., & Walker, W. W. (1975). A queuing-linear programming approach to scheduling police patrol cars. *Operations Research, 23*, 1045–1062.

Koper, C. S. (1995). Just enough police presence: Reducing crime and disorderly behavior by optimizing patrol time in crime hot spots. *Justice Quarterly, 12*, 649–672.

Larson, R. C. (1972). *Urban police patrol analysis*. Cambridge, MA: MIT Press.

Larson, R. C. (1974). A hypercube queueing model for facility location and redistricting in urban emergency services. *Journal of Computers and Operations Research, 1*, 67–95.

Larson, R. C. (1975). What happened to patrol operations in Kansas city?: A review of the Kansas city preventive patrol experiment. *Journal of Criminal Justice, 3*, 267–297.

Larson, R. C., Cahn, M. F., & Shell, M. C. (1993). Improving the New York city arrest-to-arraignment system. *Interfaces, 23*(1), 76–96.

Larson, R. C., & McKnew, M. A. (1982). Police patrol-initiated activities within a systems queueing model. *Management Science, 28*, 759–774.

Larson, R. C., & Odoni, A. R. (1981). The hypercube queueing model. In *Urban operations research* (pp. 292–335). Englewood Cliffs, NJ: Prentice-Hall.

Larson, R. C., & Rich, T. (1987). Travel time analysis of New York city police patrol cars. *Interfaces, 17*(2), 15–20.

Lattimore, P. K., Baker, J. R., & Matheson, L. A. (1996). Monitoring drug use using Bayesian acceptance sampling: The Illinois experiment. *Operations Research, 44*, 274–285.

Lui, F. T. (1986). A dynamical model of corruption deterrence. *Journal of Public Economics, 31*, 215–236.

Maltz, M. D. (1984). *Recidivism*. Orlando, FL: Academic Press.

Maltz, M. D. (1994). Operations research in studying crime and justice: Its history and accomplishments. In S. M. Pollock, A. Barnett, & M. Rothkopf (Eds.), *Operations research and public systems*. Amsterdam: Elsevier.

Maltz, M. D., Gordon, A. C., & Friedman, W. (1991). *Mapping crime in its community setting: Event geography analysis*. New York: Springer.

Maltz, M. D., & Pollock, S. M. (1980). Artificial inflation of a delinquency rate by a selection artifact. *Operations Research, 28*, 547–559.

Marjit, S., & Shi, H. L. (1998). On controlling crime with corrupt officials. *Journal of Economic Behavior and Organization, 34*(1), 163–172.

Meyer, J. L., & Savory, P. A. (1997). Selecting employees for random drug tests at union pacific railroad. *Interfaces, 27*(5), 58–67.

Mookherjee, D., & Png, I. P. L. (1995). Corruptible law enforcers: How should they be compensated? *The Economic Journal, 105*, 145–159.

Morgan, P. M. (1985). *Modelling the criminal justice system*. Home office research and planning unit paper 35. London: Home Office.

Naik, A. V., Baveja, A., Batta, R., & Caulkins, J. P. (1996). Scheduling crackdowns on illicit drug markets. *European Journal of Operational Research, 88*, 231–250.

Olligschlaeger, A. M. (1998). Artificial neural networks and crime mapping. In D. Weisburd & T. McEwen (Eds.), *Crime mapping, crime prevention, crime prevention studies 8*. New York: Criminal Justice Press.

Powers, K., Hanssens, D. M., & Hser, Y. I. (1991). Measuring the long-term effects of public policy: The case of narcotics use and property crime. *Management Science, 37*, 627–644.
Rose-Ackerman, S. (1978). *Corruption: A study in political economy*. New York: Academic Press.
Rydell, C. P., Caulkins, J. P., & Everingham, S. (1996). Enforcement or treatment: Modeling the relative efficacy of alternatives for controlling cocaine. *Operations Research, 44*, 687–695.
Schelling, T. C. (1973). Hockey helmets, concealed weapons, and daylight saving: A study of binary choices with externalities. *Journal of Conflict Resolution, 17*, 381–428.
Shinnar, R., & Shinnar, S. (1975). The effects of the criminal justice system on the control of crime: A quantitative approach. *Law and Society Review, 9*, 581–611.
Shleifer, A., & Vishny, R. W. (1993). Corruption. *Quarterly Journal of Economics, 108*, 599–617.
STIF: Science and Technology Task Force. (1967). *Task force report: Science and technology. President's commission on law enforcement and the administration of justice*. Washington, DC: US Government Printing Office.
Stigler, S. M. (1986). *The history of statistics: The measurement of uncertainty before 1900*. Cambridge, MA: The Belknap Press of Harvard University Press.
Stollmack, S., & Harris, C. (1974). Failure-rate analysis applied to recidivism data. *Operations Research, 22*, 1192–1205.
Swersey, A. J. (1994). The deployment of police, fire, and emergency medical units. In S. M. Pollock, A. Barnett, & M. Rothkopf (Eds.), *Operations research and public systems*. Amsterdam: Elsevier.
Thanassoulis, E. (1995). Assessing police forces in England and Wales using data envelopment analysis. *European Journal of Operational Research, 87*, 641–658.
Tierney, L. (1983). A selection artifact in delinquency data revisited. *Operations Research, 31*, 852–865.
Tragler, G., Caulkins, J. P., & Feichtinger, G. (2000). Optimal dynamic allocation of treatment and enforcement in illicit drug control. *Operations Research, 49*, 352–362.
Washburn, A., & Wood, K. (1995). Two-person zero-sum games for network interdiction. *Operations Research, 43*, 243–252.
Wirl, F., Novak, A., Feichtinger, G., & Dawid, H. (1997). Indeterminacy of open-loop Nash equilibria: The ruling class versus the tabloid press. In H. G. Natke & Y. Ben-Haim (Eds.), *Uncertainty: Models and measures* (pp. 124–136). Berlin: Akademie-Verlag.

Criterion Cone

► Multiobjective Programming

Criterion Space

► Multiobjective Programming

Criterion Vector

► Multiobjective Programming

Critical Activity

A project work item on the critical path having zero float time.

See

► Critical Path
► Critical Path Method (CPM)
► Network Planning

Critical Path

The longest continuous path of activities through a project network from beginning to end. The total time elapsed on the critical path is the shortest duration of the project. The critical path will have zero float time, if a date for completion has not been specified. Any delay of activities on the critical path will cause a corresponding delay in the completion of the project. It is possible to have more than one critical path.

See

► Critical Path Method (CPM)
► Network Planning

Critical Path Method (CPM)

A project planning technique that is used for developing strategy and schedules for an undertaking using a single-time estimate for each activity of which the project is comprised. In its basic form, i.e., concerned with determining the critical path, that is, the longest sequence of activities through the project

network from beginning to end. CPM arose from a jointly sponsored venture of E.I. du Pont de Nemours and Company and the Sperry-Rand Corporation (Kelley 1961).

See

- ▶ Network Planning
- ▶ Program Evaluation and Review Technique (PERT)
- ▶ Project Management

References

Kelley, J. E. (1961). Critical-path planning and scheduling: Mathematical basis. *Operations Research, 9*, 296–320.csm.

Critical Systems Thinking

Werner Ulrich
University of Fribourg, Fribourg, Switzerland
The Open University, Milton Keynes, UK

Introduction

Critical systems thinking (CST) is a development of systems thinking that aims to support good practice of all forms of applied systems thinking and professional intervention. In its simplest definition, CST is applied systems thinking in the service of good practice. Three essential ideas are as follows:

1. Professional practice in all its stages and activities, from the formulation of problems to the implementation of solutions and the evaluation of outcomes, involves choices that need to be made transparent and require systematic examination and validation.
2. Systems thinking, although it does not protect against the need for such choices, at least offers a methodological basis for examining them systematically.
3. Consequently, applied systems thinking should make it standard practice to employ not only a hard (quantitative, scientific) and/or a soft (qualitative, interpretive) but always also a systematically critical (reflective, questioning) perspective and mode of analysis.

Taking these three elements together, CST not only recognizes that all applied systems thinking involves choices in need of critical reflection but also draws on systems thinking itself as a source of systematic critical reflection and deliberation.

CST and OR

Critical systems thinking has essential roots in operations research and management science (OR/MS), along with some equally important roots in philosophy, social theory, and other disciplines. It has applications in OR/MS as well as in many other professional fields that it is increasingly influencing; among them are environmental planning and management, public policy analysis, information systems design, social planning, evaluation research, technology assessment and risk regulation, and others. Unlike most of these fields, OR/MS was from the outset conceived as applied systems thinking; its systems perspective was to distinguish it from conventional notions of applied science and professional intervention. Critical systems thinking may be understood as an expansion of that original idea. CST's focus is on the fundamental theoretical and normative assumptions that inform the formulation and analysis of problems within their contexts, rather than on the more technical aspects of model building, analysis, and validation, or on procedural aspects of project management and consensus formation.

Two Main Sources of CST Within OR/MS

Critical systems thinking developed from the confluence of two largely independent strands of thought about OR practice. The first strand originated in the 1970s at the University of California at Berkeley and can be regarded as a development of, and response to, Churchman's (1968, 1971, 1979) philosophy of social systems design, which itself was a development of his earlier pioneering work on OR/MS (Churchman et al. 1957). The second strand originated in the 1980s at the University of Hull in England and can be regarded as a response to the development, in British OR, of soft systems methodology (Checkland 1981, 1985; Checkland and Scholes 1990), along with a number of soft OR methods or problem structuring methods (Rosenhead 1989) and

some other approaches to complex and dynamic problem contexts (e.g., management cybernetics and viable systems diagnosis, Beer 1972, 1985), all of which not only led to a growing variety of methods and underlying research paradigms but also to a perception of paradigmatic insecurity or crisis in parts of the OR profession.

Two Key Issues of Critical Systems Thinking

CST responded to these developments in American and British OR/MS by focusing its methodological efforts on two key issues:

- The first issue emerged from recognizing that the way professionals understand and define problem contexts has value implications, in the practical sense that it may do more or less justice to the different views and needs of people. Professional practice cannot avoid, in every specific context of intervention, choices as to what views (observations, data) and what needs (concerns, interests) of people are to be considered relevant and what other views and needs should not or cannot be considered equally relevant. The question is: "What should constitute the basis of knowledge and values for rational practice?" When it comes to this normative core of practice, there is a need to support professionals and everyone else concerned in handling their assumptions in a transparent and self-critical way, as well as to deal adequately with the consequences these assumptions may have for the different parties concerned.
- The second issue emerged from recognizing that different problem situations put different demands on professional competence and accordingly also on the methods professionals use. Professional practice cannot avoid, in defining and employing its methods of analysis and intervention, assumptions about the nature of problem situations, particularly with respect to the kind of complexity that matters; for real-world complexity takes different forms and there is consequently no single best way to understand and handle it. The question is: "What are the assumptions, strengths, and weaknesses of different approaches and methods regarding the nature of problem contexts, that is, different kinds of social reality? When it comes to the variety of methodological options available today in applied systems thinking, there is a need to support professionals in handling these options in a theoretically informed and justifiable way.

Critical systems thinking, then, is the use of systems ideas for probing into these two different (though not entirely independent) sources of contextual selectivity, that is, assumptions that shape the understanding and handling of problem contexts—the selection of relevant facts and values, and the selection of adequate methodologies and methods. Both shape the way problems will be understood within their contexts. However, they place rather different demands on good practice. What assumptions different systems approaches make regarding the nature and complexity of problem contexts depends on their theoretical underpinnings and thus can be identified theoretically once and for all; good practice in this respect means informed methodology choice. By contrast, relevant facts and values need to be identified anew in each specific problem situation and therefore are mainly a responsibility of practice itself; good practice in this respect means reflective practice.

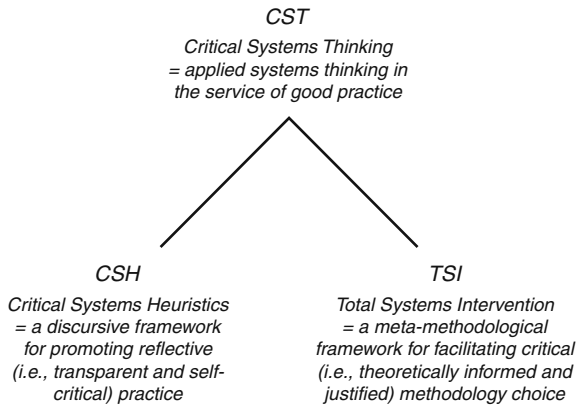
Two different strands of critical systems thinking have accordingly developed: critical systems heuristics (CSH) and total systems intervention (TSI). Their shared core idea is that systems thinking can be a useful source of critical reflection about contextual selectivity. A precise yet comprehensive definition of CST may therefore be formulated as follows.

Definition

Critical systems thinking (CST) is an application of systems thinking that aims to support good practice with regard to (a) the normative core of the knowledge and value basis that informs professional findings and conclusions and (b) the theoretical assumptions that inform the variety of methodologies and methods employed. The common denominator of (a) and (b) is that they both condition the perception of relevant problem contexts.

Terminology: CST, CSH, and TSI

The term "critical systems thinking" was coined in July 1989, when the originators of the two strands met at the 33rd Annual Conference of the International Society for the Systems Sciences (ISSS) in Edinburgh, Scotland, and decided to unite their efforts under the umbrella of critical systems



Critical Systems Thinking, Fig. 1 Critical systems thinking (CST) and its two strands—basic terminology (Source: Adapted from Ulrich 2003, 327)

thinking. Due to differing methodological conceptions and philosophical backgrounds, the cooperation between the two strands of CST remained a brief episode in the late 1980s and early 1990s; but the term CST has survived as a name for their shared interest in handling contextual assumptions critically.

Some confusion was subsequently caused by the circumstance that both strands have continued to refer to their efforts as critical systems thinking. For the sake of terminological clarity, it is advisable to use the term as a higher-level concept under which CSH and TSI may meaningfully be subsumed, rather than identifying it with either strand (Fig. 1).

Due to their separate development and also to different theoretical foundations, the two strands, despite their shared core idea and complementary ends, have brought forth partly incompatible frameworks for CST. They are therefore introduced separately. However, to facilitate comparison and synthesis, the account follows the same structure and uses the same criteria.

Critical Systems Heuristics (CSH): Facing the Normative Core of Professional Practice

CSH was fully worked out in the late 1970s at the University of California at Berkeley but became widely known only in the early 1980s, when the main theoretical work (Ulrich 1983) was published with some delay after the author's return to Switzerland. With a view to submitting his work to the test of practice, Ulrich assumed a position as chief policy analyst and evaluation researcher in the public sector and also returned to teaching at his home university, the University of Fribourg (Philosophical Faculty). This double experience in public policy making and university teaching has helped Ulrich to develop CSH continuously since. CSH has meanwhile found resonance and applications in many applied disciplines and is gradually evolving into a more comprehensive framework for reflective practice in the civil society (Ulrich 2000), critical pragmatism (Ulrich 2006 and 2007), and philosophy for professionals (Ulrich 2007).

Core Idea

Professional practice involves validity claims (e.g., to truth, rightness, sincerity, objectivity, rationality, and relevance) that have practical consequences but which it cannot fully justify. Applied systems thinking makes no exception, for its effort to appreciate the systemic nature of problems, and thus to gain a comprehensive or whole-systems view of problem situations, does not supersede the need for making value judgments as to what exactly is to be considered the problem to be dealt with (i.e., what merits improvement), what constitutes the relevant problem context (i.e., what is the sum total of the relevant facts and concerns), and wherein would consist a good solution (i.e., how to define improvement). No kind of systems methodology or other methodology can fully justify the answers to such inevitable questions as "whose problem is to be solved in the first place?" and "for whom should improvement be achieved and for whom should it not?" What is possible, however, is a conscious and careful handling of this normative core of all professional intervention.

Critical systems thinking as understood in CSH therefore begins with the idea that holistic or whole-systems thinking—the quest for comprehensiveness—is a meaningful effort but not a meaningful claim. Doing full and equal justice to the views and values of all the people concerned is an ideal; but applied systems thinking should not be expected to achieve ideals. To put it differently, holism is not a philosophically and methodologically credible source of justification, it is a problem. Hence, rather than trying to be holistic, CSH tries to support practice—professionals as well as ordinary citizens—in

appreciating the inevitable selectivity of the claims involved (e.g., to putting a problem well and to securing improvement) with regard to the facts (observations) and values (concerns) it takes to be relevant and on which its rationality and consequences depend.

In practical contexts of action, selectivity usually translates into partiality in the sense that different parties will be affected differently. CSH consequently also aims to help professionals and citizens in analyzing these consequences and how they may change if assumptions about relevant observations and concerns are modified. Good practice cannot avoid selectivity and partiality, but it will make it transparent to all those concerned how the selectivity of assumptions and the partiality of consequences depend on one another. It will give all the parties an opportunity to articulate their critique, and will then try to modify assumptions and consequences accordingly. Critical systems thinking, thus understood, is reflective practice—a methodologically disciplined effort to support such processes of critique systematically.

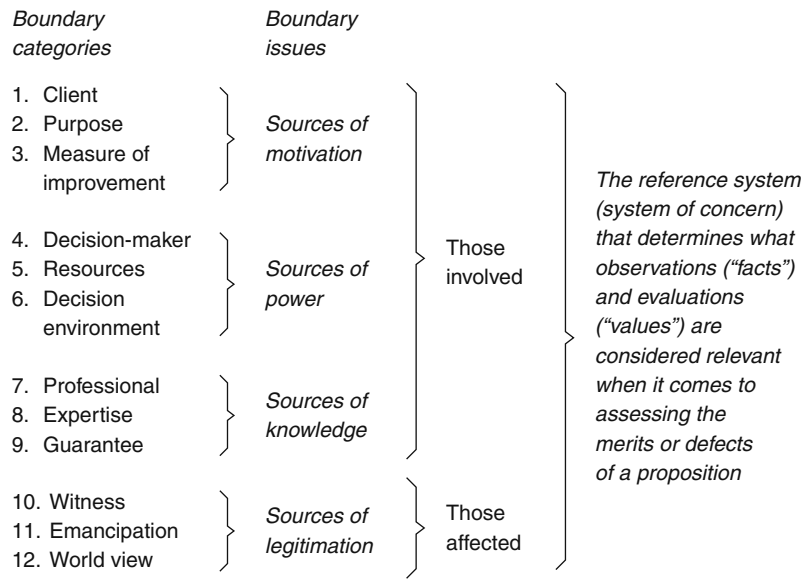
Methodological Approach

CSH is both a new philosophical foundation and a practical implementation of a discursive framework for value clarification and critique. Like the previously used concept of the normative core of rational practice, the term "value clarification" again refers to the selectivity of both considerations of facts (the empirical or knowledge basis of rational action) and of values (the normative or value basis of rational action) in contexts of practical action. The choice of the knowledge basis of professional interventions—of relevant data, judgments of fact, personal views, and other empirical conjectures (e.g., anticipated consequences of action)—has no less normative implications than does the choice of its value basis, that is, of relevant concerns, notions of improvement, and ethical standards. Both sources of selectivity and partiality demand a critical handling.

But applied systems thinking not only implies empirical and normative selectivity, it also holds a key to handling such selectivity critically. Systems thinking compels professionals, as well as everyone else concerned, to pay attention to the systems boundaries that delimit any specific system of interest. Systems thinking can thus be understood as a tool for reflecting about the boundaries of concern that (consciously or not) inform all analysis of problems and related proposals and arguments, regardless of whether systems terms are used in the first place or others. Systems thinking then becomes a source of critique—of questioning boundary assumptions and the ways they condition validity claims—rather than, as it is more usually understood, a source of justification, that is, a way of buttressing validity claims by more comprehensive considerations of fact and value.

In the terms of CSH, critical systems thinking can support professionals and all the parties concerned in identifying and questioning boundary judgments that delimit the reference systems for defining problems and relevant contexts, solution designs, evaluations, proposals for action, and so on. Boundary judgments determine for a number of basic boundary issues and related boundary categories what is to be considered and what is to be left out when it comes to defining relevant observations (judgments of fact) and concerns (judgments of value). A reference system is the set of boundary judgments that together define the context of application which a specific claim or proposal refers to and for which it is valid.

Boundary judgments are the perfect device for questioning the relevance and quality of reference systems; for unlike what one might assume at first glance, they define not just the scope of the context considered (i.e., how narrow or comprehensive it is delimitated) but equally its content, that is, what observations about that context are collected and taken to be relevant; how they are formulated, interpreted, and used; what importance is attached to them; and how well related conjectures are argued. This is so because any aspects of a problem situation that are not properly considered, say, because those involved argue incoherently or anticipate consequences incorrectly, or fail to do justice to the concerns of others, have in fact been excluded from the relevant knowledge and value basis. Even if one recognizes some aspects as relevant and agrees with others they should be considered but then fails to take them properly into account, due to lacking knowledge, to an error of judgment, or some communicative misunderstanding, or because those in control of the situation decide to suppress their discussion, the aspects are in fact (deliberately or not) excluded from the considered reference system. Thus the



Critical Systems Thinking, Fig. 2 Boundary categories of critical systems heuristics (Source: Ulrich 1983, 258)

argumentative quality of a validity claim or related discussion very well reflects itself in boundary judgments.

The main device to promote such argumentative quality is critical systems discourse, a dialogical form of boundary critique. Boundary critique is basically a systematic process of identifying the boundary judgments that are built into any specific validity claims, in an effort to unfold their normative core (selectivity) and what it may mean for the parties concerned (partiality). A second basic aim is to show that there are always options for defining boundary judgments, and to make it visible how different the claims in question may look in the light of such options. In cooperative settings where the parties are prepared to try and see whether they can agree on their boundary judgments, these can be modified accordingly. In controversial settings this may not be possible; boundary critique then gains a new meaning and consists in employing boundary judgments for critical purposes against those who are not prepared to disclose and question them or who even try to impose them on the basis of authority and power rather than argumentation. Critical systems discourse thus becomes a discursive process of challenging validity claims by demonstrating that and how they depend on boundary judgments that have not been declared or are imposed by nonargumentative means.

To be sure, selectivity, not comprehensiveness, is the fate of everyone who tries to solve problems and to do something about the state of the world. The point of boundary critique consists, in terms of CSH, in a critical turn of applied systems thinking and its notion of good professional practice. It recognizes that there is no objective but only a critical solution to the fundamental problem of practical reason, of how claims to rational practice can be justified in the face of their inevitable selectivity and partiality. The problem has remained unresolved in practical philosophy, the philosophical discipline concerned with the normative dimension of rational action, in that no theoretical solutions have been found that would at the same time be practicable. (A more complete account of the concept of a critical solution is given in Ulrich (1983, 2001, 2003).)

Methodological Core Principle

CSH's answer to the unresolved problem of practical reason is the principle of boundary critique. It says that both the meaning and the validity of claims depend on the reference system which these claims refer to and, hence, that one cannot understand and qualify (appreciate and criticize) their adequacy without examining the boundary judgments that define that reference system. The basic idea and aim of CSH, then, is to support systematic processes of boundary critique as a way to secure at least a critical solution of the problem of practical reason. To this end, there are 12 CSH boundary categories (Fig. 2).

These boundary categories stand for four crucial sources of selectivity built into all practice. Each boundary category translates into two boundary questions: one asking what is the case ("is" mapping, i.e., descriptive analysis) and the other what should be the case ("ought" mapping, i.e., normative analysis). This yields an extensive checklist of boundary questions that explicitly define the precise intent of each boundary category (Ulrich 1987, 1996, 2000; Ulrich and Reynolds 2010). They can be used, first, to identify boundary judgments systematically; second, to examine how alternative boundary judgments may change the way one sees problem definitions, findings, and conclusions, and thus what is considered to be adequate and rational; and third, to challenge any claims to knowledge, rationality, or improvement that rely on hidden boundary judgments or take them for granted.

The last-mentioned application leads to an argumentative employment of boundary judgments, known as polemical or emancipatory boundary critique, that creates an improved symmetry of critical competence among all the parties concerned, professionals and citizens alike, regardless of their theoretical knowledge or special expertise with respect to the problem at issue. As a practicable model of cogent critical argumentation (Ulrich 1983, 1993, 2000), it embodies a critical pragmatization of Habermas' (1973, 1979) ideal model of rational practical discourse (a model that underpins his discourse ethics and confines it to being a moral theory rather than a practicable model of moral justification). It constitutes a chief methodological backing of the critical turn of the concept of rational practice proposed above.

In sum, CSH can be defined as a methodological framework for boundary critique, that is, for identifying and debating boundary judgments, with the aim of securing at least a critical solution to the unsolved problem of practical reason—the question of how claims to rational practice can be justified despite the unavoidable selectivity and partiality of all practice. Despite its emancipatory implications (the aspect for which it is best known), CSH should not be misunderstood and used as an emancipatory systems approach only; its principle of systematic boundary critique is vital for sound professional practice in general, whatever importance may be attached to emancipatory issues. For the same reason, CSH does not aim to be a self-contained systems methodology, but is better understood as an approach that should inform all critical professional practice, whatever specific methodology is used.

Practical Implementation (Main Procedure)

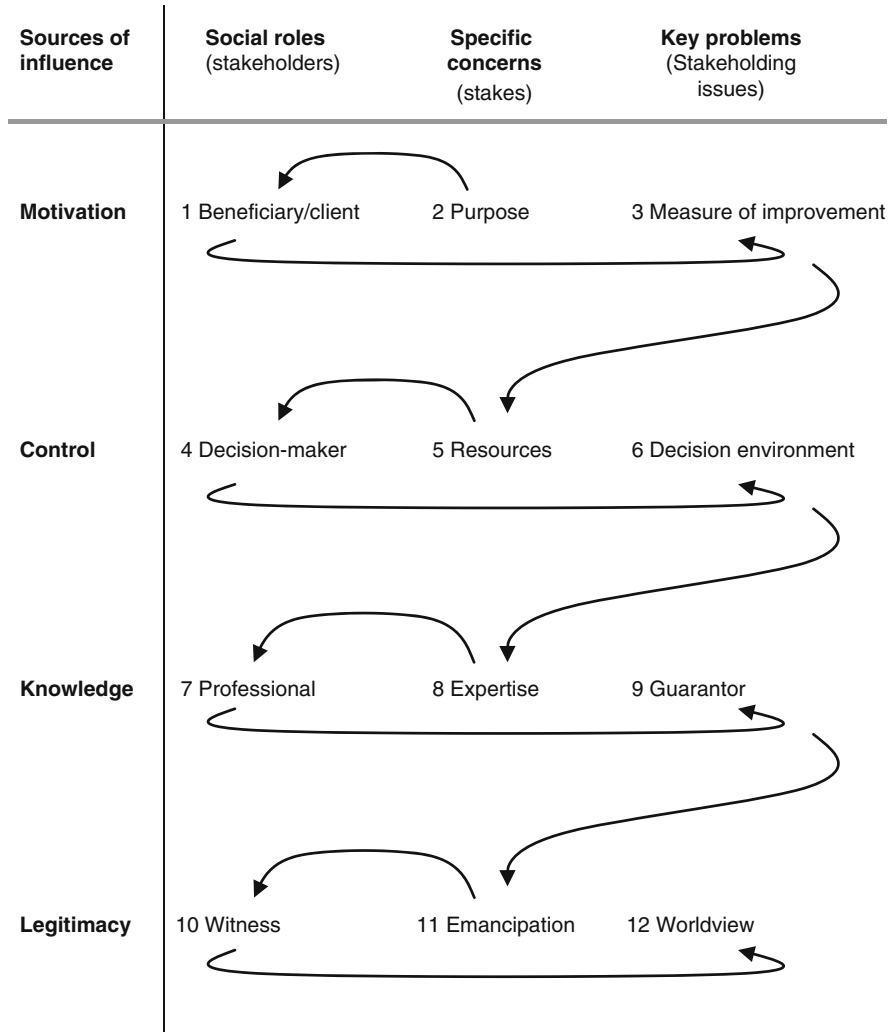
Boundary critique is best implemented as an iterative process of reflecting on, and discussing, the implications of alternative boundary judgments. When some boundary judgment changes, the reference system of which it is constitutive will change too; consequently, all other boundary judgments may need being reconsidered and adapted. However, iterative processes are not particularly easy to learn and to handle; experience with boundary critique suggests that it is useful for beginners to have available, and follow, a standard sequence for unfolding the boundary categories and questions of CSH (Fig. 3).

Total Systems Intervention (TSI) or Creative Holism (CH): Ensuring Informed Methodology Choice

TSI stems from work done at the University of Hull, England, in the mid and late 1980s, about the evolution of OR and systems thinking in terms of changing underlying theoretical assumptions. This work resulted in the early 1990s in the proposal of a meta-methodology for choosing among methodologies according to situational requirements (Flood and Jackson 1991; Jackson 1991). By that time CSH and TSI had joined their efforts under the new name of "critical systems thinking" (CST), after previously using different names such as critical systems approach (CSH) and critical management science (TSI). But due to differing notions of what critical practice was to mean, the two strands of CST ultimately found it difficult to integrate their approaches and consequently returned to developing their frameworks separately. Both have nevertheless continued to use the name critical systems thinking. Meanwhile, Jackson (2003; 2006b) refers to his work on critical systems thinking and practice as creative holism (CH).

Core Idea

Applied systems thinking depends for its choice of systems methodologies and methods on basic assumptions regarding the nature of the problem contexts (typically: organizational contexts) with



Critical Systems Thinking, Fig. 3 CSH's process of unfolding: a standard sequence of boundary critique (Source: Ulrich and Reynolds 2010, 259; Adapted from Reynolds 2007, 106)

which it is dealing. Some of these assumptions can usefully be captured in terms of a number of sociological paradigms for describing the nature of social reality as they have been analyzed, for example, by Burrell and Morgan (1979), as well as by organizational images or systems metaphors as they have been described most systematically by Morgan (1986). Different systems methodologies, because they usually are developed with different problem contexts in mind, can similarly be characterized in terms of underlying metaphors and paradigms. Hence, since the characteristics of both problem contexts and systems methodologies can be captured in terms of adequate paradigms and metaphors, it becomes possible to match contexts and methodologies in a systematic way and thus to support professionals in choosing among the increasing number of available systems methodologies and conforming methods that are best suited to deal with a problem situation at hand.

CST as understood in TSI/CH therefore begins with the idea that systems thinking—the attempt to understand organizational or societal problem contexts in systems terms—is meaningful only to the extent people are aware of the sociological paradigms and organizational metaphors that inform it. Since different systems methodologies rely on different paradigms and metaphors—that is, on different theoretical assumptions about the nature of problem contexts—applied systems thinking depends for its justification and rationality on paradigmatic fit between systems methodologies and problem contexts.

In applied OR/MS, as in other forms of applied research, the requirement of paradigmatic fit translates into a need for informing the selection and use of methodologies and methods by previous paradigm analysis as well as, where relevant, metaphor analysis, as a condition for doing justice to the nature of the problem context at issue. TSI/CH consequently puts its critical focus on the theoretical underpinnings of alternative research paradigms rather than on the normative core of professional practice, as does CSH. CST, thus understood, is about methodology choice—a theoretically informed way to support processes of matching methodologies and methods with problem contexts.

Methodological Approach

The basic strategy of TSI/CH can be described as a contingency approach to methodology choice, based on paradigm analysis and, to a lesser degree, also on metaphor analysis of the three major traditions of systems thinking thus far—hard, soft, and critical systems thinking. The idea is that there is no such thing as a best systems methodology and underpinning tradition of systems thinking; rather, situational aspects of the problem context at hand determine what tradition of systems thinking is best suited as a source of methodological guidance and specific methods or tools of intervention. In OR/MS, such an approach promises to resolve the OR in crisis debate of the 1970s and 1980s, for it allows hard and soft OR approaches to be seen as appropriate for dealing with different problem contexts rather than competing for the same ones.

Contingency frameworks are also called contingency theories, as they involve theoretical generalizations about the crucial aspects of the application domain to which the framework is to be applied. This theoretical device is often used in the social sciences (e.g., in management and organization theories) when a variety of approaches are required to handle a given class of problems, as the proper approach is dependent (contingent) on the situation or, more precisely, on a range of changing situations.

Applied to contexts of professional intervention, using a contingency approach implies that some independent (contextual) variables can be identified empirically which regularly, for reasons that can be explained theoretically, may be expected to condition the outcome of interventions. A contingency approach can then (and only then) make sure that the way one deals with a situation matches situational requirements and, on that basis, can also justify the credibility of the results. To the extent this condition is fulfilled, one can properly speak of a contingency theory. It follows that the crucial question for any contingency approach is whether it can identify and theoretically justify a small number of empirical dimensions (ideally only two) in terms of which the range of situations in question can be usefully classified, so that each type of empirical situation can then be identified in a relevant and reliable way.

Methodological Core Principle

TSI/CH's answer to the problem of ensuring paradigmatic fit between intervention approaches and problem contexts is a classification of problem contexts, and of systems methodologies assigned to them, called the system of systems methodologies (SOSM). It says that systems methodologies and conforming methods are well chosen if their underlying systems metaphor (machine, organism, etc.) and/or paradigm (functionalist, interpretive, etc.) match with the nature of the problem context, or more exactly, with assumptions about the kind of complexity that needs to be handled in the problem context. The basic idea and aim of TSI/CH, then, is to support systematic processes of informed methodology choice, as a way to secure paradigmatic fit between intervention methods and intervention contexts. To this end, TSI/CH proposes the SOSM (see Fig. 4).

There was an earlier, four-celled version of the SOSM (Jackson and Keys 1984) that is now often cited as the origin of the TSI strand of CST. However, it only distinguished hard and soft methodologies, and its discussion in that early paper did not yet introduce the notion of critical systems thinking.

CSH became known to Jackson and Keys shortly after publishing their 1984 paper. First hints at a planned extension of their work appeared in a few articles in the late 1980s (Jackson 1987, 1990); the extended SOSM was presented later in Flood and Jackson (1991) and Jackson (1991).

Due to the underlying logic of the SOSM, the extended scheme could not manage to include CSH except by constricting its notion of critical systems thinking considerably. This logic assumes that any

Systems dimension of contexts (increasing complexity)	Participants dimension of contexts (increasing diversity of values)		
	Unitary (paradigm: functional) **HARD SYSTEMS THINKING**	*Pluralist (paradigm: interpretive)* **SOFT SYSTEMS THINKING**	*Coercive (paradigm: emancipatory)* **EMANCIPATORY SYSTEMS THINKING**
Simple	*Simple-unitary problem contexts (systems metaphor: machine)* • Operations research (OR) • Systems engineering (SE) • Systems analysis (SA)	*Simple-pluralist problem contexts (systems metaphors: culture, coalition)* • Systems approach (Churchman) • Strategic assumption surfacing and testing (SAST)	*Simple-coercive problem contexts (systems metaphor: prison)* • Critical systems heuristics (CSH)
Complex	*Complex-unitary problem contexts (systems metaphors: organism, brain)* • Organizational cybernetics/ viable systems diagnosis (VSD) • Socio-technical systems thinking	*Complex-pluralist problem contexts (systems metaphors: culture, coalition)* • Interactive planning (Ackoff) • Soft systems methodology (SSM)	*Complex-coercive problem contexts (systems metaphor: prison)* • ?

Critical Systems Thinking, Fig. 4 The extended system of systems methodologies (SOSM) (Source: Adapted from Flood and Jackson 1991, 42; Jackson 1991, 29 and 31; 2000, 359)

methodology can be meaningfully assigned to a single type of problem context and to a conforming (dominant) theoretical paradigm. There is no room in such a scheme for an approach that focuses on the genuinely normative core of practice as such, whatever the theoretical paradigm adopted may be and the choice of methodology and conforming methods it may inspire. This makes it understandable why the extended SOSM rather arbitrarily assigned CSH a merely emancipatory purpose, as opposed to the critical purpose of the SOSM. To render this choice more plausible, CSH was associated with a prison metaphor, which then seemed to render CSH adequate for coercive problem contexts only and thus provided a rationale for assigning it to a specific emancipatory paradigm (for critical discussion and alternatives, see Ulrich 2003). In this way, CSH became in the SOSM scheme an apparently self-contained methodology that, quite against its original intentions, was to be chosen (or not) as an alternative to soft and hard systems methodologies. Its concern for the practical-normative side of all practice thus moved out of sight.

In British OR/MS, CSH was henceforth understood mainly through the lens of the SOSM, and critical systems thinking became widely identified with TSI. Consequently, CST was now almost the same as the SOSM—an updated contingency framework for methodology choice, as well as for continuing discussions about the evolution of OR/MS (e.g., Jackson 2006a). Both uses attracted much interest and the mentioned difficulties of the extended SOSM did not hamper its success in helping to raise awareness in the profession that there are options for conceiving of good professional practice. The discussion that the SOSM was able to generate in turn has helped to make CSH more known, so that its core principle of boundary critique is increasingly being recognized as an important, independent source of critical thought on practice. These diverse successes of the SOSM certainly have contributed to the comparatively high level of methodological awareness and discussion by which the OR/MS profession distinguishes itself from other fields, which has allowed it to pioneer soft and critical systems ideas that are now radiating into many other fields.

Practical Implementation (Main Procedure)

To support methodology choice in practice, the SOSM still needed to be embedded in a methodology, properly speaking, that is, a framework that would guide practitioners in asking relevant questions and proceeding systematically. This is what total systems intervention (TSI), a name adopted in 1991, is all about. It stands for the practical procedure of methodology choice and implementation that Flood and Jackson (1991) proposed on the basis of the SOSM. The aim is to provide a meta-methodology for methodology choice and implementation. The procedure may be employed in a linear or iterative way. Originally it consisted of three phases labeled creativity, choice, and implementation, to which Jackson (2003, 2006b) later, in the extended

Critical Systems Thinking, Table 1 The meta-methodology of TSI/CH: standard phases of methodology choice and use

Phase	Activity/Aim
(1) CREATIVITY	
Task	To identify major aims and issues of the problem context
Tools	Use of different metaphors and paradigms to gain different perspectives
Outcome	Appreciation of dominant and dependent metaphors/paradigms and related issues
(2) CHOICE	
Task	To choose appropriate systems methodologies and methods
Tools	Use of SOSM to reveal strengths and weaknesses of methodologies and methods
Outcome	Choice of dominant and dependent systems methodologies and methods
(3) IMPLEMENTATION	
Task	To arrive at and implement specific positive change proposals
Tools	Systems methodologies and methods used properly according to the logic of TSI/CH
Outcome	Relevant change according to the concerns of the different paradigms
(4) REFLECTION	
Task	To evaluate the intervention and ensure methodological learning
Tools	Understanding of the concerns of different paradigms regarding good practice
Outcome	Methodological progress

TSI total systems intervention = phases 1–3, *CH* creative holism = phases 1–4, *SOSM* system of systems methodologies
(Source: Adapted from Flood and Jackson 1991, 54; Jackson 1991, 276; 2000, 372; and 2006b, 654)

version he now prefers to call creative holism, added a fourth phase, Reflection (Table 1).

The creativity phase is intended to encourage consideration of what alternative systems paradigms and root metaphors might mean for thinking about a problem context at hand, so that a dominant metaphor could be identified as most adequate, that is, in effect, preference for a hard (functionalist), soft (interpretive), or critical (emancipatory) orientation. In the choice and implementation phases, a conforming particular systems methodology should then be chosen based on the SOSM and used to implement specific change proposals.

A new element in CH as compared to its predecessor TSI is the reflection phase, which brings in an element of reflective practice as CSH understands it, by looking at the outcomes of methodology choice and implementation rather than at its theoretical justification only. Although the underlying notion of evaluation is still not genuinely practical in the sense of CSH and practical philosophy, this development does promise to open up new chances for reflective practice.

Another new element, following a considerable amount of discussion in the literature about methodological complementarism or pluralism (Jackson 1997, 1999), mixing methods (Midgley 1997), and multi-methodology (Mingers and Gill 1997), is that creative holism, unlike TSI, no longer insists on choosing a single dominant paradigm. Instead, a combination of methodologies, or parts of methodologies and conforming methods, is now encouraged, which makes the framework more flexible and brings it closer to actual practice. As Jackson describes it, CH now is a "meta-methodological" framework that aims to help practitioners to "harness the various systems methodologies, methods and models" by being "multi-paradigm, multi-methodology and multi-method in orientation" (Jackson 2006b, 248 and 253).

A Summary Comparison of CSH and TSI

To provide an overview of the discussed aspects of critical systems thinking, Table 2 summarizes the accounts of CSH and TSI in a way that should facilitate comparison.

Concluding Remarks

The claim of professional practice to relevance, rigor, and rationality depends on many requirements. Among these, two crucial ones are putting the problem well

Critical Systems Thinking, Table 2 CSH and TSI compared

Aspect	CSH	TSI/CH
Core idea	Professional practice involves *validity claims* that cannot be justified theoretically but at least can be handled openly and critically in the process of intervention itself	Professional practice involves *methodological choices* that can be justified theoretically by analyzing underpinning research paradigms and systems metaphors
Critical focus	*Reflective practice:* surfacing the reference systems underpinning all judgments of fact and value and analyzing how they condition practical claims (e.g., problem definitions, relevant contexts, standards of improvement, and proposals for action)	*Paradigm analysis:* surfacing the theoretical underpinnings of alternative research paradigms (e.g., functionalist, interpretive, emancipatory, or post-modern) and analyzing how they condition different perceptions of problem contexts and suitable methodological choices
Approach	*Critical systems discourse:* a discursive framework for value clarification and critique	*Contingency theory:* a contingency framework for methodology choice and use
Methodological core principle	*Boundary critique:* unfolding the selectivity of reference systems	*Informed methodology choice:* matching systems methodologies with problem contexts
Main critical device	*Checklist of boundary questions:* a definition of boundary categories for "is" and "ought" mapping (i.e., descriptive and normative analysis) of reference systems	*System of systems methodologies* (SOSM): a classification of problem contexts and conforming systems methodologies
Implementation	A discursive *process of unfolding selectivity:* a standard sequence of boundary critique	A holistic *meta-methodology of paradigm analysis:* standard phases of methodology choice and reflection

CSH critical systems heuristics, *TSI/CH* total systems intervention/creative holism

and tackling it by means of adequate methods. In different ways, both of these embody crucial requirements of professional competence. Both of them stand for efforts to make sure that relevant issues are properly identified and the implications of related assumptions are made transparent and evaluated.

- *Putting problems well* is an issue that involves empirical (observational) as well as normative (ethical) problem structuring and reflection. The selection of relevant facts and values depends on a proper understanding of the problem, which is hardly achievable without questioning the scope and diversity of the social context that matters. It also depends on the extent to which justice is done in practice to the diversity of views and concerns of the different parties concerned. A problem may be ill defined so long as this normative core of any quest for rational practice is not well understood.
- *Choosing and employing methods properly* involves analysis and reflection about the demands of problem situations on the one hand and about the availability of methods that respond to these demands on the other. The selection of adequate methodologies and methods depends on a proper understanding of the theoretical and paradigmatic assumptions involved, which is hardly achievable without questioning the nature of the complexity that matters. It also depends on the extent to which the matching of such assumptions with specific situations is successful in practice. A methodology and conforming methods may be ill chosen so long as this theoretical core of the quest for rational practice is not well understood.

Neither effort replaces or precludes the other. Critical systems thinking, properly understood, aims to promote good practice with regard to both. To this end, the two strands of CST bring to bear within the field of OR/MS, and in the applied sciences in general, new philosophical and theoretical foundations, along with new practical tools for analyzing contextual complexity and diversity. CSH draws on practical philosophy and consequently conceives of rational practice in terms of discursive tools of value clarification and critique, in particular boundary critique and discourse. TSI/CH draws on organizational sociology and conceives of rational practice in terms of theoretically informed tools of methodology choice, in particular paradigm analysis and metaphor analysis.

Different as the resulting frameworks of CSH and TSI are, their shared concern remains the idea that

good professional practice depends crucially on making sure that problems are well put and methods of intervention are well chosen; and that to meet both requirements, it is essential to properly situate problems in their contexts and make sure one understands those contexts well. Formulated in everyday terms, the essential message of CST to professionals might thus be summarized as follows:

Critical Systems Thinking: Its Operational Imperative

As a professional intervening in a specific context, pay attention to your contextual assumptions and try to identify and examine them systematically, so as to understand them well. Then make sure everyone concerned understands them well too. Work toward mutual understanding about how problem definitions and solutions depend on and change with the facts and values considered relevant. Make sure divergent views and values are properly addressed. Adapt your choice of methodologies and methods to the amount of diversity that you find in the problem context, and to the resulting nature of the complexity that matters. Finally, whatever problem definitions and methods your professional practice ultimately relies on, reflect on the validity claims your professional findings and conclusions imply and how, if taken as a basis for action, they may affect the different parties concerned. Make boundary critique a standard practice to this end, and always remember that no professional intervention can do justice to all views and values, that is, can justify all its implications. But at least it can deal with this inevitable lack of complete justification in a transparent and self-reflecting way. This is what critical professional practice is all about.

See

- ▶ Community OR
- ▶ Cybernetics and Complex Adaptive Systems
- ▶ Practice of Operations Research and Management Science
- ▶ Problem Structuring Methods
- ▶ Soft Systems Methodology
- ▶ System Dynamics
- ▶ Systems Analysis

References

Beer, S. (1972). *Brain of the firm* (2nd ed. Chichester: Wiley, 1981). Harmondsworth: Penguin Press.

Beer, S. (1985). *Diagnosing the system for organizations*. Chichester: Wiley.

Burrell, G., & Morgan, G. (1979). *Sociological paradigms and organizational analysis: Elements of the sociology of corporate life*. London: Heinemann.

Checkland, P. (1981). *Systems thinking, systems practice*. Chichester: Wiley.

Checkland, P. (1985). From optimizing to learning: A development of systems thinking for the 1990s. *Journal of the Operational Research Society, 36*, 757–767.

Checkland, P., & Scholes, J. (1990). *Soft systems methodology in action*. Chichester: Wiley.

Churchman, C. W. (1968). *The systems approach*. New York: Dell Publishing.

Churchman, C. W. (1971). *The design of inquiring systems: Basic concepts of systems and organization*. New York: Basic Books.

Churchman, C. W. (1979). *The systems approach and its enemies*. New York: Basic Books.

Churchman, C. W., Ackoff, R. L., & Arnoff, E. L. (1957). *Introduction to operations research*. New York/London: Wiley/Chapman & Hall.

Flood, R. L., & Jackson, M. C. (1991). *Creative problem solving: Total systems intervention*. Chichester: Wiley.

Habermas, J. (1973). Wahrheitstheorien. In H. Fahrenbach (Ed.), *Wirklichkeit und Reflexion: Walter Schulz zum. 60 Geburtstag* (pp. 211–265). Neske: Pfullingen.

Habermas, J. (1979). What is universal pragmatics? In J. Habermas (Ed.), *Communication and the evolution of society* (pp. 1–68). Boston: Beacon Press.

Jackson, M. C. (1987). New directions in management science. In M. C. Jackson & P. Keys (Eds.), *New directions in management science* (pp. 133–164). Aldershot: Gower.

Jackson, M. C. (1990). Beyond a system of systems methodologies. *Journal of the Operational Research Society, 41*, 657–668.

Jackson, M. C. (1991). *Systems methodology for the management sciences*. New York: Plenum.

Jackson, M. C. (1997). Pluralism in systems thinking and practice. In J. Mingers & A. Gill (Eds.), *Multimethodology: The theory and practice of integrating management science methodologies* (pp. 347–378). Chichester: Wiley.

Jackson, M. C. (1999). Towards coherent pluralism in management science. *Journal of the Operational Research Society, 50*, 12–22.

Jackson, M. C. (2000). *Systems approaches to management*. New York: Kluwer/Plenum.

Jackson, M. C. (2003). *Systems thinking: Creative holism for managers*. Chichester: Wiley.

Jackson, M. C. (2006a). Beyond problem structuring methods: Reinventing the future of OR/MS. *Journal of the Operational Research Society, 57*, 868–878.
Jackson, M. C. (2006b). Creative holism: A critical systems approach to complex problem situations. *Systems Research and Behavioral Science, 23*, 647–657.
Jackson, M. C., & Keys, P. (1984). Towards a system of system methodologies. *Journal of the Operational Research Society, 35*, 473–486.
Midgley, G. (1997). Mixing methods: Developing systemic intervention. In J. Mingers & A. Gill (Eds.), *Multimethodology: The theory and practice of integrating management science methodologies* (pp. 249–290). Chichester: Wiley.
Mingers, J., & Gill, A. (Eds.). (1997). *Multimethodology: The theory and practice of integrating management science methodologies*. Chichester: Wiley.
Morgan, G. (1986). *Images of organization* (3rd ed. 2006). Beverly Hills, CA: Sage.
Reynolds, M. (2007). Evaluation based on critical systems heuristics. In B. Wiliams & I. Imam (Eds.), *Systems concepts in evaluation: An expert anthology* (pp. 101–122). Point Reyes, CA: Edge Press.
Rosenhead, J. (Ed.). (1989). *Rational analysis for a problematic world: problem structuring methods for complexity, uncertainty and conflict*. Chichester: Wiley (Revised edition: Rosenhead, J., & Mingers, J. (Eds.). (2001). *Rational analysis for a problematic world revisited: Problem structuring methods for complexity, uncertainty and conflict*. Chichester: Wiley).
Ulrich, W. (1983). *Critical heuristics of social planning: A new approach to practical philosophy*. Bern: Paul Haupt. Reprinted Chichester: Wiley (1994).
Ulrich, W. (1987). Critical heuristics of social systems design. *European Journal of Operational Research, 31*, 276–283.
Ulrich, W. (1993). Some difficulties of ecological thinking, considered from a critical systems perspective: A plea for critical holism. *Systems Practice, 6*, 583–611.
Ulrich, W. (1996). *A primer to critical systems heuristics for action researchers*. Hull: Centre for Systems Studies, University of Hull.
Ulrich, W. (2000). Reflective practice in the civil society: The contribution of critically systemic thinking. *Reflective Practice, 1*, 247–268.
Ulrich, W. (2001). The quest for competence in systemic research and practice. *Systems Research and Behavioral Science, 18*, 3–28.
Ulrich, W. (2003). Beyond methodology choice: Critical systems thinking as critically systemic discourse. *Journal of the Operational Research Society, 54*, 325–342.
Ulrich, W. (2006). Critical pragmatism: A new approach to professional and business ethics. In L. Zsolnai (Ed.), *Interdisciplinary yearbook of business ethics* (Vol. 1, pp. 53–85). Oxford: Peter Lang.
Ulrich, W. (2007). Philosophy for professionals: Towards critical pragmatism. *Journal of the Operational Research Society, 58*, 1109–1113.
Ulrich, W., & Reynolds, M. (2010). Critical systems heuristics. In M. Reynolds & S. Holwell (Eds.), *Systems approaches to managing change: A practical guide* (pp. 243–292). London: Springer (in association with The Open University, Milton Keynes, UK).

Cross-Entropy Method

Dirk P. Kroese[1], Reuven Y. Rubinstein[2], Izack Cohen[2], Sergey Porotsky[3] and Thomas Taimre[1]
[1]The University of Queensland, Brisbane, Australia
[2]Technion – Israel Institute of Technology, Haifa, Israel
[3]A.L.D. Ltd., Tel-Aviv, Israel

Introduction

The cross-entropy (CE) method is a versatile Monte Carlo technique introduced by Rubinstein (1999, 2001), extending earlier work on variance minimization (Rubinstein 1997). A tutorial on the CE method is given in de Boer et al. (2005). A comprehensive treatment can be found in Rubinstein and Kroese (2004); see also Rubinstein and Kroese (2007, Chap. 8).

The CE method can be applied to two types of problems:

1. *Estimation*: Estimate $\ell = \mathbb{E}[H(\mathbf{X})]$, where $\mathbf{X}$ is a random object taking values in some set $\mathcal{X}$ and H is a function on $\mathcal{X}$. An important special case is the estimation of a probability $\ell = \mathbb{P}(S(X) \geq \gamma)$, where S is another function on $\mathcal{X}$.
2. *Optimization*: Optimize (i.e., maximize or minimize) $S(\mathbf{x})$ over all $\mathbf{x} \in \mathcal{X}$, where S is some objective function on $\mathcal{X}$.

In the estimation setting, the CE method can be viewed as an adaptive importance sampling procedure that uses the cross-entropy or Kullback–Leibler divergence as a measure of closeness between two sampling distributions. In the optimization setting, the optimization problem is first translated into a rare-event estimation problem, and then the CE method for estimation is used as an adaptive algorithm to locate the optimum.

Estimation

Consider the estimation of

$$\ell = \mathbb{E}_f[H(\mathrm{X})] = \int H(\mathrm{x}) f(\mathrm{x})\, \mathrm{dx}, \qquad (1)$$

where H is a real-valued function and f is the probability density function (pdf) of the random vector $\mathbf{X}$. It is assumed, for simplicity, that $\mathbf{X}$ is a continuous random variable. For the discrete case, replace the integral in (1) by a sum. Let g be another pdf—which must be nonzero for every $\mathbf{x}$ for which $H(\mathbf{x})\, f(\mathbf{x}) \neq 0$. Using the pdf g, ℓ can be represented as

$$\ell = \int H(\mathbf{x}) \frac{f(\mathbf{x})}{g(\mathbf{x})} g(\mathbf{x}) \mathrm{d}\mathbf{x} = \mathbb{E}_g \left[H(\mathbf{X}) \frac{f(\mathbf{X})}{g(\mathbf{X})} \right], \quad (2)$$

where the subscript g indicates that the expectation is taken with respect to g rather than f. Consequently, if $\mathbf{X}_1, \ldots, \mathbf{X}_N$ are independent random vectors with pdf g, written as $\mathbf{X}_1, \ldots, \mathbf{X}_N \sim_{\text{iid}} g$, then

$$\hat{\ell} = \frac{1}{N} \sum_{k=1}^{N} H(\mathbf{X}_k) \frac{f(\mathbf{X}_k)}{g(\mathbf{X}_k)} \quad (3)$$

is an unbiased estimator of ℓ—a so-called importance sampling estimator. The optimal importance sampling pdf, that is, the pdf g^* that minimizes the variance of $\hat{\ell}$, is proportional to $|H|\, f$ (see, e.g., Rubinstein and Kroese (2007, 132)), but is in general difficult to evaluate. The idea of the CE method is to choose the importance sampling pdf g in a specified class of pdfs such that the Kullback–Leibler divergence between the optimal importance sampling pdf g^* and g is minimal. The Kullback–Leibler divergence between two pdfs g and h is given by

$$\begin{aligned} \mathcal{D}(g,h) &= \mathbb{E}_g \left[\ln \frac{g(\mathbf{X})}{h(\mathbf{X})} \right] = \int g(\mathbf{x}) \ln \frac{g(\mathbf{x})}{h(\mathbf{x})} \, \mathrm{d}\mathbf{x} \\ &= \int g(\mathbf{x}) \ln g(\mathbf{x}) \, \mathrm{d}\mathbf{x} - \int g(\mathbf{x}) \ln h(\mathbf{x}) \, \mathrm{d}\mathbf{x}. \end{aligned} \quad (4)$$

In most cases of interest the function H is nonnegative, and the nominal pdf f is parameterized by a finite-dimensional vector $\mathbf{u}$; that is, $f(\mathbf{x}) = f(\mathbf{x}; \mathbf{u})$. It is then customary to choose the importance sampling pdf g in the same family of pdfs; thus, $g(\mathbf{x}) = f(\mathbf{x}; \mathbf{v})$ for some reference parameter $\mathbf{v}$. The CE minimization procedure then reduces to finding an optimal reference parameter vector, say $\mathbf{v}^*$, by cross-entropy minimization:

$$\begin{aligned} \mathbf{v}^* &= \operatorname*{argmin}_{\mathbf{v}} \mathcal{D}(g^*, f(\cdot; \mathbf{v})) \\ &= \operatorname*{argmax}_{\mathbf{v}} \int H(\mathbf{x}) f(\mathbf{x}; \mathbf{u}) \ln f(\mathbf{x}; \mathbf{v}) \, \mathrm{d}\mathbf{x} \\ &= \operatorname*{argmax}_{\mathbf{v}} E_{\mathbf{u}} H(\mathbf{X}) \ln f(\mathbf{X}; \mathbf{v}) \\ &= \operatorname*{argmax}_{\mathbf{v}} E_{\mathbf{w}} H(\mathbf{X}) \ln f(\mathbf{X}; \mathbf{v}) \frac{f(\mathbf{X}; \mathbf{u})}{f(\mathbf{X}; \mathbf{w})}, \end{aligned} \quad (5)$$

where $\mathbf{w}$ is any reference parameter. This $\mathbf{v}^*$ can be estimated via the stochastic counterpart of (5):

$$\hat{\mathrm{v}} = \operatorname*{argmax}_{\mathrm{v}} \frac{1}{N} \sum_{k=1}^{N} H(\mathrm{X}_k) \frac{f(\mathrm{X}_k; \mathrm{u})}{f(\mathrm{X}_k; \mathrm{w})} \ln f(\mathrm{X}_k; \mathrm{v}), \quad (6)$$

where $\mathbf{X}_1, \ldots, \mathbf{X}_N \sim_{\text{iid}} f(\cdot; \mathbf{w})$. The optimal parameter $\hat{\mathrm{v}}$ in (6) can often be obtained in explicit form, in particular when the class of sampling distributions forms an exponential family; see, for example, Rubinstein and Kroese (2007, 319–320). Indeed, analytical updating formulas can be found whenever explicit expressions for the maximal likelihood estimators of the parameters can be found, cf. de Boer et al. (2005, 36).

Example: Exponential Random Variables. Consider the case where $\mathbf{X}_1 = (X_1, \ldots, X_n)$ is a vector of independent exponential random variables with expectations $u_1, \ldots, u_n$. Let $\mathbf{u} = (u_1, \ldots, u_n)$ and let $\mathbf{v} = (v_1, \ldots, v_n)$ be the reference parameter of the importance sampling pdf $f(\mathbf{x}; \mathbf{v})$, given by

$$f(\mathbf{x}, \mathbf{v}) = \prod_{i=1}^{n} \frac{\mathrm{e}^{-x_i / v_i}}{v_i}.$$

Hence, under this importance sampling pdf, $X_1, \ldots, X_n$ are again independent and exponentially distributed, but now with expectations $v_1, \ldots, v_n$. Writing $H_k = H(\mathbf{X}_k)$ and the likelihood ratio $W_k = f(\mathbf{X}_k; \mathbf{u}) / f(\mathbf{X}_k; \mathbf{w})$ in (6), the optimal parameter $\hat{\mathrm{v}}$ is found by maximizing

$$\begin{aligned} &\sum_{i=1}^{n} \sum_{k=1}^{N} H_k W_k \ln f(\mathbf{X}_k; \mathbf{u}) \\ &= \sum_{i=1}^{n} \sum_{k=1}^{N} H_k W_k \left(\frac{-X_{ki}}{v_i} - \ln v_i \right), \end{aligned} \quad (7)$$

where X_{ki} is the i-th component of $\mathbf{X}_k$. This maximum can be found by differentiating and equating to zero the right-hand side of (7) for each v_i, resulting in the equations

$$\sum_{k=1}^{N} H_k W_k \left(\frac{X_{ki}}{v_i^2} - \frac{1}{v_i} \right) = 0, \quad i = 1, \ldots, n,$$

from which it follows that

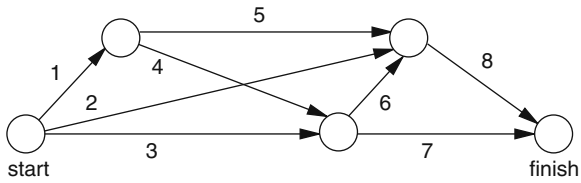
$$\hat{v}_i = \frac{\sum_{k=1}^{N} H_k W_k X_{ki}}{\sum_{k=1}^{N} H_k W_k}, \quad i = 1, \ldots, n. \tag{8}$$

Often $\ell = \mathbb{P}(S(\mathrm{X}) \geqslant \gamma)$ for some function S and level γ, in which case $H(\mathbf{x})$ takes the form of an indicator function: $H(\mathrm{x}) = I_{\{S(\mathrm{x}) \geqslant \gamma\}}$; that is, $H(\mathbf{x}) = 1$ if $S(\mathbf{x}) \geqslant \gamma$, and 0 otherwise. A complication in solving (6) occurs when ℓ is a rare-event probability; that is, a very small probability (say less than 10^{-4}). Then, for moderate sample size N, most or all of the values $H(\mathbf{X}_k)$ in (6) are zero, and the maximization problem becomes useless. In that case a multilevel CE procedure is used, where a sequence of reference parameters and levels is constructed with the goal that the former converges to $\mathbf{v}^*$ and the latter to γ. This leads to the following algorithm; see, for example, Rubinstein and Kroese (2007, 238).

Algorithm 1 (CE Algorithm for Rare-Event Estimation).

1. *Define* $\hat{\mathbf{v}}_0 = \mathbf{u}$. *Let* $N^{\mathrm{e}} = \lceil \varrho N \rceil$. *Set* $t = 1$ (*iteration counter*).
2. *Generate* $\mathbf{X}_1, \ldots, \mathbf{X}_N \sim_{\mathrm{iid}} f(\cdot; \hat{\mathbf{v}}_{t-1})$. *Calculate* $S_i = S(\mathbf{X}_i)$ *for all* i, *and order these from smallest to largest:* $S_{(1)} \leq \ldots \leq S_{(N)}$. *Let* $\hat{\gamma}_t$ *be the sample* $(1 - \varrho)$*-quantile of performances; that is,* $\hat{\gamma}_t = S_{(N - N^{\mathrm{e}} + 1)}$. *If* $\hat{\gamma}_t > \gamma$, *reset* $\hat{\gamma}_t$ *to* γ.
3. *Use the* **same** *sample* $\mathbf{X}_1, \ldots, \mathbf{X}_N$ *to solve the stochastic program* (6), *with* $\mathbf{w} = \hat{\mathbf{v}}_{t-1}$. *Denote the solution by* $\hat{\mathbf{v}}_t$.
4. *If* $\hat{\gamma}_t < \gamma$, *set* $t = t + 1$ *and reiterate from Step 2; otherwise, proceed with Step 5.*
5. *Let* T *be the final iteration counter. Generate* $\mathbf{X}_1, \ldots, \mathbf{X}_{N_1} \sim_{iid} f(\cdot; \hat{\mathbf{v}}_T)$ *and estimate* ℓ *via importance sampling, as in* (3).

Apart from specifying the family of sampling pdfs, the sample sizes N and N_1, and the rarity parameter ϱ (typically between 0.01 and 0.1), the algorithm is completely self-tuning. The sample size N for determining a good reference parameter can usually be chosen much smaller than the sample size N_1 for the final importance sampling estimation, say $N = 1000$ versus $N_1 = 100{,}000$. Under certain technical conditions the deterministic version of Algorithm 1 is guaranteed to terminate (reach level γ) provided that ϱ is chosen small enough; see Sect. 3.5 of Rubinstein and Kroese (2004).



Cross-Entropy Method, Fig. 1 A stochastic activity network

Example: Rare-Event Probability Estimation. A stochastic activity network is a frequently used tool in project management to schedule concurrent activities. Each arc corresponds to an activity and is weighted by the duration of that activity. The maximal project duration corresponds to the length of the longest path in the graph. Figure 1 shows a stochastic activity network with eight activities. Suppose the durations of the activities are independent exponential random variables $X_1, \ldots, X_8$, each with mean 1.

Let $S(\mathbf{X})$ denote length of the longest path in the graph; that is,

$$\begin{aligned} S(\mathbf{X}) = \max\{&X_1 + X_4 + X_6 + X_8,\ X_1 + X_4 + X_7,\ X_1 \\ &+ X_5 + X_8,\ X_2 + X_8,\ X_3 + X_6 + X_8,\ X_3 + X_7\} \end{aligned}$$

Suppose the objective is to estimate the rare-event probability $\mathbb{P}(S(\mathbf{X}) \geq 20)$ using importance sampling where the random vector $\mathbf{X} = (X_1, \ldots, X_8)$ has independent exponentially distributed components with mean vector $\mathbf{v} = (v_1, \ldots, v_8)$. Note that the nominal pdf is obtained by setting $v_i = 1$ for all i. At the t-th iteration of the multilevel CE Algorithm 1, the solution to (6) with $H(\mathbf{X}) = \mathrm{I}_{\{S(\mathbf{X}) \geq \hat{\gamma}_t\}}$, using (8), is given by

$$\hat{v}_{t,i} = \frac{\sum_{k=1}^{N} \mathrm{I}_{\{S(\mathrm{X}_k) \geq \hat{\gamma}_t\}} W_k X_{ki}}{\sum_{k=1}^{N} \mathrm{I}_{\{S(\mathrm{X}_k) \geq \hat{\gamma}_t\}} W_k}, \tag{9}$$

where $\mathbf{X}_1, \ldots, \mathbf{X}_N \sim_{\mathrm{iid}} f(\cdot; \hat{\mathbf{v}}_{t-1})$, $W_k = f(\mathbf{X}_k; \mathbf{u}) / f(\mathbf{X}_k; \hat{\mathbf{v}}_{t-1})$, and X_{ki} is the i-th element of $\mathbf{X}_k$.

Cross-Entropy Method, Table 1 Convergence of the sequence $\{(\hat{\gamma}_t, \hat{\mathbf{v}}_t)\}$

t	$\hat{\gamma}_t$	$\hat{v}_t$							
0	–	1	1	1	1	1	1	1	1
1	7.32	1.93	1.12	1.39	1.83	1.32	1.81	1.37	1.96
2	12.01	3.33	1.09	1.58	2.98	1.50	2.95	1.58	3.32
3	20	5.03	1.00	1.88	4.63	1.51	4.73	1.47	5.14

Table 1 lists the successive estimates for the optimal importance sampling parameters obtained from the multilevel CE algorithm, using $N = 10^5$ and $\varrho = 0.1$.

The last step in Algorithm 1 gives an estimate of $4.15 \cdot 10^{-6}$ with an estimated relative error of 1%, using a sample size of $N_1 = 10^6$. A typical crude Monte Carlo estimate (i.e., taking $\mathbf{v} = \mathbf{u} = (1,1,\ldots,1)$) using the same sample size is $3 \cdot 10^{-6}$, with an estimated relative error of 60%, and is therefore of little use.

For large-size activity networks the accurate estimation of the optimal parameters via (9) runs into problems due to the degeneracy behavior of the likelihood ratio; cf. Rubinstein and Kroese (2007, 133). For such systems it is recommended to estimate the optimal CE parameters by drawing samples directly from g^*, for example, via Markov chain Monte Carlo; see Chan (2010).

Optimization

Let $\mathcal{X}$ be an arbitrary set of states and let S be a real-valued performance function on $\mathcal{X}$. Suppose the goal is to find the maximum of S over $\mathcal{X}$, and the corresponding maximizer $\mathbf{x}^*$ (assuming, for simplicity, that there is only one). Denote the maximum by γ^*, so that

$$S(\mathbf{x}^*) = \gamma^* = \max_{\mathbf{x} \in \mathcal{X}} S(\mathrm{X}). \tag{10}$$

Associate with the above problem the estimation of the probability $\ell = \mathbb{P}(S(\mathbf{X}) \geq \gamma)$, where $\mathbf{X}$ has some probability density $f(\mathbf{x}; \mathbf{u})$ on $\mathcal{X}$ (e.g., corresponding to the uniform distribution on $\mathcal{X}$) and γ is some level. Thus, for optimization problems, randomness is purposely introduced in order to make the model stochastic, as in the estimation setting. If γ is chosen close to the unknown γ^*, then ℓ is typically a rare-event probability, and the CE approach of section "Estimation" can be used to find an importance sampling distribution close to the theoretically optimal importance sampling density, which concentrates all its mass on point $\mathbf{x}^*$. Sampling from such a distribution thus produces optimal or near-optimal states. Note that the final level $\gamma = \gamma^*$ is generally not known in advance, in contrast to the rare-event simulation setting. The CE method for optimization produces a sequence of levels $\{\hat{\gamma}_t\}$ and reference parameters $\{\hat{\mathbf{v}}_t\}$ such that the former tends to the optimal γ^* and the latter to the optimal reference vector $\mathbf{v}^*$ corresponding to the point mass at $\mathbf{x}^*$; see, for example, Rubinstein and Kroese (2007) p. 251.

Algorithm 2 (CE Algorithm for Optimization).

1. *Choose an initial parameter vector* $\hat{\mathbf{v}}_0$. *Let* $N^{\mathrm{e}} = \lceil \varrho N \rceil$. *Set* $t = 1$ (*level counter*).
2. *Generate* $\mathbf{X}_1, \ldots, \mathbf{X}_N \sim_{\mathrm{iid}} f(\cdot; \hat{\mathbf{v}}_{t-1})$. *Calculate the performances* $S(\mathbf{X}_i)$ *for all* i, *and order them from smallest to largest:* $S_{(1)} \leq \ldots \leq S_{(N)}$. *Let* $\hat{\gamma}_t$ *be the sample* $(1 - \varrho)$*-quantile of performances; that is,* $\hat{\gamma}_t = S_{(N-N^{\mathrm{e}}+1)}$.
3. *Use the* **same** *sample* $\mathbf{X}_1, \ldots, \mathbf{X}_N$ *and solve the stochastic program*

$$\max_{\mathbf{v}} \frac{1}{N} \sum_{k=1}^{N} I_{\{S(X_k) \geq \hat{\gamma}_t\}} \ln f(\mathbf{X}_k; \mathbf{v}). \tag{11}$$

Denote the solution by $\hat{\mathbf{v}}_t$.

4. *If some stopping criterion is met, stop; otherwise, set* $t = t + 1$, *and return to Step 2.*

To run the algorithm, one needs to provide the class of sampling pdfs, the initial vector $\hat{\mathbf{v}}_0$, the sample size N, the rarity parameter ϱ, and the stopping criterion. Any CE algorithm for optimization involves thus the following two main iterative phases:

1. *Generate* a random sample of objects in the search space $\mathcal{X}$ (trajectories, vectors, etc.) according to a specified probability distribution.
2. *Update* the parameters of that distribution, based on the N^{e} best performing samples (the so-called elite samples), using CE minimization.

Note that Step 5 of Algorithm 1 is missing in Algorithm 2. Another main difference between the two algorithms is that the likelihood ratio term $f(\mathbf{X}_k; \mathbf{u})/f(\mathbf{X}_k; \hat{\mathbf{v}}_{t-1})$ in (6) is missing in (11).

Often a smoothed updating rule is used, in which the parameter vector $\hat{\mathbf{v}}_t$ is taken as

$$\hat{\mathbf{v}}_t = \alpha\,\tilde{\mathbf{v}}_t + (1-\alpha)\hat{\mathbf{v}}_{t-1}, \tag{12}$$

where $\tilde{\mathbf{v}}_t$ is the solution to (11) and $0 \le \alpha \le 1$ is a smoothing parameter. Many other modifications can be found in Kroese et al. (2006), Rubinstein and Kroese (2004, 2007). When there are two or more optimal solutions, the CE algorithm typically "fluctuates" between the solutions before focusing on one of the solutions. The effect that smoothing has on convergence is discussed in detail in Costa et al. (2007). In particular, it is shown that with appropriate smoothing the CE method converges and finds the optimal solution with probability arbitrarily close to 1. Necessary conditions and sufficient conditions under which the optimal solution is generated eventually with probability 1 are also given. Other convergence results, including a proof of convergence along the lines of the convergence proof for simulated annealing can be found in Margolin (2005). The CE method is also effective for solving noisy optimization problems, for example, when the objective function value is obtained via simulation. Typical examples may be found in Alon et al. (2005) and Cohen et al. (2007).

Combinatorial Optimization

When the state space $\mathcal{X}$ is finite, the optimization problem (10) is often referred to as a discrete or combinatorial optimization problem. For example, $\mathcal{X}$ could be the space of combinatorial objects such as binary vectors, trees, paths through graphs, permutations, etc. To apply the CE method, one needs to first specify a convenient parameterized random mechanism to generate objects $\mathbf{X}$ in $\mathcal{X}$. An important example is where $\mathbf{X} = (X_1, \ldots, X_n)$ has independent components such that $X_i = j$ with probability p_{ij}, $i = 1, \ldots, n$, $j = 1, \ldots, m$. In that case, the CE updating rule (see de Boer et al. 2005, 56) at the t-th iteration is

$$\hat{p}_{t,ij} = \frac{\sum_{k=1}^{N} \mathbf{I}_{\{S(\mathbf{X}_k) \ge \hat{\gamma}t\}}\, \mathbf{I}_{\{X_{ki}=j\}}}{\sum_{k=1}^{N} \mathbf{I}_{\{S(\mathbf{X}_k) \ge \hat{\gamma}t\}}}, \quad i = 1, \ldots, n, \quad j = 1, \ldots, m, \tag{13}$$

where $\mathbf{X}_1, \ldots, \mathbf{X}_N$ are independent copies of $\mathbf{X} \sim \{\hat{p}_{t-1,ij}\}$ and X_{ki} is the *i-th* element of $\mathbf{X}_k$. Thus, the updated probability $\hat{p}_{t,ij}$ is simply the number of elite samples for which the i-th component is equal to j, divided by the total number of elite samples.

A possible stopping rule for combinatorial optimization problems is to stop when the overall best objective value does not change over a number of iterations. Alternatively, one could stop when the sampling distribution has "degenerated" enough. For example, when in (13) the $\{\hat{p}_{t,ij}\}$ differ less than some small $\varepsilon > 0$ from the $\{\hat{p}_{t-1,ij}\}$.

Example: Max-Cut Problem. The max-cut problem in a graph can be formulated as follows. Given a weighted graph $G(V, E)$ with node set $V = \{1, \ldots, n\}$ and edge set E, partition the nodes of the graph into two subsets V_1 and V_2 such that the sum of the (nonnegative) weights of the edges going from one subset to the other is maximized. Let $C = (C(i, j))$ be the matrix of weights. The objective is to maximize

$$\sum_{(i,j)\in V_1 \times V_2} (C(i,j) + C(j,i)) \tag{14}$$

over all cuts $\{V_1, V_2\}$. Such a cut can be conveniently represented by a binary cut vector $\mathbf{x} = (1, x_2, \ldots, x_n)$, where $x_i = 1$ indicates that $i \in V_1$. Let $\mathcal{X}$ be the set of cut vectors and let $S(\mathbf{x})$ be the value of the cut represented by $\mathbf{x}$, as given in (14).

To maximize S via the CE method one can generate the random cut vectors by drawing each component (except the first one, which is set to 1) independently from a Bernoulli distribution, that is, $\mathbf{X} = (1, X_2, \ldots, X_n) \sim \text{Ber}(\mathbf{p})$, where $\mathbf{p} = (1, p_2, \ldots, p_n)$. Given an elite sample set $\mathcal{E}$, with size N^{e}, the updating formula (13) is then:

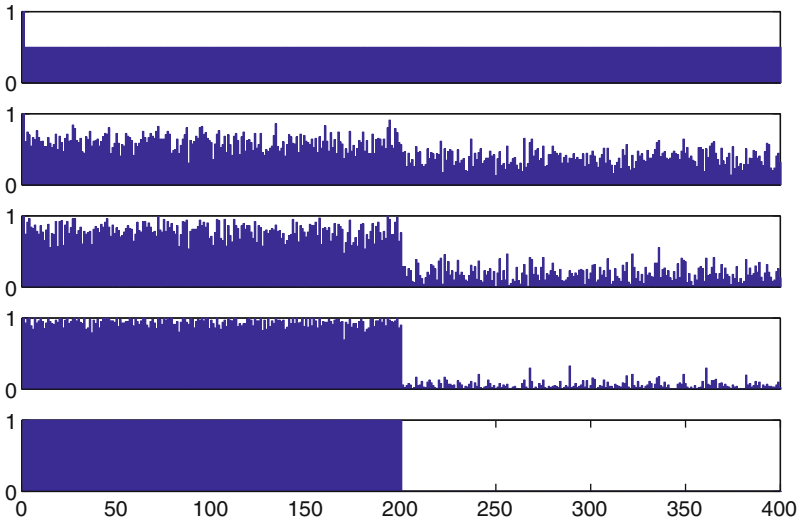
$$\hat{p}_{t,i} = \frac{\sum_{\mathbf{X}\in\mathcal{E}} X_i}{N^{\text{e}}}, \qquad i = 2, \ldots, n. \tag{15}$$

That is, the updated success probability for the *i-th* component is the mean of the i-th components of the vectors in the elite set.

Figure 2 illustrates the evolution of the Bernoulli parameters for a max-cut problem from de Boer et al. (2005) of dimension $n = 400$, for which the optimal solution is given by $\mathbf{x}^* = (1, \ldots, 1, 0, \ldots, 0)$.

Continuous Optimization

When the state space is continuous, in particular when $\mathcal{X} = \mathbb{R}^n$, the optimization problem is often referred to



Cross-Entropy Method, Fig. 2 Sequence of reference vectors for a synthetic max-cut problem with 400 nodes. Iterations 0, 5, 10, 15, and 20 are displayed

as a continuous optimization problem. The sampling distribution on $\mathbb{R}^n$ can be quite arbitrary and does not need to be related to the function that is being optimized. The generation of a random vector $\mathbf{X} = (X_1, \ldots, X_n) \in \mathbb{R}^n$ is most easily performed by drawing the coordinates independently from some 2-parameter distribution. In most applications, a normal (Gaussian) distribution is employed for each component. Thus, the sampling distribution for $\mathbf{X}$ is characterized by a vector of means $\boldsymbol{\mu}$ and a vector of standard deviations $\boldsymbol{\sigma}$. At each iteration of the CE algorithm, these parameter vectors are updated simply as the vectors of sample means and sample standard deviations of the elements in the elite set; see, for example, Kroese et al. (2006).

Algorithm 3 (CE for Continuous Optimization: Normal Updating).

1. ***Initialize***: *Choose* $\hat{\boldsymbol{\mu}}_0$ *and* $\hat{\boldsymbol{\sigma}}_0^2$. *Set* $t = 1$.
2. ***Draw***: *Generate a random sample* $\mathbf{X}_1, \ldots, \mathbf{X}_N$ *from the* $\mathrm{N}(\hat{\boldsymbol{\mu}}_{t-1}, \hat{\boldsymbol{\sigma}}_{t-1}^2)$ *distribution.*
3. ***Select***: *Let* $\mathcal{I}$ *be the indices of the* N^{e} *best performing (= elite) samples.*
   Update: *For all* $j = 1, \ldots, n$ *let*

$$\tilde{\mu}_{t,j} = \sum_{i \in \mathcal{I}} X_{ij} / N^{\mathrm{e}} \tag{16}$$

and

$$\tilde{\sigma}_{t,j}^2 = \sum_{i \in \mathcal{I}} (X_{ij} - \tilde{\mu}_{t,j})^2 / N^{\mathrm{e}}. \tag{17}$$

4. ***Smooth***:

$$\hat{\boldsymbol{\mu}}_t = \alpha \tilde{\boldsymbol{\mu}}_t + (1-\alpha)\hat{\boldsymbol{\mu}}_{t-1}, \quad \hat{\boldsymbol{\sigma}}_t = \alpha \tilde{\boldsymbol{\sigma}}_t + (1-\alpha)\hat{\boldsymbol{\sigma}}_{t-1} \tag{18}$$

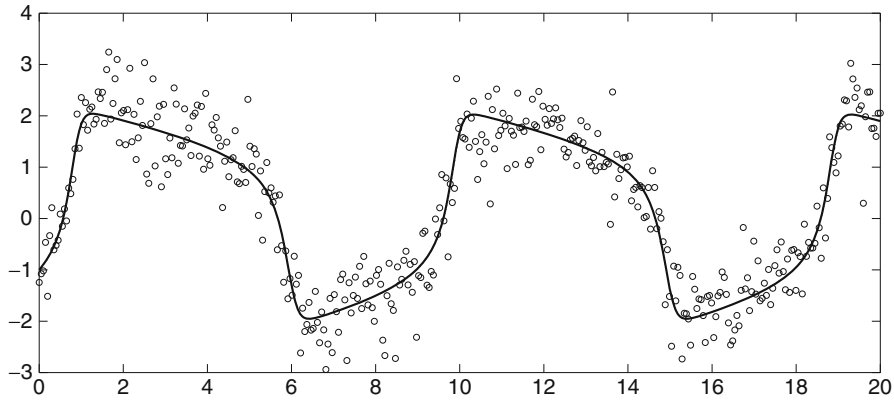
5. *If* $\max_j \{\hat{\sigma}_{t,j}\} < \varepsilon$ stop *and return* $\boldsymbol{\mu}_t$ *as an approximate solution. Otherwise, increase* t *by* 1 *and return to Step 2.*

For constrained continuous optimization problems, where the samples are restricted to a subset $\mathcal{X} \subset \mathbb{R}^n$, it is often possible to replace the normal sampling with sampling from a truncated normal distribution while retaining the updating formulas (16–17). An alternative is to use a beta distribution. Instead of returning $\hat{\boldsymbol{\mu}}_t$ as the final solution, one often returns the overall best solution generated by the algorithm.

Smoothing, as in Step 4, is often crucial to prevent premature shrinking of the sampling distribution. Instead of using a single smoothing factor, it is often useful to use separate smoothing factors for $\hat{\boldsymbol{\mu}}_t$ and $\hat{\boldsymbol{\sigma}}_t$. An alternative is to use dynamic smoothing for $\hat{\boldsymbol{\sigma}}_t$:

$$\alpha_t = \beta - \beta\left(1 - \frac{1}{t}\right)^q, \tag{19}$$

where q is an integer (typically between 5 and 10) and β is a smoothing constant (typically between 0.8 and 0.99). Another approach is to inject extra variance into the sampling distribution, for example, by increasing the components of $\boldsymbol{\sigma}$, once the distribution has degenerated; see Botev and Kroese (2004). Finally, significant speed up can be achieved by



Cross-Entropy Method, Fig. 3 Simulated data for the FitzHugh–Nagumo model and a fitted curve obtained via the CE method

using a parallel implementation of CE; see, for example, Evans et al. (2007).

Example: Parameter Estimation for Differential Equations. Consider the FitzHugh–Nagumo differential equations:

$$\frac{\mathrm{d}V_t}{\mathrm{d}t} = c\left(V_t - \frac{V_t^3}{3} + R_t\right),$$
$$\frac{\mathrm{d}R_t}{\mathrm{d}t} = -\frac{1}{c}(V_t - a + bR_t), \qquad (20)$$

which model the behavior of certain types of neurons (Nagumo et al. 1962). Ramsay et al. (2007) consider estimating the parameters a, b, and c from noisy observations of (V_t) by using a generalized smoothing approach. The simulated data in Fig. 3 correspond to the values of V_t obtained from (20) at times 0, 0.05, . . . , 20.0, adding Gaussian noise with standard deviation 0.5. The true parameter values are $a = 0.2$, $b = 0.2$, and $c = 3$. The initial conditions are $V_0 = -1$ and $R_0 = 1$.

Estimation of the parameters via the CE method can be established by minimizing the least-squares performance

$$S(\mathbf{x}) = \sum_{i=0}^{400} (y_i - V_{0.05i}(\mathbf{x}))^2,$$

where $\{y_i\}$ are the simulated data, $\mathbf{x} = (a, b, c, V_0, R_0)$, and $V_t(\mathbf{x})$ is the solution to (20) for parameter vector $\mathbf{x}$. Algorithm 3 was implemented with $\hat{\boldsymbol{\mu}}_0 = (0, 0, 5, 0, 0)$, $\hat{\boldsymbol{\sigma}}_0 = (1,1,1,1,1)$, $N = 100$, $N^e = 10$, and $\varepsilon = 0.001$. Constant smoothing parameters $\alpha_1 = 0.9$ and $\alpha_2 = 0.5$ were used for the $\{\hat{\boldsymbol{\mu}}_t\}$ and the $\{\hat{\boldsymbol{\sigma}}_t\}$, respectively. The following solution was found (note that the initial condition was assumed to be unknown): $\hat{a} = 0.19$, $\hat{b} = 0.21$, $\hat{c} = 3.00$, $\hat{V}_0 = -1.02$, and $\hat{R}_0 = 1.02$. The smooth curve in Fig. 3 gives the corresponding estimated curve, which is practically indistinguishable from the true one.

See

- Monte Carlo Methods
- Monte Carlo Simulation
- Rare Event Simulation
- Simulation of Stochastic Discrete-Event Systems
- Simulation Optimization

References

Alon, G., Kroese, D. P., Raviv, T., & Rubinstein, R. Y. (2005). Application of the cross-entropy method to the buffer allocation problem in a simulation-based environment. *Annals of Operations Research, 134*(1), 137–151.

Botev, Z. I., & Kroese, D. P. (2004). Global likelihood optimization via the cross-entropy method with an application to mixture models. *Proceedings of the 36th Winter Simulation Conference*, Washington, DC, pp. 529–535.

Chan, J. C. C. (2010). *Advanced Monte Carlo methods with applications in finance*. PhD thesis, University of Queensland.

Cohen, I., Golany, B., & Shtub, A. (2007). Resource allocation in stochastic, finite-capacity, multi-project systems through the cross entropy methodology. *Journal of Scheduling, 10*(1), 181–193.

Costa, A., Owen, J., & Kroese, D. P. (2007). Convergence properties of the cross-entropy method for discrete optimization. *Operations Research Letters, 35*(5), 573–580.

de Boer, P. T., Kroese, D. P., Mannor, S., & Rubinstein, R. Y. (2005). A tutorial on the cross-entropy method. *Annals of Operations Research, 134*(1), 19–67.

Evans, G. E., Keith, J. M., & Kroese. D. P. (2007). Parallel cross-entropy optimization. *Proceedings of the 2007 Winter Simulation Conference*, Washington, DC, pp. 2196–2202.
Kroese, D. P., Porotsky, S., & Rubinstein, R. Y. (2006). The cross-entropy method for continuous multi-extremal optimization. *Methodology and Computing in Applied Probability, 8*(3), 383–407.
Margolin, L. (2005). On the convergence of the cross-entropy method. *Annals of Operations Research, 134*(1), 201–214.
Nagumo, J., Arimoto, S., & Yoshizawa, S. (1962). An active pulse transmission line simulating nerve axon. *Proceedings of the IRE, 50*(10), 2061–2070.
Ramsay, J. O., Hooker, G., Campbell, D., & Cao, J. (2007). Parameter estimation for differential equations: A generalized smoothing approach. *Journal of the Royal Statistical Society, Series B, 69*(5), 741–796.
Rubinstein, R. Y. (1997). Optimization of computer simulation models with rare events. *European Journal of Operational Research, 99*(1), 89–112.
Rubinstein, R. Y. (1999). The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability, 1*(2), 127–190.
Rubinstein, R. Y. (2001). Combinatorial optimization, cross-entropy, ants and rare events. In S. Uryasev & P. M. Pardalos (Eds.), *Stochastic optimization: Algorithms and applications* (pp. 304–358). Dordrecht: Kluwer.
Rubinstein, R. Y., & Kroese, D. P. (2004). *The cross-entropy method: A unified approach to combinatorial optimization, Monte Carlo simulation and machine learning*. New York: Springer.
Rubinstein, R. Y., & Kroese, D. P. (2007). *Simulation and the Monte Carlo Method* (2nd ed.). New York: Wiley.

Crossover

A genetic-algorithm operator which exchanges corresponding genetic material from two parent chromosomes (i.e., solutions), allowing genes on different parents to be combined in their offspring.

See

► Genetic Algorithms

CS

Computer science.

See

► Computer Science and Operations Research Interfaces

Curse of Dimensionality

The situation that arises in such areas as dynamic programming, control theory, integer programming, combinatorial problems, and, in general, time-dependent problems in which the number of states and/or data storage requirements increases exponentially with small increases in the problems' parameters or dimensions; sometimes referred to as combinatorial explosion.

See

► Combinatorial Explosion
► Control Theory
► Dynamic Programming
► Integer and Combinatorial Optimization

References

Bellman, R. (1957). *Dynamic programming*. Princeton, NJ: Princeton University Press. (Dover Publications reprint 2003).

Customer Distribution

The probability distribution of the state of the process that customers observe upon arrival to a queueing system. In general, it is not the same as the distribution seen by a random outside observer; but the two distributions are the same for queueing systems with Poisson arrivals (PASTA). Since customers entering a queue must also exit, the probability distribution seen by arriving customers who are accepted is the same as that for the number of customers left behind by the departures.

See

► Outside Observer Distribution
► PASTA
► Queueing Theory

Cut

A set of arcs in a graph (network) whose removal eliminates all paths joining a node *s* (source node) to a node *t* (sink node).

See

- ▶ Graph Theory
- ▶ Max-Flow Min-Cut Theorem
- ▶ Maximum-Flow Network Problem

Cutset

A minimal set of edges whose removal disconnects a graph.

See

- ▶ Cut
- ▶ Graph Theory

Cutting Stock Problems

Robert W. Haessler
University of Michigan, Ann Arbor, MI, USA

Introduction

Solid materials such as aluminum, steel, glass, wood, leather, paper and plastic film are generally produced in larger sizes than required by the customers for these materials. As a result, the producers or primary converters must determine how to cut the production units of these materials to obtain the sizes required by their customers. This is known as a cutting stock problem. It can occur in one, two or three dimensions depending on the material. The production units may be identical, may consist of a few different sizes, or may be unique. They may be of consistent quality throughout or may contain defects. The production units may be regular (rectangular) or irregular. The ordered sizes may be regular or irregular. They may all have the same quality requirements or some may have different requirements. They may have identical or different timing requirements which impact inventory. The first

Some examples follow:

- cutting rolls of paper from production reels of the same diameter.
- cutting rectangular pieces of glass from rectangular production sheets.
- cutting irregular pieces of steel from rectangular plates.
- cutting rectangular pieces of leather from irregular hides.
- cutting dimensional lumber from logs of various size.

There are two other classes of problems which are closely related to the above cutting problems. The first is the layout problem. An example of this would be the problem of determining the smallest rectangle which will contain a given set of smaller rectangles without overlap. Solving this problem is essentially the same as being able to generate a cutting pattern in the discussion of cutting stock problems which follows. The second type of problem, which in many cases can be solved by the same techniques as cutting stock problems, is the (bin) packing problem. A one-dimensional example of this would be to determine the minimum number of containers required to ship a set of discrete items where weight and not floor space or volume is the determinant of what can be placed in the container. If floor space or volume is the key determinant, then the problem is equivalent to a two or three-dimensional cutting stock problem in which guillotine cuts are not required. Even though the following discussions focuses on cutting stock problems, it is also applicable to solving both packing and layout problems.

Although cutting stock problems are relatively easy to formulate, many of them especially those with irregular shapes, are difficult to solve; there are no efficient solution procedures available. The major difficulty has to do with the generation of feasible low trim loss cutting pattern. As will be discuseed below, this ranges from being simple in one-dimension to complex in two-dimensions, even with regular shapes.

The first known formulation of a cutting stock problem was given in 1939 by the Russian economist Kantorovich (1960). The first and most significant advance in solving cutting problems was the seminal

work of Gilmore and Gomory (1961, 1963) in which they described their delayed pattern generation technique for solving the one-dimensional trim loss minimization problem using linear programming. Since that time, there has been an explosion of interest in this application area. Sweeney and Paternoster (1992) have identified more than 500 papers which deal with cutting stock and related problems and applications. The primary reasons for this activity are that cutting stock problems occur in a wide variety of industries, there is a large economic incentive to find more effective solution procedures, and it is easy to compare alternative solution procedures and to identify the potential benefits of using a proposed procedure.

Cutting stock problems are introduced with a discussion of the one-dimensional problem and the techniques available for solving it. The article concludes with an extension to the regular two dimensional problem.

One-Dimensional Problems

An example of a one-dimensional cutting stock problem is the trim loss minimization problem that occurs in the paper industry. In this problem, known quantities of rolls of various widths and the same diameter are to be slit from stock rolls of some standard width and diameter. The objective is to identify slitting patterns and their associated usage levels that satisfy the requirements for ordered rolls at the least possible total cost for scrap and other controllable factors. The basic cutting pattern feasibility restriction in this problem is that the sum of the roll widths slit from each stock roll must not exceed the usable width of the stock roll.

Let R_i be the nominal order requirements for rolls of width W_i, $i = 1, \ldots, n$, to be cut from stock rolls of usable width UW. We have RL_i and RU_i as lower and upper bounds on the order requirement, for customer order i, reflecting the general industry practice of allowing overruns or underruns within specified limits. Depending on the situation, R_i may be equal to RL_i and/or RU_i. All orders are for rolls of the same diameter. This problem can be formulated as follows, with X_j as the number of stock rolls to be slit using pattern j and T_j as the trim loss incurred by pattern j:

$$\text{minimize} \sum_j T_j X_j \tag{1}$$

$$\text{s.t } RL_i \leq \sum_j A_{ij} X_j \leq RU_i \text{ for all } i \tag{2}$$

$$T_j = UW - \sum_i A_{ij} W_i \tag{3}$$

$$X_j \geq 0, \quad \text{integer.} \tag{4}$$

where A_{ij} is the number of rolls of width W_i to be slit from each stock roll that is processed using pattern j. In order for the elements A_{ij}, $i = 1, \ldots, n$, to constitute a feasible cutting pattern, the following restrictions must be satisfied:

$$\sum_i A_{ij} W_i \leq UW, \tag{5}$$

$$A_{ij} \geq 0, \quad \text{integer} \tag{6}$$

Note that the objective in this example is simply to minimize trim loss. In most industrial applications, it is necessary to consider other factors in addition to trim loss. For example, there may be a cost associated with pattern changes and, therefore, controlling the number of patterns used to satisfy the order requirements would be an important consideration.

Because optimal solutions to integer cutting stock problems can be found only for values of n smaller than typically found in practice, heuristic procedures represent the only feasible approach to solving this type of problem. Two types of heuristic procedures have been widely used to solve one-dimensional cutting stock problems. One approach uses the solution to a linear programming (LP) relaxation of the integer problem above as its starting point. The LP solution is then modified in some way to provide a integer solution to the problem. The second approach is to generate cutting patterns sequentially to satisfy some portion of the remaining requirements. This sequential heuristic procedure (SHP) terminates when all order requirements are satisfied.

Linear Programming Solutions

Almost all LP-based procedures for solving cutting stock problems can be traced back to Gilmore and Gomory (1961, 1963). They described how the next pattern to enter the LP basis could be found by solving

an associated knapsack problem. This made it possible to solve the trim loss minimization problem by linear programming without first enumerating every feasible slitting pattern. This is extremely important because a large number of feasible patterns may exist when narrow widths are to be slit from a wide stock roll. Pierce (1964) showed that in such situations the number of slitting patterns can easily run into the millions. Because only a small fraction of all possible slitting patterns need to be considered in finding the minimum trim loss solution, the delayed pattern generation technique developed by Gilmore and Gomory made it possible to solve trim loss minimization problems in much less time than would be required if all the slitting patterns were input to a general-purpose linear programming algorithm.

A common LP relaxation of the integer programming problem given in (1)–(3) can be stated as follows:

$$\text{minimize} \sum_j X_j \tag{7}$$

$$\text{s.t} \sum_j A_{ij} X_j \geq RU_i \text{ for all } i, \tag{8}$$

$$X_j \geq 0, \quad \text{integer}. \tag{9}$$

Let U_i be the dual variable associated with constraint i. Then the dual of this problem can be stated as:

$$\text{minimize} \sum_i R_i U_i \tag{10}$$

$$\text{s.t.} \sum_i A_{ij} U_i \leq 1 \tag{11}$$

$$U_i \geq 0. \tag{12}$$

The dual constraints in (11) provide the means for determining if the optimal LP solution has been obtained or if there exists a pattern which will improve the LP solution because the dual problem is still infeasible.

The next pattern $\boldsymbol{A} = (A_1, \ldots, A_n)$ to enter the basis, if one exists, can be found by solving the following knapsack problem:

$$Z = \text{maximize} \sum_i U_i A_i \tag{13}$$

$$\text{s.t.} \sum_i W_i A_i \leq UW \tag{14}$$

$$A_i \geq 0, \quad \text{integer} \tag{15}$$

If $Z \leq 1$, the current solution is optimal. If $Z > 1$, then **A** can be used to improve the LP solution.

Once found, the LP solution can be modified in a number of ways to obtain integer values for the X_j which satisfy the order requirements. One common approach is to round the LP solution down to integer values, then increase the values of X_j by unit amounts for any patterns whose usage can be increased without exceeding RU_i. Finally, new patterns can be generated for any rolls still needed using the sequential heuristic described in the next section.

Sequential Heuristic Procedures (SHP)

With an SHP, a solution is constructed one pattern at a time until all the order requirements are satisfied. The first documented SHP capable of finding better solutions than those found manually by schedulers was described by Haessler (1971). The key to success with this type of procedure is to make intelligent choices as to the patterns which are selected early in the SHP. The patterns selected initially should have low trim loss, high usage and leave a set of requirements for future patterns which will combine well without excessive side trim.

The following procedure is capable of making effective pattern choices in a variety of situations:

1. Compute descriptors of the order requirements yet to be scheduled. Typical descriptors would be the number of stock rolls still to be slit and the average number of ordered rolls to be cut from each stock roll.
2. Set goals for the next pattern to be entered into the solution. Goals should be established for trim loss, pattern usage, and number of ordered rolls in the pattern.
3. Search exhaustively for a pattern that meets those goals.
4. If a pattern is found, add this pattern to the solution at the maximum possible level without exceeding R_i, for all i. Reduce the order requirements and return to 1.
5. If no pattern is found, reduce the goal for the usage level of the next pattern and return to 3.

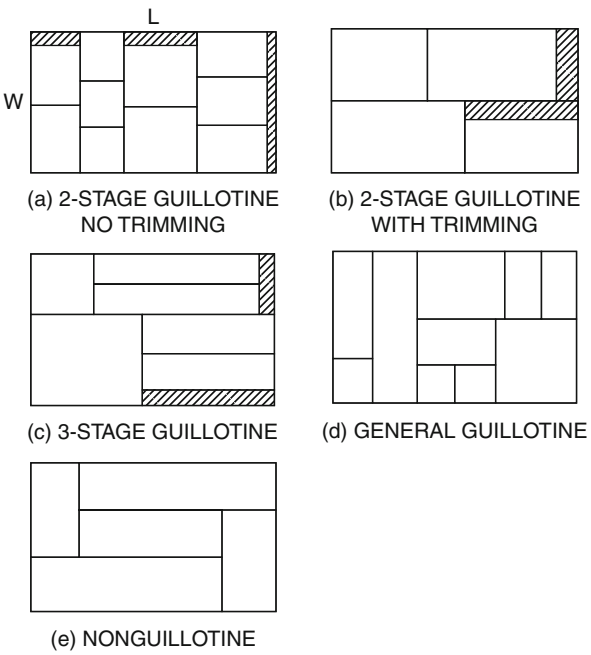
The pattern usage goal provides an upper bound on the number of times a size can appear in a pattern. For example, if some ordered width has an unmet requirement of 10 rolls and the pattern usage goal is 4, that width may not appear more than twice in a pattern. If after exhaustive search no pattern satisfies the goals set, then at least one goal, most commonly pattern usage, must be relaxed. This increases the number of patterns to be considered. If the pattern usage goal is changed to 3 in the above example, then the width can appear in the pattern three times. Termination can be guaranteed by selecting the pattern with the lowest trim loss at the usage level of one.

The primary advantage of this SHP is its ability to control factors other than trim loss and to eliminate rounding problems by working only with integer values. For example, if there is a cost associated with a pattern change, a sequential heuristic procedure which searches for high usage patterns may give a solution which has less than one-half the number of patterns required by an LP solution to the same problem. The major disadvantage of an SHP is that it may generate a solution which has greatly increased trim loss because of what might be called ending conditions. For example, if care is not taken as each pattern is accepted and the requirements reduced, the widths remaining at some point in the process may not have an acceptable trim loss solution. Such would be the case if only 34-inch rolls are left to be slit from 100-inch stock rolls.

Rectangular Two-Dimensional Problems

The formulation of a higher dimensional cutting stock problem is exactly the same as that of the one-dimensional problem given in (1)–(4). The only added complexity comes in trying to define and generate feasible cutting patterns. The simplest two-dimensional case is one in which both the stock and ordered sizes are rectangular. Most of the important issues regarding cutting patterns for rectangular two-dimensional problems can be seen in the examples shown in Fig. 1.

One important issue not covered in Fig. 1 is a limit on the number of times an ordered size can appear in a pattern. This generally is a function of the maximum quantity of pieces, RU_i, required for order i. If R_i is small, it is just as important for the two-dimensional case as the one dimensional case that the number of times size i appears in a pattern should be limited. This becomes less important as R_i becomes larger and as the difference between RU_i and RL_i becomes larger.



Cutting Stock Problems, Fig. 1 Sample cutting patterns

The cutting pattern shown in Fig. 1(a) is an example of two-stage guillotine cuts. The first cut can be in either the horizontal or vertical direction. A second cut perpendicular to the first, yields a finished piece. Figure 1(b) is similar except a third cut can be made to trim the pieces down to the correct dimension. Figure 1(c) shows the situation in which the third cut can create 2 ordered pieces.

For simple staged cutting such as shown in Fig. 1 (a, b, c), Gilmore and Gomory (1965) showed how cutting patterns can be generated by solving two one-dimensional knapsack problems. To simplify the discussion, assume that the orientation of each ordered piece is fixed relative to stock piece and the first guillotine cut on the stock pieces must be along the length (larger dimension) of the stock piece. For each ordered width W_k, find the contents of a strip of width W_k and length L which gives the maximum contribution to dual infeasibility:

$$Z_k = \text{maximize} \sum_{i \varepsilon I_k} U_i A_{ik} \qquad (16)$$

$$\text{s.t.} \sum_{i \varepsilon I_k} L_i A_{ik} \leq L \tag{17}$$

$$A_{ik} \geq 0, \text{ integer.} \tag{18}$$

$$I_k = \{i | W_i \leq W_k\}. \tag{19}$$

Next find the combination of strips which solve the problem

$$Z = \text{maximize} \sum_k Z_k A_k \tag{20}$$

$$\sum_k W_k A_k \leq W \tag{21}$$

$$A_k \geq 0, \quad \text{integer.} \tag{22}$$

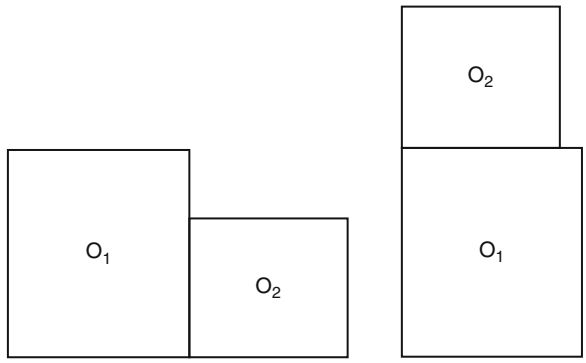
Any pattern for which Z is greater than one will yield an improvement in the LP solution.

The major difficulty with this approach is the inability to limit the number of times an ordered size appears in a pattern. It is easy to restrict the number of times a size appears in a strip and to restrict the number of strips in a pattern. The problem is that small ordered sizes with small quantities may end up as filler in a large number of different strips. This makes the two-stage approach to developing patterns ineffective when the number of times a size appears in a pattern must be limited.

Wang (1983) developed an alternative approach to generating general guillotine cutting patterns which limits the number of times a size appears in a pattern. She combined rectangles in a horizontal and vertical build process as shown in Fig. 2 where O_i is an ordered rectangle of width W_i and length L_i.

She used an acceptable value for trim loss, B, rather than the shadow price of the ordered sizes to drive her procedure which is as follows:

- Step 1
  - (a) Choose a value for B the maximum acceptable trim waste.
  - (b) Define $L^{(O)} = F^{(O)} = \{O_1, O_2, \ldots, O_n\}$, and set $K = 1$.
- Step 2
  - (a) Compute $F^{(K)}$ which is the set of all rectangles T satisfying (i) T is formed by a horizontal or vertical build of two rectangles from $L^{(K-1)}$, (ii) the amount of trim waste in T does not exceed B, and (iii) those rectangles O_i, appearing in T do not violate the constraints on the number of times a size can appear in a pattern.



(a) Horizontal build of O_1 and O_2 (b) Vertical build of O_1 and O_2

Cutting Stock Problems, Fig. 2 Guillotine cutting patterns

  - (b) Set $L^{(K)} = L^{(K-1)} UF^{(K)}$. Remove any equivalent (same component rectangles) rectangle patterns from $L^{(K)}$.
- Step 3. If $F^{(K)}$ is non-empty, set $K = K + 1$ and go to Step 2; otherwise, set $M = K - 1$, and choose the rectangle in $L^{(M)}$ which has the smallest total trim waste when placed in the stock rectangle.

Concluding Remarks

It is clear that moving from one to two-dimensions causes significant difficulty in the pattern generating process. This is all the more alarming in light of the fact that only rectangular shapes were considered.

This clearly suggests that there is much more research needed on procedures for solving two-dimensional cutting stock problems. An alternative worth considering, especially in those cases where there are many different ordered sizes with small order quantities, might be to first select a subset of orders to consider by solving a one-dimensional knapsack problem as in (13)–(15) based on area and then see if the resulting solution can be put together into a feasible two-dimensional pattern. Wang's algorithm seems to be ideal for this purpose inasmuch as the trim loss in the pattern would be known.

A candidate set of items to be included in the next pattern could be found by solving the following problem:

$$Z = \text{maximize} \sum_i U_i A_i \tag{23}$$

$$\sum_i AR_i A_i \leq UAR \text{ for all } i \quad (24)$$

$$A_i \leq b_i \quad (25)$$

$$A_i \geq 0, \quad \text{integer.} \quad (26)$$

where AR_i is the area of ordered rectangle i, UAR is the usable area of the stock rectangle, and b_i is the upper limit on the number of times order i can be included in the pattern.

The candidate pattern $(A_1, \ldots, A_n)$ could then be tested for feasibility using Wang's procedure. If the AR_i are small, the chances are that there will be little trim loss in the candidate patterns generated. This may require that UAR be reduced to force some trim loss to make it more likely that feasible patterns are found.

See

- ▶ Bin-Packing
- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming

References

Gilmore, P. C., & Gomory, R. E. (1961). A linear programming approach to the cutting stock problem. *Operations Research, 9*, 848–859.

Gilmore, P. C., & Gomory, R. E. (1963). A linear programming approach to the cutting stock problem, part II. *Operations Research, 11*, 863–888.

Gilmore, P. C., & Gomory, R. E. (1965). Multistage cutting stock problems of two and more dimensions. *Operations Research, 13*, 94–120.

Gilmore, P. C., & Gomory, R. E. (1966). The theory and computation of knapsack functions. *Operations Research, 14*, 1045–1074.

Haessler, R. W. (1971). A heuristic programming solution to a nonlinear cutting stock problem. *Management Science, 17*, 793–802.

Haessler, R. W., & Sweeney, P. E. (1991). Cutting stock problems and solution procedures. *European Journal of Operational Research, 54*, 141–150.

Kantorovich, L. V. (1960). Mathematical methods of organizing and planning production. *Management Science, 6*, 366–422. reprinted in.

Paull, A. E. (1956). Linear programming: A Key to optimum newsprint production. *Paper Magazine of Canada, 57*, 85–90.

Pierce, J. F. (1964). *Some large scale production problems in the paper industry*. Englewood Cliffs, NJ: Prentice-Hall Inc.

Sweeney, P. E., & Paternoster, E. R. (1992). Cutting and packing problems: A categorized, application-oriented research bibliography. *Journal of the Operational Research Society, 43*, 691–706.

Wang, P. Y. (1983). Two algorithms for constrained two-dimensional cutting stock problems. *Operations Research, 31*, 573–586.

C

CV

- ▶ Coefficient of Variation

Cybernetics and Complex Adaptive Systems

Andrew P. Sage
George Mason University, Fairfax, VA, USA

Introduction

Cybernetics is a term that is occasionally used in the literature of such areas as systems engineering and OR/MS to denote the study of control and communication in, and, in particular between humans, machines, organizations, and society. The word cybernetics comes from the Greek word *Kybernetes*, which means controller, or governor, or steersman. The first modern use of the term was due to Professor Norbert Wiener, an MIT professor of mathematics, who made many early and seminal contributions to mathematical system theory (Wiener 1949). The first book formally on this subject was titled *Cybernetics* and published in 1948 (Wiener 1948). In this book, Wiener defined the term as "control and communication in the animal and the machine." This emphasized the concept of feedback control as a construct presumably of value in the study of neural and physiological relations in the biological and physical sciences. In the historical evolution of cybernetics, major concern was initially devoted to the study of feedback control and servomechanisms, studies which later evolved into the area of control systems or control engineering (Singh 1990). Cybernetic concerns also have involved analog and digital computer development, especially computer

efforts that were presumed to be models of the human brain and the combination of computer and control systems for purposes of automation and remote control (Ashby 1952, 1956; George 1971; Lerner 1976).

There were a number of other early influences on cybernetics, including artificial intelligence (AI). The two are quite different subjects, however. Artificial Intelligence is generally concerned with endowing computers with machine intelligence such that they can emulate certain forms of human behavior, generally cognitive behavior. Cybernetics is an epistemological subject that is fundamentally concerned with limits on how we know what we know. It seeks to understand systems in a variety of media — technological, biological, social, or organizational — and descriptions of these limits as a most important result. So while AI seeks to endow computers with human cognitive capabilities, a subject associated with much controversy (Dreyfus 1992), cybernetics is much more concerned with using computational capabilities to develop models of systems based on the information, feedback, and control properties of these systems. In a cybernetic system, information and knowledge are attributes of interactions that occur within the system. It was the initial presumed resemblance, at a neural or physiological level, between physical control systems and the central nervous system and human brain that concerned Wiener. He and close associates, Warren McCulloch, Arturo Rosenblueth, and Walter Pitts, were the initial seminal thinkers in this new field of cybernetics. Soon, it became clear that it was fruitless to study control independent of information flow; cybernetics thus took on an identification with the study of communications and control in humans and machines. An influence in the early notions of cybernetics was the thought that physical systems could be made to perform better by, somehow, enabling them to emulate human systems at the physiological or neural level. Thus, early efforts in what is now known as neural networks began as cybernetic studies.

Another early concept explored in cybernetics was that of homeostasis, which has come to be known as the process by which systems maintain their level of organization in the face of disturbances, often occurring over time, and generally of a very large scale (Ashby 1952). Cybernetics soon became concerned with purposive organizational systems, or viable systems, as contrasted with systems that are static over time and purpose (Beer 1979). Further, organizations operate in the face of incomplete and redundant information by establishing useful patterns of communications (Beer 1979). Thus, organizations can potentially be modeled and have been modeled as cybernetic systems (Steinbrunner 1974).

Cybernetics has often been viewed as a way of looking at systems, or as a philosophical perspective concerning inquiry, as contrasted with a very specific method. This is perhaps much more the case now than during the very early history of use of the term. An excellent collection of Norbert Wiener's original papers on cybernetics studies is contained in Volume IV of an edited anthology (Masani 1985). Fundamental to any cybernetic study is the notion of modeling, and, in particular, the interpretation of the results of a modeling effort as theories that have normative or predictive value. Today, there is little explicit or implicit agreement concerning a precise definition for cybernetics. Some users of the term cybernetics infer that the word implies a study of control systems. Some uses refer to modeling only at the neural and physiological level. Some refer to cognitive ergonomic modeling without necessary consideration of, or connection to, neuronal level elements. Other uses of the word are so general that cybernetics might seem to infer either nothing, or everything. Automation, robotics, artificial intelligence, information theory, bionics, automata theory, pattern recognition and image analysis, control theory, communications, human and behavioral factors, and other topics have all, at one time or another, been assumed to be a portion of cybernetics.

Complex adaptive systems (CAS) involve phenomena associated with interactions of many individual agents that self-organize at higher aggregate system levels. This results in emergent and adaptive properties that are not exhibited by the individual agents. These systems are cybernetic like systems that receive data and information from their environments, find regularities in the data and information, and then identify internal models that process this data and information in order to describe and forecast likely futures. These systems are

evolutionary in the sense that these internal models are subject to selection pressures based on particular environmental conditions and this results in changes to the structure and parameters associated with the internal models. These systems function best under conditions between chaos and order, sometimes referred to as at the edge of chaos (Langton 1990) or self-organized criticality (Bak and Chen 1991; Bak 1996). The emergent characteristics of a particular complex system (Holland 1996) are often equivalent to individual agents acting in a higher level complex system. Adaptation occurs when either the functional or structural properties of an agent change in such a manner as to improve survival probabilities in the environment of the agent. Often the only way to study these complex adaptive systems is through computer simulation.

Definition of Cybernetics and Complex Adaptive Systems

The notion of the physiological aspects of the human nervous system as playing a necessarily critical role in modern cybernetics has all but vanished, except in very specialized classic works. This does not suggest that interest in neural type studies has vanished as there is much interest today in neural networks and related subjects (Freeman and Skapura 1991; Zurada 1992). A much more cognitive perspective is now prevalent, at least in many systems engineering views of cybernetics. In this article, cybernetics is defined as the study of the communication and control processes associated with human-machine interaction in systems that are intended to support accomplishment of purposeful tasks. While this is not a universally accepted definition of cybernetics, it is a useful one for many systems engineering studies involving human-system interaction through communications and control (Sage 1992). Complex systems theory is a general approach to understanding the overall behavior of system comprised of many nonlinearly interacting parts. The complex systems approach tries to construct minimal underlying rule sets from which desired behaviors naturally emerge. Complex adaptive system theory also assumes that systems are composed of interacting agents that continually adapt by changing their internal rules as the environment, and their experience in that environment, evolve over time. Systems transition naturally between equilibrium points through environmental adaptation and self-organization. A complex adaptive system behaves and evolves according to three key principles: (1) order is emergent as opposed to predetermined, (2) the system's history is irreversible, and (3) the system's future is generally unpredictable. Complex adaptive systems are complex systems consisting of many nonlinearly interacting parts or agents. These agents can adapt to changing environments where each agent typically exists within a nested hierarchy of agents within agents.

The purpose of this article is to discuss cybernetics and complex adaptive systems, and the design of support systems based on these concepts for such purposes as knowledge support to humans. Especial concern is with the human-system interactions that occur in such an effort. Thus, the discussions here are particularly relevant to knowledge-based system design concerns relative to human-machine cybernetic problem solving tasks, such as fault detection, diagnosis and correction. These are very important concerns for a large number of knowledge-support systems engineering applications that require fundamentally cognitive support to humans in supervisory control tasks (Sage 1991, 1992, 1995; Sheridan 1992; Rasmussen 1986; Rasmussen et al. 1994).

The need for humans to monitor and maintain the conditions necessary for satisfactory operation of systems and to cope with poorly structured and imprecise knowledge is greater than ever. Ultimately, these primarily cognitive efforts, which involve a great variety of human problem solving activities, are often translated into physical control signals for controlling or manipulating some physical process. As a consequence of this, there are a number of human interface issues that naturally occur between the human and the machines over which the human must exercise control. Many advances in information technology result in systems that enable a significant increase in the amount of information that is available for judgment and decision-making tasks at the problem solving level. Even the highest quality information, however, will generally be associated with considerable uncertainty, imprecision, and other forms of imperfections. Above all else, there is

a major need for information to be associated with context such that it becomes knowledge useful for judgment and choice. The contemporary use of information technology has led to and is expected to continue to lead to major organizational transformations in the future (Harrington 1991; Scott Morton 1991; Davenport 1993; Drucker 1995, 1998.

Cybernetics, Complex Adaptive Systems, and Systems Management

A human-machine cybernetic system may be defined as a functional synthesis of a human system and a technological system or machine. The interaction and functional interdependence between these two elements pre-dominantly characterize human-machine systems. The introduction of communication and control concerns results in a cybernetic system. All kinds of technological systems, regardless of their degree of complexity, may be viewed as parts of a human-machine cybernetic system: industrial plants, vehicles, manipulators, prostheses, computers or management information systems. A human-machine system may, of course, be a subsystem that is incorporated within another system. For example, a decision support system may be incorporated as part of a larger enterprise management, process control, or computer-aided design system that also involves human interaction. This use of the term human-machine cybernetic system corresponds, therefore, to a specific way of looking at technological systems through the integration of technological systems and human-enterprise systems, generally through a systems management or systems engineering process.

The overall purpose of any human-machine cybernetic system is to provide a certain function, product, or service with reasonable cost under constraint conditions and disturbances. This concept involves and influences the human, the machine, and the processes through which they function as an integrated whole. The primary inputs to a human-machine cybernetic system are a set of purposeful performance objectives that are typically translated into a set of expected values of performance, costs, reliability, and safety. Also, the design must be such that an acceptable level of workload and job satisfaction is maintained. It is on the basis of these that the human is able to perform the following activities (Sage 1992):

1. Identify task requirements, such as to enable determination of the issues to be examined further and the issues to be not considered;
2. Identify a set of hypotheses or alternative courses of actions which may resolve the identified issues to be resolved;
3. Identify the probable impacts of the alternative courses of action;
4. Interpret these impacts in terms of the objectives or inputs to the task;
5. Select an alternative for implementation and implement the resulting control;
6. Monitor performance such as to enable determination of how well the integrated combination of human and system are performing.

Many researchers have described activities of this sort in a number of frameworks that include behavioral psychology, organizational management, human factors, systems engineering, operations research and management science.

Many questions can be raised concerning the use of information for judgment and choice activities, as well as activities that lead to the physical control of an automated process. Any and all of these questions can arise in different application areas. These questions relate to the control of technological systems. They concern the degree of automation with respect to flexible task allocation. They also concern the design and use of computer-generated displays. Further, they relate to all kinds of human-computer interaction concerns, as well as management tasks at different organizational levels: strategic, tactical, and operational. For example, computer-based support systems to aid human performance continue to invade more and more areas of the engineering of systems: design, operation, maintenance, and management. The importance of augmenting hardware and micro-level programming aspects of system design to architectural and software systems management considerations is great. The integrated consideration of systems engineering and systems management for software productivity is expressed by the term software systems engineering (Sage and Palmer 1990).

Human tasks in human-machine cybernetic systems can be condensed into three primary categories: (1) controlling (physiological); (2) communicating

(cognitive), and (3) problem solving (cognitive) (Johannsen et al. 1983). In addition, there exists a monitoring or feedback portion of the effort that enables learning over time. Ideally, but not always, the humans involved learn well. There needs to be metalevel learning, or learning how to learn if improvements are to truly be lasting, as contrasted with only specific task performance learning. Associated with the rendering of a single judgment and the associated control implementation, the human monitors the result of the effect of these activities. The effect of present and past monitoring is to provide an experiential base for present problem conceptualization. In the categorization above, activities 1 through 4 may be viewed as problem (finding and) solving, activity 5 involves implementation or controlling, and activity 6 involves communications or monitoring and feedback in which responses to the question "How good is the process performance?" enables improvement and learning through iteration. Of course, the notion of information flow and communication is involved in all of these activities.

These three human task categories are fairly general. Controlling should be understood in a much broader sense than in many control theory studies. Controlling in this narrower sense includes open loop vs. closed-loop and continuous vs. intermittent controlling, as well as discrete tasks such as reaching, switching and typing. It is only through these physiological aspects of controlling that outputs of the human-machine cybernetic system can be produced. Controlling, in the sense of the cognitive ergonomic concerns that support human information processing and associated judgment and choice, is also included. Although human functions on a cognitive level can and do play a role in control implementation, their major importance occurs in problem solving activities. Tasks such as fault detection, fault diagnosis, fault compensation or managing, and planning are particularly important in problem solving. Fault detection concerns the identification of a potential difficulty concerning the operation of a system. Fault diagnosis is concerned with identification of a set of hypotheses concerning the likely cause of a system malfunction, and the evaluation and selection of a most likely cause. It is primarily a cognitive activity. Fault compensation or managing is concerned with solving problems in actual failure situations. This may occur through the use of rules that are based on past experience, and the updating of certain rules based on the results of their present application. It is accomplished with the objective of returning the overall system to a good operating state. Fault compensation or managing involves both cognitive and physiological activities. Planning is a cognitive activity concerned with solving possible future problems in the sense of mentally generating a sequence of appropriate alternatives. Appropriate planning involves the use of knowledge perspectives, knowledge principles, and knowledge practices (Sage 1992). They are based on experiential familiarity with analogous situations and are often expressed in the form of and through the use of skills, rules, and formal knowledge based reasoning efforts (Rasmussen 1986; Rasmussen et al. 1994). Human error issues are of particular importance, especially those concerned with the design of systems that cope with human error through avoidance and amelioration efforts (Reason 1990).

Many of these systems can only be described as complex. While some components of them may be naturally adaptive, they often need to be engineered to possess adaptive characteristics. The subject of complex adaptive systems is closely related to that of complexity theory. Complexity theory (Kaufman 1995; Axelrod 1997; Holland 1998) is a field of study that has evolved from five major knowledge areas: mathematics, physics, biology, organizational science, and computational intelligence and engineering. Fundamentally, a system is complex if it cannot be understood through simple cause-and-effect relationships or other standard methods of systems analysis. In a complex system, the interplay of individual elements cannot be reduced to the study of individual elements considered in isolation. Often, several different models of the complete system, each at a different level of abstraction, are needed.

There are several sciences of complexity, and they generally deal with approaches to understanding the dynamic behavior of units that range from individual organisms to the largest technical, economic, social, and political organizations. Often, such studies involve complex adaptive systems and hierarchical systems, are multidisciplinary in nature, and involve or are at the limits of scientific knowledge (Arthur 1994; Coveney and Highfield 1995; Arthur et al. 1997; Epstein 1997).

Complexity studies attempt to pursue knowledge and discover features shared by systems described as complex. These include studies such as complex adaptive systems, complex systems theory, complexity theory, dynamic systems theory, complex nonlinear systems, and computational intelligence. Many scientific studies, prior to the development of simulation models and complexity theory, involved the use of linear models. When a study resulted in anomalous behavior, the failure was often incorrectly blamed on noise or experimental error. It is now recognized that such errors may reflect inherent inappropriateness of linear models — and linear thinking. Meeting the modeling challenge is complicated by the fact that not all critical phenomena cannot be fully understood, or even anticipated, based on analysis of the decomposed elements of the overall system. Complexity not only arises from there being many elements of the system, but also from the possibility of collective behaviors that even the participants in the system could not have anticipated (Casti 1997).

Thus, many critical phenomena can only be studied once they emerge. In other words, the only way to identify such phenomena is to let them happen. The challenge is to create ways to recognize the emergence of unanticipated phenomena and be able to manage their consequences, especially in situations where likely consequences are highly undesirable. One measure of system complexity is the complexity of the simulation model necessary to effectively predict system behavior (Casti 1997). The more the simulation model must embody the actual system to yield the same behavior, the more complex the system. In other words, outputs of complex systems cannot be predicted accurately based on models with typical types of simplifying assumptions. Consequently, creating models that will accurately predict the outcomes of complex systems is very difficult. A model can be created, however, that will accurately simulate the processes the system will use to create a given output.

This awareness has profound impacts for organizational efforts. For example, it raises concerns related to the real value of creating organizational mission statements and plans with expectations that these plans will be inexorably executed and missions thereby realized. It may be more valuable to create a model of an organization's planning processes themselves, subject this model to various input scenarios, and use the results to generate alternative output scenarios. The question then becomes one of how to manage an organization where this range of outputs is possible.

Interestingly, most studies of complex systems often run completely counter to the trend toward increasing fragmentation and specialization in most disciplines. Complexity studies tend to reintegrate the fragmented interests of most disciplines into a common pathway. This needed transdisciplinarity (Wilson 1998) provides the basis for creating a cohesive systems ecology (Sage 1998) to guide the use of information technology for managing complex systems. Whether they be human-made systems, human systems, or organizational systems, the use of systems ecology could more quickly lead to organizing for complexity (McMaster 1996), and associated knowledge and enterprise integration.

An important aspect of complex systems is path dependence (Arthur 1994). The essence of this phenomenon begins with a supposedly minor advantage or inconsequential head start in the marketplace for some technology, product, or standard. This minor advantage can have important and irreversible influences on the ultimate market allocation of resources, even if market participants make voluntary decisions and attempt to maximize their individual benefits. Such a result is not plausible with classical economic models that assume that the maximization of individual gain leads to market optimization unless the market is imperfect due to the existence of such effects as monopolies. Path dependence is a failure of traditional market mechanisms and suggests that users are locked into a sub-optimal product, even though they are aware of the situation and may know that there is a superior alternative.

This type of path lock-in is generally attributed to two underlying drivers: 1) network effects, and 2) increasing returns of scale. Both of these drivers produce the same result, namely that the value of a product increases with the number of users. network effects, or network externalities, occur because the value of a product for an individual consumer may increase with increased adoption of that product by other consumers. This, in turn, raises the potential value for additional users. An example is the telephone, which is only useful if at least one other person has one as well, and becomes increasingly

beneficial as the number of potential users of the telephone increases.

Increasing returns of scale imply that the average cost of a product decreases as higher volumes are manufactured. This effect is a feature of many knowledge-based products where high initial development cost dominates low marginal production and distribution cost. Thus, the average cost per unit decreases as the sales volume increases and the producing company is able to continuously reduce the price of the product. The increasing returns to scale, associated with high initial development costs and the low sales price, create barriers against market entry by new potential competitors, even though they may have a superior product.

The controversy in the late 1990s over the integration of the Microsoft Internet Explorer with the Windows Operating System may be regarded as a potential example of path dependence, and appropriate models of this phenomenon can potentially be developed using complexity theory. These would allow exploration of whether network effects and increasing returns of scale can potentially reinforce the market dominance of an established but inferior product in the face of other superior products, or whether a given product is successful because its engineers have carefully and foresightedly integrated it with associated products such as to provide a seamless interface between several applications.

Information technology enables systems where the interactions of many loosely structured elements can produce unpredictable and uncertain responses that may be difficult to control. The challenge is to understand such systems at a higher level. Control is likely to involve design and manipulation of incentives to participate and rewards for collaborative behaviors. It may be impossible and probably undesirable to control behaviors directly. The needed type of control is similar to policy formulation. Success depends on efficient experimentation much more than possibilities for mathematical optimization due to the inherent complexities that are involved. Thus, insights from complexity theory may be brought to bear on these situations (Merry 1995).

Information access and utilization, as well as management of the knowledge resulting from this process, are complicated in a world with high levels of connectivity and a wealth of data, information, and knowledge. The underlying problem is the usually tacit assumption that more information is inherently good to have. What users should do with this information and how value is provided by this usage are seldom clear. The result can be large investments in information technology with negligible improvements of productivity (Harris 1994). One of the major needs in this regard is to support bilateral transformations between tacit and explicit knowledge (Nonaka 1994; Nonaka and Takeuchi 1995).

Prior to the development of simulation models and complexity theory, most studies involved use of linear models and assumed time-invariant processes (i.e., ergodicity). Most studies also assumed that humans use deductive reasoning and techno-economic rationality to reach conclusions. But, information imperfections and limits on available time often suggest that rationality must be bounded. Other forms of rationality and inductive reasoning are necessary.

There are a number of descriptive models of human problem solving and decision making. Generally, the appropriate model depends upon the contingency task structure, characteristics of the environment, and the experiential familiarity of humans with tasks and environment. Thus, the context surrounding information and the experiential familiarity of users of the information is most important. In fact, it is the use of information within the context of contingency task structures and the environment that results in the transformation from information to knowledge.

It is appropriate to interpret knowledge in terms of context and experience by sensing situations and recognizing patterns. Features similar to previously recognized situations can thus be discerned. The problem can then be simplified by using these to construct internal models, hypotheses, or schemata to use on a temporary basis. Simplified deductions are attempted based on these hypotheses and one acts accordingly. Feedback of results from these interactions enables more to be learned about the environment and the nature of the task at hand. Hypotheses are revised, reinforcing appropriate ones and discarding poor ones. This use of simplified models is a central part of inductive behavior (Holland et al. 1986).

Models of inductive processes can be constructed in the following way. A collection of generally heterogeneous agents is first determined. It is assumed that the agents are able to form hypotheses based on mental models or subjective beliefs. Further, each agent is assumed to monitor performance relative to a personal

set of belief models. These models are based on the results of actions, as well as prior beliefs and hypotheses. Through this iterative procedure, learning takes place as agents discern which hypotheses are most appropriate. Hypotheses, or models, are retained not because they are correct, but because they have worked in the past. Agents differ in their approach to problems and the way in which they subjectively converge to a set of useful hypotheses.

This process may be modeled as a complex adaptive system. As noted, models cannot be created that will accurately predict the outcomes of many complex systems. But, a model can often be created that will accurately simulate the processes the system uses to create outputs. The major constructs associated with such models are: the interactions and feedback relations between the various agents whose choices depend upon the decisions of others, and linearity and return to scale considerations. There are many implications associated with these models. Among them are questions of steady state versus continued evolutionary behavior, the nature and possibility of time-invariant processes (ergodicity), and questions of path dependence.

The Design of Cybernetic and Complex Adaptive Systems

All of this has major implications with respect to the design of systems for the human user and for associated cybernetic and complex adaptive systems as well. It requires, for appropriate system design, an understanding of human performance in problem solving and decision-making tasks. This understanding has to be at a descriptive level, predicting what humans will likely do in particular situations. It has to be at a normative level, understanding what would be best performance under restrictive axiomatic conditions that will generally not exist in practice. Also, this understanding has to be at a prescriptive level such that humans can be aided in various real-world cognitive tasks. This requires much attention to the evolutionary and emergent properties of systems.

Technological advances have changed and will continue to change the specific design requirements for human-machine cybernetic systems needed in any given application area. This is especially true due to the many advances made possible through modern information technologies, for industrial plants with integrated automated manufacturing capabilities, and for aids to cognitive activities in strategic planning, design, or operational activities. Office automation systems and information systems for observation, planning, executive support, management, and command and control tasks in business, defense, and medicine are similarly influenced by efforts in human-machine cybernetic and complex adaptive systems. These involve not only the operation of technological and management oriented information systems by highly skilled and knowledgeable personnel, but also systems that are intended for use by the less skilled. A major use for new generation systems is to provide computer assistance for the maintenance of existing systems and for the design of new systems of all types.

The methods and tools for supporting emergence of a theory of complex systems that will fully satisfy the requirements posed by systems that must intentionally operate satisfactorily at the edge of chaos will always be in a state of continuous evolution. There seems little question that the methods of operations research and management science, especially those associated with modeling and simulation of large systems, have and will play a major role in the theory of design of cybernetic and complex adaptive systems. Addressing the key challenges requires utilizing many of the concepts, principles, methods, and tools of OR/MS. In addition, it will require a new, broader perspective on the nature of information access and utilization, as well as knowledge management. Fortunately, OR/ MS is an inherently dynamic field of study. However, achieving the goal of cybernetic and complex adaptive system understanding and development capability will require much attention to the integration of OR/MS approaches with those in disciplines not often involved in OR/MS studies and the development of knowledge unity and integration perspectives.

See

- ▶ Artificial Intelligence
- ▶ Control Theory
- ▶ Dynamic Programming
- ▶ Neural Networks
- ▶ Simulation Metamodeling

- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Simulation Optimization
- ▶ System Dynamics
- ▶ Systems Analysis

References

Arthur, W. B. (1994). *Increasing returns and path dependence in the economy*. Ann Arbor, MI: University of Michigan Press.

Arthur, W. B., Durlauf, S. N., & Lane, D. A. (Eds.). (1997). *The economy as an evolving complex system, II*. Reading, MA: Addison Wesley.

Ashby, W. R. (1952). *Design for a brain*. London: Chapman and Hall.

Ashby, W. R. (1956). *An introduction to cybernetics*. London: Chapman and Hall.

Axelrod, R. (1997). *The complexity of cooperation: Agent based models of competition and collaboration*. Princeton, NJ: Princeton University Press.

Bak, P., & Chen, K. (1991). Self organized criticality. *Scientific American, 271*(1), 46–53.

Bak, P. (1996). *How nature works: The science of self-organized criticality*. New York: Springer.

Barr, A., Cohen, P. R., & Feigenbaum, E. A. (Eds.). (1981/1982). *Handbook of artificial intelligence*, *Vols. I*, *II*, *and III*. Los Altos, CA: William Kaufman.

Beer, S. (1979). *The heart of enterprise*. Chichester, UK: Wiley.

Casti, J. L. (1997). *Would-be worlds how simulation is changing the frontiers of science*. New York: Wiley.

Coveney, P., & Highfield, R. (1995). *Frontiers of complexity: The search for order in a chaotic world*. Columbine, NY: Fawcett.

Davenport, T. H. (1993). *Process innovation: Reengineering work through information technology*. Boston: Harvard Business School Press.

Dreyfus, H. L. (1992). *What computers still can't do: A critique of artificial reason*. Cambridge, MA: MIT Press.

Drucker, P. (1995). *Managing in a time of great change*. New York: Dutton.

Drucker, P. (1998). *On the profession of management*. Boston: Harvard Business School Press.

Epstein, J. M. (1997). *Nonlinear dynamics, mathematical biology, and social science*. Reading, MA: Addison-Wesley.

Freeman, J. A., & Skapura, D. (1991). *Neural networks: Algorithms, applications and programming techniques*. Reading, MA: Addison-Wesley.

George, F. H. (1971). *Cybernetics*. Middlegreen, Slough, UK: St. Paul's House.

Harrington, H. J. (1991). *Business process improvement: The breakthrough strategy for total quality, productivity, and competitiveness*. New York: McGraw-Hill.

Harris, D. H. (Ed.). (1994). *Organizational linkages: Understanding the productivity paradox*. Washington, DC: National Academy Press.

Holland, J. H. (1996). *Hidden order: How adaptation builds complexity*. Reading, MA: Addison-Wesley.

Holland, J. H. (1998). *Emergence: From chaos to order*. Reading, MA: Addison-Wesley.

Holland, J. H., Holyoak, K. J., Nisbet, R. E., & Thagard, P. R. (1986). *Induction: Processes of inference, learning, and discovery*. Cambridge, MA: MIT Press.

Johannsen, G., Rijnsdorp, J. E., & Sage, A. P. (1983). Human interface concerns in support system design. *Automatica, 19*(6), 1–9.

Kaufman, S. (1995). *At home in the universe: The search for the laws of self-organization and complexity*. New York: Oxford University Press.

Langton, C. G. (1990). Computation at the edge of chaos: Phase transitions and emergent computation. *Physica D: Nonlinear Phenomena, 42*(1–3), 12–37.

Lerner, A. Y. (1976). *Fundamentals of cybernetics*. New York: Plenum.

Masani, P. (Ed.). (1985). *Norbert Wiener: Collected works volume IV–cybernetics, science and society; ethics, aesthetics, and literary criticism; book reviews and obituaries*. Cambridge, MA: MIT Press.

McMaster, M. D. (1996). *The intelligence advantage: Organizing for complexity*. Boston: Butterworth-Heinnemann.

Merry, U. (1995). *Coping with uncertainty: Insights from the new sciences of chaos, self-organization, and complexity*. Westport, CT: Praeger.

Nonaka, I. (1994). A dynamical theory of organizational knowledge creation. *Organizational Science, 5*(1), 14–37.

Nonaka, I., & Takeuchi, H. (1995). *The knowledge creating company*. New York: Oxford.

Rasmussen, J. (1986). *Information processing and human machine interaction: An approach to cognitive engineering*. Amsterdam: North Holland Elsevier.

Rasmussen, J., Pejtersen, A. M., & Goodstein, L. P. (1994). *Cognitive systems engineering*. New York: Wiley.

Reason, J. (1990). *Human error*. Cambridge, UK: Cambridge University Press.

Rockart, J. F., & DeLong, D. W. (1988). *Executive support systems: The emergence of top management computer use*. Homewood, IL: Dow Jones-Irwin.

Sage, A. P. (Ed.). (1987). *System design for human interaction*. New York: IEEE Press.

Sage, A. P. (1991). *Decision support systems engineering*. New York: Wiley.

Sage, A. P. (1992). *Systems engineering*. New York: Wiley.

Sage, A. P. (1995). *Systems management: For information technology and software engineering*. New York: Wiley.

Sage, A. P. (1998). Towards a systems ecology. *IEEE Computer, 31*(2), 107–110.

Sage, A. P., & Palmer, J. D. (1990). *Software systems engineering*. New York: Wiley.

Sage, A. P. (Ed.). (1990). *Concise encyclopedia of information processing in systems and organizations*. Oxford, UK: Pergamon Press.

Scott Morton, M. S. (Ed.). (1991). *The corporation of the 1990s: Information technology and organizational transformation*. New York: Oxford University Press.

Shapiro, S. C. (Ed.). (1987). *Encyclopedia of artificial intelligence*. New York: Wiley.

Sheridan, T. B. (1992). *Telerobotics, automation, and human supervisory control*. Cambridge, MA: MIT Press.

Sheridan, T. B., & Ferrell, W. R. (1974). *Man–machine systems: Information, control, and decision models of human performance*. Cambridge, MA: MIT Press.

Singh, M. G. (Ed.). (1990). *Systems and control encyclopedia*. Oxford, UK: Pergamon Press.
Steinbrunner, J. D. (1974). *The cybernetic theory of decision*. Princeton, NJ: Princeton University Press.
Wiener, N. (1948). *Cybernetics, or control and communication in the animal and the machine*. New York: Wiley.
Wiener, N. (1949). *Extrapolation, interpolation and smoothing of stationary time series with engineering applications*. Cambridge, MA: MIT Press.
Wilson, E. O. (1998). *Consilience: The unity of knowledge*. New York: Alfred A. Knopf.
Zurada, J. (1992). *Introduction to artificial neural systems*. St. Paul, MN: West Publishing.

Cycle

A path in a graph (network) joining a node to itself.

See

- ▶ Chain
- ▶ Path

Cyclic Queueing Network

A closed network of queues in which customer routing is serial.

See

- ▶ Networks of Queues
- ▶ Queueing Theory

Cyclic Service Discipline

When a congestion system with several different locations (service centers) of customers are served by a single service facility. For a given period of time determined by an a priori rule, the service process only works on customers from (at) a given location and then switches to the next group when the period is over.

See

- ▶ Queueing Theory

Cycling

A situation where the simplex algorithm cycles (circles) repeatedly through some sequence of bases and corresponding basic feasible solutions. This can occur at a degenerate extreme point solution where several bases correspond to the same extreme point.

See

- ▶ Anticycling Rules
- ▶ Degeneracy
- ▶ Linear Programming
- ▶ Simplex Method (Algorithm)

D

Dantzig-Wolfe Decomposition Algorithm

A variant of the simplex method designed to solve block-angular linear programs in which the blocks define subproblems. The problem is transformed into one that finds a solution in terms of convex combinations of the extreme points of the subproblems.

See

- ▶ Block-Angular System
- ▶ Decomposition Algorithms

References

Dantzig, G. (1963). *Linear programming and extensions*. Princeton, NJ: Princeton University Press.

Dantzig, G., & Thapa, M. (2003). *Linear programming 2: Theory and extensions*. New York: Springer.

Dantzig, G., & Wolfe, P. (1960). Decomposition principle for linear programs. *Operations Research, 8*(1), 101–111.

Data Envelopment Analysis

William W. Cooper
The University of Texas at Austin, Austin, TX, USA

Introduction

DEA (Data Envelopment Analysis) is a data oriented approach for evaluating the performance of a collection of entities called DMUs (Decision Making Units) which are regarded as responsible for converting inputs into outputs. Examples of its uses have included hospitals and U.S. Air Force Wings, or their subdivisions, such as surgical units and squadrons. The definition of a DMU is generic and flexible. The objective is to identify sources and to estimate amounts of inefficiency in each input and output for every DMU included in a study. Uses that have been accommodated include: (i) discrete periods of production in a plant producing semiconductors in order to identify when inefficiency occurred; and (ii) marketing regions to which advertising and other sales activities have been directed in order to identify where inefficiency occurred. Inputs as well as outputs may be multiple and each may be measured in different units.

A variety of models have been developed for implementing the concepts of DEA, for example, the following dual pair of linear programming models:

$$\begin{aligned} \min\ & h_0 = \theta_0 - \varepsilon\left(\sum_{i=1}^{m} s_i^- + \sum_{r=1}^{s} s_r^+\right) \\ \text{subject to } & 0 = \theta_0 \mathrm{x_{i0}} - \sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{x_{ij}}\,\lambda_{\mathrm{j}} - \mathrm{s_i^-} \\ & y_{r0} = \sum_{j=1}^{n} y_{rj}\,\lambda_j - s_r^+ \\ & 0 \le \lambda_j,\ s_r^+,\ s_i^- \end{aligned} \tag{1a}$$

and

$$\begin{aligned} \max\ & y_0 = \sum_{r=1}^{s} \mu_r\, y_{r0} \\ \text{subject to } & 1 = \sum_{\mathrm{i=1}}^{\mathrm{m}} \mathrm{v_i\, x_{i0}} \\ & 0 \ge \sum_{r=1}^{s} \mu_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} \\ & \varepsilon \le \mu_r,\ v_i \end{aligned} \tag{1b}$$

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

where x_{ij} = observed amount of input i used by DMU_j and y_{rj} = observed amount of output r produced by DMU_j, with $i = 1, \ldots, m$; $r = 1, \ldots, s$; $j = 1, \ldots, n$. All inputs and outputs are assumed to be positive. (This condition may be relaxed (Charnes et al. 1991).

Efficiency

The orientation of linear programming has changed here from ex-ante uses, for planning, and apply it to choices already made ex-post, for purposes of evaluation and control. To evaluate the performance of any DMU, (1) is applied to the input–output data for all DMUs in order to evaluate the performance of *each* DMU in accordance with the following definition:

> *Efficiency — Extended Pareto-Koopmans Definition* : Full (100%) efficiency is attained by any DMU if and only if none of its inputs or outputs can be improved without worsening some of its other inputs or outputs.

This definition has the advantage of avoiding the need for assigning a priori weights or other measures of relative importance to any input or output. In most management or social science applications, the theoretically possible levels of efficiency will not be known. For empirical use, the preceding definition is therefore replaced by the following:

> *Relative Efficiency*: A DMU is to be rated as fully (100%) efficient if and only if the performances of other DMUs do not show that some of its inputs or outputs can be improved without worsening some of its other inputs or outputs.

To implement this definition, it is necessary only to designate any DMU_j as DMU_0 with inputs x_{i0} and outputs y_{r0} and then apply (1) to the input and output data recorded for the collection of DMU_j, $j = 1, \ldots, n$. Leaving this $DMU_j = DMU_0$ in the constraints insures that solutions will always exist with an optimal $\theta_0 = \theta_0* \leq 1$. The above definition applied to (1) then gives

> *DEA Efficiency*: The performance of DMU_0 is fully (100%) efficient if and only if, at an optimum, both (i) $\theta_0* = 1$, and (ii) all slacks $= 0$ in (1a) or, equivalently, $\sum_{r=1}^{s} \mu_r^* y_{r0} = 1$ in (1b), where * represents an optimal value.

A value $\theta_0^* < 1$ shows (from the data) that a non-negative combination of other DMUs could have achieved DMU_0's outputs at the same or higher levels while reducing *all* of its inputs. Non-zero slacks similarly show where input reductions or output augmentations can be made in DMU_0's performance without altering other inputs or outputs. These non-zero slacks show where changes in *mixes* could have improved performance in each of DMU_0's inputs or outputs, while a $\theta_0^* < 1$ shows "technical inefficiency" in which *all* inputs could have been reduced in the same proportion. (This is a so-called input-oriented model. An output-oriented model can be similarly formulated by associating a variable φ_0 with all outputs to be maximized DMU_0. The measures are reciprocal, i.e., $\varphi_0^* \; \theta_0^* = 1$, so this topic is not developed here.)

Many applications to many different kinds of entities engaged in complex activities with no clearly defined bottom line have been reported in many publications by many different authors in many different countries. Examples include applications to schools (including universities), police forces, military units, and country performances (including United Nations evaluations of country performances). See, for example, Emrouznejad et al. (2008) who list more than 1,600 published papers by more than 2,500 different authors in more than 40 different countries. Also see Berber et al. (2011) and Cooper et al. (2009).

Farrell Measure

The scalar θ_0^* is sometimes referred to as the Farrell measure after M.J. Farrell (1957). Notice, however, that a value of $\theta_0^* = 1$ does not completely satisfy the above definition of Relative Efficiency if any of the associated slacks, s_i^{+*} or s_r^{+*}, in (1) are positive — because any such non-zero slack provides an opportunity for improvement which may be used without affecting any other variable, as should be clear from the primal problem which is shown in (1a).

There is a need to insure that an optimum with $\theta_0^* = 1$ and all slacks zero is not interpreted to mean that full (100%) efficiency has been attained when an alternate solution with $\theta_0^* = 1$ and some slacks positive is also available. To see how this is dealt with, attention is called to the fact that the slack variables s_i^- and s_r^+ in the objective of the primal (minimization) problem, (1a), are each multiplied by

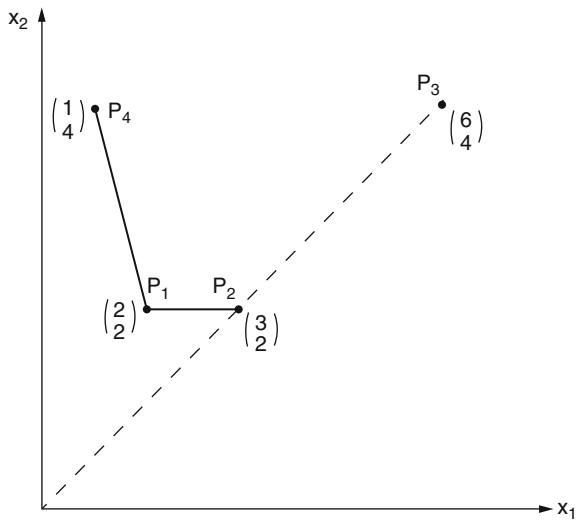
$\varepsilon > 0$ which is a non-Archimedean infinitesimal — the reciprocal of the "big M" associated with the artificial variables in ordinary linear programming — so that choices of slack values *cannot* compensate for any increase they might cause in θ_0. This accords pre-emptive status to the minimization of θ_0, and DEA computer codes generally handle optimizations in a two-stage manner which avoids the need for specifying ε explicitly. Formally, this amounts to minimizing the value of θ_0 in stage 1. Then one proceeds in a second stage to maximize the sum of the slacks with the condition $\theta_0 = \theta_0^*$ fixed for the primal in (1a). Since the sum of the slacks is maximized, one can be sure that a solution with all slacks at zero in the second stage means that DMU_0 is fully efficient if the first stage yielded $\theta_0^* = 1$.

N.B. Weak efficiency is another term used instead of Farrell efficiency when attention is restricted to (i) in DEA Efficiency above. It is also referred to as a measure of technical efficiency. However, when (1a) is used, this might be referred to as purely technical efficiency in order to distinguish these inefficiencies from the mix inefficiencies associated with changes in the proportions used that are then associated with non-zero slack. The term technical efficiency can then be used to comprehend both purely technical and mix inefficiencies as determined by reference to technical conditions without recourse to prices, costs, and/or subjective evaluations.

Example

Figure 1 is a geometric portrayal of four DMUs interpreted as points $P_1, \ldots, P_4$, with coordinate values corresponding to the amounts of two inputs which each DMU used to produce the same amount of a single output. P_3 is evidently inefficient compared to P_2 because it used more of both inputs to achieve the same output. In fact, its Farrell measure of inefficiency relative to P_2 can be determined via the formula

$$\theta_0 = \frac{d(0,\ P_2)}{d(0,\ P_3)} = \frac{\sqrt{3^2 + 2^2}}{\sqrt{6^2 + 4^2}} = \frac{1}{2},$$



Data Envelopment Analysis, Fig. 1 DEA efficiencies

where $d(.,.)$ refers to the Euclidean, or l_2, measure of distance.

Referred to as a radial measure of efficiency in the DEA literature, θ_0 is really a ratio of two distance measures, namely, the distance along the ray from the origin to the point being evaluated relative to the distance from the origin to the frontier measured along this same ray. This same value of θ_0 is obtained, and hence this same radial measure, by omitting the slacks and rewriting the primal problem in (1a) in the following inequality form,

$$\begin{aligned} &\text{minimize } \theta_0 \\ &\text{subject to} \\ &6\theta_0 \ \geq\ 2\lambda_1 + 3\lambda_2 + 6\lambda_3 + 1\lambda_4 \\ &4\theta_0 \ \geq\ 2\lambda_1 + 2\lambda_2 + 4\lambda_3 + 4\lambda_4 \\ &1 \ \leq\ 1\lambda_1 + 1\lambda_2 + 1\lambda_3 + 1\lambda_4 \\ &0 \ \leq\ \lambda_1,\ \ldots,\ \lambda_4, \end{aligned} \tag{2}$$

where the third constraint reflects the output $y = 1$ which was produced by each of these DMUs.

An optimum is achieved with $\theta_0^* = 1/2$, $\lambda_2^* = 1$ and this designates P_2 for the evaluation of P_3. However, it is also needed to take account of the slack possibilities. This is accomplished without specifying $\varepsilon > 0$ explicitly by proceeding to

a second stage by using the thus obtained value of θ_0^* to form the following problem:

$$\begin{aligned}
&\text{maximize } s_1^- + s_2^- + s^+ \\
&\text{subject to} \\
&0 = -6\theta_0 + 2\lambda_1 + 3\lambda_2 + 6\lambda_3 + 1\lambda_4 + s_1^- \\
&0 = -4\theta_0 + 2\lambda_1 + 2\lambda_2 + 4\lambda_3 + 4\lambda_4 + s_2^- \\
&-1 = -1\lambda_1 - 1\lambda_2 - 1\lambda_3 - 1\lambda_4 + s^+ \\
&0.5 = \theta_0 \\
&0 \le \lambda_1, \ldots, \lambda_4, s_1^-, s_2^-, s^+
\end{aligned} \tag{3}$$

Following through in this second stage, with $\theta_0^* = 0.5$, it can be found that $\lambda_2^* = 1$ and $s_1^{-*} = 1$, with all other variables zero. This solution is interpreted to mean that the evidence from other DMUs (as exhibited by P_1's performance) shows that P_3 should have been able (a) to reduce both inputs to one-half their observed values, as given by the value of θ_0, and should also have been able (b) to reduce the first input by the additional amount given by $s_1^{-*} = 1$.

This slack, $s_1^{-*} = 1$, represents the excess amount of the first input used by P_2, and it, too, must be accounted for if the above definition of relative efficiency is to be satisfied. In fact, using the primal in (1a) to evaluate P_2, it will be found that it is also inefficient with $\theta_1{}^* = 1$ and $\lambda^* = s_1^{-*} = 1$. The use of (1a) to determine whether the conditions (i) and (ii) for relative efficiency are satisfied has a further consequence in that it insures that only efficient DMUs enter into the solutions with positive coefficients in the basis sets that are used to effect efficiency evaluations. Computer codes that have been developed for DEA generally use this property to reduce the number of computations by identifying all such members of an optimal basis as efficient and, hence, not in need of further evaluation.

As can be seen from Fig. 1, P_1 dominates P_2 and hence also dominates P_3. Only P_1 and P_4 are not dominated and hence can be regarded as efficient when DEA is restricted to dominance, as in Bardhan et al. (1996). However, if an assumption of continuity is added, then the entire line segment connecting P_1 and P_4 becomes available for use in effecting efficiency evaluations. This line segment is referred to as the efficiency frontier. The term efficient frontier is appropriate because it is not possible to move from one point to another on the line connecting P_1 and P_4 without worsening one input to improve the other input.

Given the assumption of continuity, points not on the efficiency frontier are referred to it for evaluation. Even when not dominated by actually observed performances, the nonnegative combinations of λ_j^* and slack values will locate points on the frontier which can be used for effecting efficiency evaluations of any DMU in the observation set.

The following formulas, called the CCR projection formulas, may be used to move points up to the efficiency frontier:

$$\begin{cases} \hat{x}_{i0} = \theta_0^* \hat{x}_{i0} - s_i^{-*} \le \hat{x}_{i0}, & i = 1, \ldots, m \\ \hat{y}_{r0} = y_{r0} + s_r^{+*} \ge y_{r0}, & r = 1, \ldots, s \end{cases} \tag{4}$$

where each $(\hat{x}_{i0}, \hat{y}_{i0})$ represents a point on the efficiency frontier obtained from (x_{i0}, y_{r0}), DMU_0's observed values. The point on the efficiency frontier thus obtained from these CCR projections is the point used to evaluate (x_{i0}, y_{r0}), $i = 1, \ldots, m$; $r = 1, \ldots, s$, for any DMU_0.

Ratio Form Models

The name Data Envelopment Analysis is derived from the primal (minimization) problem (1a) by virtue of the following considerations. The objective is to obtain as tight a fit as possible to the input–output vector for DMU_0 by enveloping its observed inputs from below and its observed outputs from above. As can be seen from (1a), an optimal envelopment will always involve a touching of the envelopment constraints to at least one of DMU_0's inputs and one of its outputs.

The primal problem, (1a), is said to be in envelopment form. The dual problem, (1b), is said to be in multiplier form by reference to the values of μ and v as dual multipliers. The objective is to maximize y_0, which is called the virtual output. This maximization is subject to the condition that the corresponding virtual input is unity, that is, $\sum_{i=1}^{m} v_i x_{i0} = 1$, as given in the first constraint. The other constraints require that the virtual output cannot exceed virtual input for any of the DMU_j, $j = 1, \ldots, n$, that is,

$$\sum_{r=1}^{s} \mu_r y_{rj} \le \sum_{i=1}^{m} v_i x_{ij} \quad j = 1, \ldots, n.$$

Finally, the conditions $\mu_r, v_i \geq \varepsilon > 0$ mean that every input and every output is to be assigned "some" positive value in this "multiplier" form, where as previously noted, the value of ε need not be specified explicitly.

To add interpretive power for the use in DEA, all of the variables in (1b) are multiplied, the (dual) problem of (1a), by $t > 0$ and then introduce new variables defined in the following manner:

$$\mu_r = t\mu_r \geq t\varepsilon, \; v_i = tv_i \geq t\varepsilon, \qquad t = \sum_{i=1}^{m} tv_i \, x_{i0}. \tag{5}$$

Multiplying and dividing the objective of the dual problem in (1b) by $t > 0$ and then multiplying all constraints by t gives the following model, which accords a ratio form to the DEA evaluations:

$$\begin{aligned} \max \quad & \frac{\sum_{r=1}^{s} u_r \, y_{r\,0}}{\sum_{i=1}^{m} v_i \, x_{i0}} \\ \text{subject to} \quad & \frac{\sum_{r=1}^{s} \mathrm{u_r y_{rj}}}{\sum_{i=1}^{m} \mathrm{v_i \, x_{ij}}} \leq 1, \quad \mathrm{j} = 1, \ldots, \mathrm{n} \\ & \frac{u_r}{\sum_{i=1}^{m} v_i x_{i0}} \geq \varepsilon, \quad r = 1, \ldots, s \\ & \frac{v_i}{\sum_{i=1}^{m} v_i \, x_{i0}} \geq \varepsilon, \quad i = 1, \ldots, m. \end{aligned} \tag{6}$$

An immediate corollary from this development is

$$0 \leq \frac{\sum_{r=1}^{s} u_r^* y_{r0}}{\sum_{i=1}^{m} v_i^* x_{i\,0}} = \sum_{r=1}^{s} u_r^* y_{r0} = \theta_0^* \qquad -\sum_{i=1}^{m} s_i^{-*} + \sum_{r=1}^{s} s_r^{+*} \leq 1, \tag{7}$$

where "*" designates an optimal value. Thus, in accordance with the theory of fractional programming, as given in Charnes and Cooper (1962), the optimal values in (6) and (1b) are equal.

The formulation (6) has certain advantages. For instance, Charnes and Cooper (1985) used it to show that the optimal ratio value in (6) is invariant to the units of measure used in any input and any output and, hence, this property carries over to (1b). Equation 6 also add interpretive power and provide a basis for unifying definitions of efficiency that stretch across various disciplines. For instance, as shown in Charnes et al. (1978), the usual single-output to single-input efficiency definitions used in science and engineering are derivable from (6). It follows that these definitions contain an implicit optimality criterion. The relation of (6) to (4), established via fractional programming, also relates these optimality conditions to the definitions of efficiency used in economics. (See the above discussion of Pareto-Koopmans efficiency.) This accords a ratio form (as well as a linear programming form) to the DEA evaluations.

As (6) makes clear, DEA also introduces a new principle for determining weights. In particular the weights are not assigned a priori, but are determined directly from the data. A best set of weights is determined for each of the $j,\ldots, n$ DMUs to be evaluated. Given this set of best weights the test of inefficiency for any DMU_0 is whether any other DMU_j achieved a higher ratio value than DMU_0 using the latter's best weights [Care needs to be exercised in interpreting these weights, since (a) their values will in general be determined by reference to different collections of DMUs and (b) when determined via (1), allowance needs to be made for non-zero slacks. See the discussion in Charnes et al. (1989), where dollar equivalents are used to obtain a complete ordering to guide the use of efficiency audits by the Texas Public Utility Commissions].

DEA also introduces new principles for making inferences from empirical data. This flows from its use of n optimizations — to come as close as possible to *each* of n observations — in place of other approaches, as in statistics, for instance, which uses a single optimization to come as close as possible to all of these points. In DEA, it is also not necessary to specify the functional forms explicitly. These forms may be nonlinear and they may be multiple (differing, perhaps, for each DMU) provided they satisfy the mathematical property of isotonicity (Charnes et al. 1985).

Other Models

The models in (1) and (6) are a subset of several DEA models that are now available. Thus, DEA may be regarded as a body of concepts, and methods which unite these models and their uses to each other. These concepts, models and methods comprehend extensions to identify scale, and allocative and other inefficiencies.

By virtue of the already described relations between (6) and (1) the models are referred to as the CCR ratio model. Other models include the additive model, namely,

$$\begin{aligned} &\max \sum_{i=1}^{m} s_i^- + \sum_{r=1}^{s} s_r^+ \\ &\text{subject to} \\ &0 = \hat{x}_{i0} - \sum_{j=1}^{n} \hat{x}_{ij}\lambda_j - s_i^- \\ &\hat{y}_{r0} = \sum_{j=1}^{n} \hat{y}_{rj}\lambda_j - s_r^+ \\ &0 \le \lambda_j,\ s_r^+,\ s_i^-;\ \ \forall i, j, r \end{aligned} \tag{8}$$

for which the conditions for efficiency are given by Additive Model Efficiency: DMU_0 is fully (100%) efficient if and only if all slacks are zero — namely, $s_i^{-*}, s_r^{+*} = 0, \forall\, i, r$ in (8).

With the constraint $\sum_{j=1^n} \lambda_j = 1$ adjoined, the model (8) becomes "translation invariant." That is, as shown by Ali and Seiford (1990), the solution to (8) is not altered if the original data $(\hat{x}_{ij},\hat{y}_{rj})$ are replaced by new data

$$\begin{aligned} \hat{x}'_{ij} &= \hat{x}'_{ij} + d_i, \quad i = 1, \ldots, m \\ \hat{y}'_{rj} &= \hat{y}'_{rj} + c_r, \quad r = 1, \ldots, s \end{aligned} \tag{9}$$

where the d_i and c_r are arbitrarily constants. This property can be of value in treating negative data since most theorems in DEA assume that the data are positive or at least semi-positive. See Pastor (1996) for examples and extensions of the Ali-Seiford theorems. Theorems like the following from Ahn et al. (1988) relate the additive models to their CCR counterparts.

Theorem: A DMU_0 will be evaluated as fully (100%) efficient by the CCR model if and only if it is rated as fully (100%) efficient by the corresponding additive model.

Note, however, that the CCR and additive models use different metrics, so they need not identify the same sources and amounts of inefficiency in an inefficient DMU.

The additive model (8) can also be related to another class, called multiplicative models (Charnes et al. 1982). An easy way is to assume that the $(\hat{x}_{ij},\hat{y}_{rj})$ are stated in logarithmic units. Taking antilogs then gives

$$\begin{aligned} x_{i0} &= a_i^* \prod_{j=1}^{n} x_{ij}^{\lambda_j^*}, \quad i = 1, \ldots, m, \\ y_{r0} &= b_r^* \prod_{j=1}^{n} y_{rj}^{\lambda_j^*}, \quad r = 1, \ldots, s, \end{aligned} \tag{10}$$

where $a_i^* = e^{s-*i}$, $b_r^* = e^{s+*r}$, and the (x_{ij}, y_{rj}) are stated in natural units. Each x_{i0}, y_{r0} is thus generated by a Cobb-Douglas process with estimated parameters given by the starred values of the variables.

To relate these results to a ratio form for efficiency evaluation, the dual to (8) is written as

$$\begin{aligned} &\min \sum_{i=1}^{m} v_i \hat{x}_{i0} - \sum_{r=1}^{s} \mu_r \hat{y}_{r0} \\ &\text{subject to} \\ &\sum_{i=1}^{m} v_i \hat{x}_{ij} - \sum_{r=1}^{s} \mu_r \hat{y}_{rj} \ge 0, \quad j = 1, \ldots, n \\ &v_i,\ \mu_r \ge 1, \quad i = 1, \ldots, m; \quad r = 1, \ldots, s, \end{aligned} \tag{11}$$

where the $(\hat{x}_{ij}, \hat{y}_{rj})$ are stated in logarithmic units. Recourse to antilogarithms then produces

$$\begin{aligned} &\max \prod_{r=1}^{s} \hat{y}_{r0}^{\mu r} \Big/ \prod_{i=1}^{m} \hat{x}_{i0}^{v_i} \\ &\text{subject to} \\ &\prod_{r=1}^{s} \hat{y}_{rj}^{\mu r} \Big/ \prod_{i=1}^{m} \hat{x}_{i0}^{v_i} \le 1, \quad j = 1, \ldots, n \\ &v_i, \mu_r \ge 1, \quad i = 1, \ldots, m; \quad r = 1, \ldots, s, \end{aligned} \tag{12}$$

and we once again make contact with a ratio form for effecting efficiency evaluations.

To obtain conditions for efficiency, antilogs to (8) are applied and (10) is used to obtain

$$\max \frac{\prod_{r=1}^{s} e^{s_r^{+*}}}{\prod_{i=1}^{m} e^{-s_i^{-*}}} = \frac{\prod_{r=1}^{s}\prod_{j=1}^{n} y_{rj}^{\lambda_j^*}/y_{r0}}{\prod_{i=1}^{m}\prod_{j=1}^{n} x_{ij}^{\lambda_j^*}/x_{i0}} \geq 1. \qquad (13)$$

The lower bound on the right is obtainable if and only if all slacks are zero. Thus the efficiency conditions for the multiplicative model are the same as for the additive model.

An interpretation of (13) can be secured by noting that

$$\left(\prod_{j=1}^{n} y_{rj}^{\lambda_j^*}\right)^{1\Big/\sum_{j=1}^{n}\lambda_j^*}, \quad \left(\prod_{j=1}^{n} x_{ij}^{\lambda_j^*}\right)^{1\Big/\sum_{j=1}^{n}\lambda_j^*}$$

represent weighted geometric means of outputs and inputs, respectively. Thus (13) is a ratio of the product of weighted geometric totals relative to the outputs and inputs which each of these expressions is evaluating.

It is necessary to note that the results in (13) are not units invariant (i.e., they are not dimension free in the sense of dimensional analysis) except in the case of constant returns to scale (see Thrall, 1996). This property, when wanted, can be secured by adjoining $\sum_{j=1^n}\lambda_j = 1$ to (8). See also Charnes et al. (1983). To conclude this discussion it is noted that the expression on the left of (13) is simpler and easier to interpret and the computations from (8) are straightforward.

The class of multiplicative models has not been much used, possibly because other models are easier to comprehend. Even allowing for this, however, they have potentials for use either on their own or in combination with other DEA models as when, for instance, returns to scale characterization are needed that differ from those which are available from other types of DEA models. See Banker and Maindiratta (1986) for further discussion of such uses.

Extensions and Uses of Dea Models

1. *Returns to Scale* — There is an extensive literature on returns to scale and their uses in DEA which reflects two different approaches. One approach, due to Färe et al. (1985, 1994) proceeds in an axiomatic manner and employs only radial measures. The other approach is based on mathematical programming. Conceptualized by Banker et al. (1984), it was subsequently ex-tended (and made wholly rigorous) by Banker and Thrall (1992). As might be expected, equivalences between the two approaches have been established in (among other places) Banker et al. (1996). See also Banker et al. (1998).
2. *Returns to Scope* — Partly because of difficulty in assembling data in pertinent forms, the literature on returns to scope is relatively sparse in DEA. Indeed, a bare beginning has been made in Chapter 10 of Färe et al. (1994).
3. *Assurance Regions and Allocative Inefficiency* — Many other developments have occurred and continue to occur. Thompson, Dharmapala and Thrall and their associate introduced the now widely used concept of assurance regions (Thompson et al. 1986; Dyson and Thanassoulis, 1988). This approach uses a priori knowledge to set upper and lower bounds on the values of the multiplier variables in DEA models like (1b). This can alleviate problems encountered in treating allocative or price efficiency either because (i) exact data on prices, costs, etc., are not available, or (ii) because the presence of wide variations in these data make the use of exact value a questionable undertaking. See Schaffnit et al. (1997), where limiting arguments are used to establish an exact relation between allocative efficiency and the bounds used in assurance region approaches.
4. *Cone Ratio Envelopments* — In a similar spirit, but in a different manner, Charnes et al. (1990) and their associates developed what they refer to as a cone-ratio envelopment approach. In contrast to the assurance region treatments of bounds on the variables, these cone-ratio approaches utilize a priori information to adjust the data. This makes it possible to take account of complex (multiple) considerations that might otherwise be difficult to articulate. See Brockett et al. (1997), who show how to implement the Basle Agreement, which was recently adopted by U.S. bank regulators to treat multiple risk factors in banking by adjusting the data reported in the FDIC call reports. These regulations are rigid and ill-fitting, so Brockett

et al. (1997) provide an alternative Cone-ratio envelopment approach which uses results from excellent banks (that are also found to be efficient) to adjust the call-report data for other banks in a use of DEA to effect such risk-adjusted evaluations.

5. *Exogenous and Categorical Variables* — Other important developments include methods for treating input or output values which are exogenously fixed for some, or all, DMUs. Developed by Banker and Morey (1986a) for treating demographic variables as important inputs in different locations for a chain of fast food outlets, these methods have found widespread use in many other applications. Similar remarks apply to the Banker and Morey (1986b) introduction of methods for treating categorical (classificatory) variables in work which has since been modified and extended by other authors; see Neralić and Wendell (2000).
6. *Statistical Treatments* — Various attempts have recently been made to join statistical and probabilistic characterizations to the deterministic models and methods of inference in DEA. For instance, using relatively mild postulates, Banker (1993) has shown that (i) DEA estimators of θ_0^* are statistically consistent; (ii) DEA estimates maximize the likelihood of obtaining the corresponding true values; and (iii) these properties hold under fairly general structures that do not require assumptions about the parametric forms of the probability density functions. See pages 272–275 in Banker and Cooper (1994) for a succinct discussion. See also Korostelev et al. (1995), who show that the rates of convergence are slow.

   Simar and Wilson (1998) utilize bootstrap procedures to study sampling properties of the efficiency measures in DEA. Unlike Banker, who restricts his analysis to the single output case, this bootstrap approach accommodates multiple outputs as well as multiple inputs. Omitted, however, is any treatment of nonzero slacks. Brockett and Golany (1996) also approach the topic of statistical characterizations by means of Mann–Whitney rank order statistics, but do not note that need for explicitly stating a ranking principle. This is needed because (as noted above) the DEA efficiency scores are generally determined relative to different reference sets (or peer groups) of efficient DMUs. (For a discussion of how this problem is treated for the efficiency audits conducted by Texas Public Utility Commission, see Charnes et al. 1989).
7. Probabilistic Models — Alternate approaches via chance constrained programming were initiated by Land et al. (1994) and have been ex-tended by others to include the use of joint chance constraints in addition to the conditional chance constraints used by Land, Lovell and Thore (Olesen and Petersen 1995; Cooper et al. 1998). Of special interest is the use of chance constraints to obtain a satisficing approach for efficiency evaluation, as in Cooper et al. (1996), where the term satisficing is used in the sense of H.A. Simon's (1957) behavioral characterizations in terms of (i) achievement of a satisfactory level of efficiency, and (ii) a satisfactory probability (=chance) of achieving this level. Finally, allowance is also made for situations in which these levels or probabilities may need to be revised because the data show that they are not possible of attainment.
   Unlike the statistical characterizations described in item 6, these chance constrained programs generally require knowledge of the parameters as well as the forms of the probability functions so that here, too, there is more work to be done. See Jagannathan (1985) for a start.
8. *Cross-Checking* — As noted in the earlier discussions, the inference principles in DEA differ from those in statistics. This suggests additional possibilities for their joint use. One such possibility is to use the two approaches as cross checks on each other to help avoid what is referred to as methodological bias in Charnes et al. (1988). See also Ferrier and Lovell (1990).
9. *Complementary Uses* — Another possibility is to use statistics and DEA in a complementary manner. An example is provided by Arnold et al. (1996), who applied this strategy in a two-stage manner to a study of Texas public schools as follows. At stage 1, DEA is used to identify efficient schools; then, at stage 2, these results are incorporated as dummy variables in an OLS (Ordinary Least Squares) regression. This yielded very satisfactory results on data which had previously yielded unsatisfactory results with an OLS regression. A subsequent simulation study by Bardhan et al. (1998)

compares this approach not only to OLS but also to stochastic frontier regressions (i.e., regressions which apply statistical principles to obtain frontier estimates for efficiency evaluations). Using observations that reflected mixtures of efficient and inefficient performances the OLS and SF approaches always failed to provide correct estimates whereas, with only one minor exception, the complementary two-stage use of DEA and statistics always yielded estimates that did not differ significantly from the true parameter values.

Sources and References

As the above discussions suggest, many important developments have been effected in DEA since its initiation by Charnes et al. (1978). These developments have occurred pari passu with numerous and widely varied applications of DEA which are being reported from many different parts of the world. See the bibliography by Seiford (1994). For a comprehensive text, see Cooper et al. (1999).

See

- ▶ Dual Linear-Programming Problem
- ▶ Fractional Programming
- ▶ Linear Programming

References

Ahn, T., Charnes, A., & Cooper, W. W. (1988). A note of the efficiency characterizations obtained in different DEA models. *Socio-Economic Planning Sciences, 22*, 253–257.

Ali, A. I., & Seiford, L. M. (1990). Translation invariance in data envelopment analysis. *Operations Re-search Letters, 9*, 403–405.

Arnold, V., Bardhan, I., & Cooper, W. W. (1993). *DEA models for evaluating efficiency and excellence in Texas Secondary Schools* (Working Paper, IC2) Austin: Institute of the University of Texas.

Arnold, V., Bardhan, I., Cooper, W. W, & Gallegos, A. (1984). Primal and dual optimality in ideas (integrated data envelopment analysis systems) and related computer codes. *Proceeding of a Conference in Honor of G.L. Thompson*, Quorum Books

Arnold, V., Bardhan, I., Cooper, W. W., & Gallegos, A. (1998). Primal and dual optimality in IDEAS (Integrated Data Envelopment Analysis Systems) and related computer codes, operations research: Methods, models and applications. *Proceedings of a Conference in Honor of G.L. Thompson*, Westport, CT: Quorum Books.

Arnold, V., Bardhan, I., Cooper, W. W., & Kumbhakar, S. (1996). New uses of DEA and statistical regressions for efficiency evaluation and estimation–with an illustrative application to Public Secondary Schools in Texas. *Annals Operations Research, 66*, 255–278.

Banker, R. D. (1993). Maximum likelihood, consistency and data envelopment analysis: A statistical foundation. *Management Science, 39*, 1265–1273.

Banker, R. D., Chang, H. S., & Cooper, W. W. (1996). Equivalence and implementation of alternative methods for determining returns to scale in data envelopment analysis. *European Journal Operational Research, 89*, 473–481.

Banker, R. D., Charnes, A., & Cooper, W. W. (1984). Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Management Science, 30*, 1078–1092.

Banker, R. D., & Cooper, W. W. (1994). Validation and generalization of DEA and its uses. *TOP (Sociedad de Estadistica e Investigación Operativa), 2*, 249–297. (with discussions by E. Grifell-Tatje, J. T. Pastor, P. W. Wilson, E. Ley and C. A. K. Lovell).

Banker, R.D., Cooper, W.W., & Thrall, R. M. (1998). *Finished and unfinished business for returns to scale in DEA*. Research Report, Graduate School of Business, University of Texas at Austin.

Banker, R. D., & Maindiratta, A. (1986). Piecewise loglinear estimation of efficient production surfaces. *Management Science, 32*, 385–390.

Banker, R. D., & Morey, R. C. (1986a). Efficiency analysis for exogenously fixed inputs and outputs. *Operations Research, 34*, 513–521.

Banker, R. D., & Morey, R. C. (1986b). Data envelopment analysis with categorical inputs and outputs. *Management Science, 32*, 1613–1627.

Banker, R. D., & Thrall, R. M. (1992). Estimation of returns to scale using data envelopment analysis. *European Journal Operational Research, 62*, 74–84.

Bardhan, I., Bowlin, W. F., Cooper, W. W., & Sueyoshi, T. (1996). Models and measures for efficiency dominance in DEA. Part I: Additive models and med measures. Part II: Free disposal hulls and Russell measures. *Journal of the Operations Research Society Japan, 39*, 322–344.

Bardhan, I. R., Cooper, W. W., & Kumbhakar, S. C. (1998). A simulation study of joint uses of DEA and statistical regression for production function estimation and efficiency evaluation. *Journal Productivity Analysis, 9*, 249–278.

Berber, P., et al. (2011). Efficiency in fundraising and distributions to cause-related social profit enterprises. *Socio-Economic Planning Sciences, 45*, 1–9.

Bowlin, W. F., Brennan, J., Cooper W. W., & Sueyoshi, T. (1984). *A DEA model for evaluating efficiency dominance*, Research Report. Texas: Center for Cybernetic Studies, Austin, (submitted for publication).

Brockett, P. L., Charnes, A., Cooper, W. W., Huang, Z. M., & Sun, D. B. (1997). Data transformations in DEA cone ratio envelopment approaches for monitoring bank performances. *European Journal of Operational Research, 95*, 250–268.

Charnes, A., & Cooper, W. W. (1962). Programming with linear fractional functionals. *Naval Research Logistics Quarterly, 9*, 181–186.

Charnes, A., & Cooper, W. W. (1985). Preface to topics in data envelopment analysis. In R. Thompson & R. M. Thrall (Eds.), *Annals operations research* (Vol. 2, pp. 59–94).

Charnes, A., Cooper, W. W., Divine, D., Ruefli, T. W., & Thomas, D. (1989). Comparisons of DEA and existing ratio and regression systems for efficiency evaluations of regulated electric cooperatives in Texas. *Research in Governmental and Nonprofit Accounting, 5*, 187–210.

Charnes, A., Cooper, W. W., Golany, B., Seiford, L., & Stutz, J. (1985). Foundations of data envelopment analysis and Pareto-Koopmans efficient empirical production functions. *Journal of Econometrics, 30*, 91–107.

Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring efficiency of decision making units. *European Journal of Operational Research, 1*, 429–444.

Charnes, A., Cooper, W. W., Seiford, L., & Stutz, J. (1982). A multiplicative model for efficiency analysis. *Socio-Economic Planning Sciences, 16*, 223–224.

Charnes, A., Cooper, W. W., Seiford, L., & Stutz, J. (1983). Invariant multiplicative efficiency and piecewise Cobb-Douglas envelopments. *Operations Research Letters, 2*, 101–103.

Charnes, A., Cooper, W. W., & Sueyoshi, T. (1988). Goal programming-constrained regression review of the bell system breakup. *Management Science, 34*, 1–26.

Charnes, A., Cooper, W. W., Sun, D. B., & Huang, Z. M. (1990). Polyhedral cone-ratio DEA models with an illustrative application to large commercial banks. *Econometrics Journal, 46*, 73–91.

Charnes, A., Cooper, W. W., & Thrall, R. M. (1991). A structure for classifying and characterizing efficiency and inefficiency in data envelopment analysis. *Journal of Productivity Analysis, 2*, 197–237.

Cooper, W. W., Huang, Z. M., & Li, S. (1996). Satisficing DEA models under chance constraints. *Annals Operations Research, 6*, 279–295.

Cooper, W. W., Huang, Z., Lelas, V., Li, X. S., & Olesen, O. B. (1998). Chance constrained programming formulations for stochastic characterizations of efficiency and dominance in DEA. *Journal of Productivity Analysis, 9*, 53–79.

Cooper, W. W., Seiford, L. M., & Tone, K. (1999). *Data envelopment analysis*. Boston, MA: Kluwer Academic Publishers.

Cooper, W. W., Seiford, L. M., & Zhu, J. (2011). *Handbook on data envelopment analysis*. New York: Springer.

Cooper, W. W., Thore, S., & Traverdyan, R. (2009). A utility function approach for evaluating country performances — The twin goals of decent work and affair globalization. In R. R. Hockley (Eds.), *Global operations management*. NOVA Science Publishers.

Dyson, R. G., & Thanassoulis, E. (1988). Reducing weight flexibility in data envelopment analysis. *Journal of Operational Research Society, 39*, 563–576.

Emrouznejad, A., Parker, B. R., & Tavares, G. (2008). Evaluation of research in efficiency and productivity: A survey and analysis of the first 30 years of scholarly literature in DEA. *Socio-Economic Planning Sciences, 42*, 151–157.

Färe, R., Grosskopf, S., & Lovell, C. A. K. (1994). *Production frontiers*. Cambridge, UK: Cambridge University Press.

Färe, R., Grosskopf, S., & Lovell, C. A. K. (1995). *The measurement of efficiency of production*. Norwell, MA: Kluwer.

Farrell, M. J. (1957). The measurement of productive efficiency. *Journal of Royal Statistical Society*, Series A, 253–290.

Ferrier, G. D., & Lovell, C. A. K. (1990). Measuring cost efficiency in banking: Econometric and linear programming evidence. *Journal of Econometrics, 46*, 229–245.

Jagannathan, R. (1985). Use of sample information in stochastic recourse and chance constrained programming models. *Management Science, 31*, 96–108.

Kamakura, W. A. (1988). A note on the use of categorical variables in data envelopment analysis. *Management Science, 34*, 1273–1276.

Korostelev, A., Simar, L., & Tsybakov, A. (1995). Efficient estimation of monotone boundaries. *Annals Statistics, 23*, 476–489.

Land, K., Lovell, C. A. K., & Thore, S. (1994). Chance constrained data envelopment analysis. *Managerial and Decision Economics, 14*, 541–554.

Neralić, L., & Wendell, R. (2000). A generalized additive, categorical model in data envelopment analysis. *TOP: Journal of the Spanish Society of Statistics and Operations Research, 8*, 235–263.

Olesen, O. B., & Petersen, N. C. (1995). Chance constrained efficiency evaluation. *Management Science, 41*, 442–457.

Pastor, J. T. (1996). Translation invariance in data envelopment analysis. *Annals Operations Research, 66*, 93–102.

Rousseau, J. J., & Semple, J. H. (1993). Categorical outputs in data envelopment analysis. *Management Science, 39*, 384–386.

Schaffnit, C., Rosen, D., & Paradi, J. C. (1997). Best practice analysis of bank branches: An application of DEA in a large canadian bank. *European Journal of Operational Research, 98*, 269–289.

Seiford, L. M. (1994). A bibliography of data envelopment analysis. In A. Charnes, W. W. Cooper, A. Y. Lewin, & L. M. Seiford (Eds.), *Data envelopment analysis: Theory, methodology and applications*. Norwell, MA: Kluwer.

Seiford, L. M., & Thrall, R. M. (1990). Recent development in DEA. In A. Y. Lewin, & C. A. Knox Lovell (Eds.), *Frontier analysis, parametric and nonparametric approaches*. *Journal of Econometrics*, *46*, 7–38.

Simar, L., & Wilson, P. W. (1998). Sensitivity analysis of efficiency scores: How to bootstrap in nonparametric frontier models. *Management Science, 44*, 49–61.

Simon, H. A. (1957). *Models of man*. New York: John Wiley.

Thompson, R., Singleton, F., Thrall, R. M., & Smith, B. (1986). Comparative site evaluations for locating a high energy physics laboratory in Texas. *Inter-faces, 16*, 35–49.

Thrall, R. M. (1996). Duality, classification and slacks in DEA. *Annals Operations Research, 66*, 109–138.

Data Mining

Syam Menon[1] and Ramesh Sharda[2]
[1]The University of Texas at Dallas, Richardson, TX, USA
[2]Oklahoma State University, Stillwater, OK, USA

Introduction

When Wal-Mart installed their 24 terabyte data warehouse, it was among the largest in the world. Just a few years later, they were adding over a billion rows of data a day (Babcock 2006), and operating a 5 petabyte database (Lai 2008). An even more striking example is eBay, which started with a 14 terabyte database in 2002. It has since been adding over 40 terabytes of auction and purchase data every day into a data warehouse that is expected to exceed 20 petabytes by 2011. Clearly, as the cost of capturing data has decreased and easier-to-use data capture tools have become available, the volumes of data being accumulated have grown at a very rapid pace. Technological developments, with the evolution of the Internet playing a fundamental role, have enabled an increase in the volume of traditional data being recorded. Further, such developments have made possible the capture of information in far greater detail than ever before (based on barcodes or RFID, for example) and often of information that was not easily recordable before, such as eye or mouse movements.

What is Data Mining?

The availability of large data repositories has resulted in significant developments in the methodologies to analyze them, both in terms of the technology available for analysis, and in terms of its mainstream acceptance. From what was a relatively esoteric technology at the close of the 20th century, data mining – defined succinctly as "the science of extracting useful information from large data sets" (Hand et al. 2001) – has developed into a powerful set of tools indispensable to most organizations. In fact, it is gradually morphing into a key component of the merger of quantitative techniques into a new label called business analytics.

Many of the techniques used in data mining have their roots in traditional statistics, artificial intelligence, and machine learning. Developments in data mining techniques went hand-in-hand with developments in data warehousing and online analytical processing (OLAP). From the early 1990s when data mining started being viewed as a viable business solution, the cost of computing has dropped steadily, while processing power has increased. This made the benefits of data mining apparent, and triggered many companies to start using it regularly.

Commercial applications of data mining abound. A 2010 poll of data miners (conducted by KDNuggets) listed customer relations management, banking, healthcare, and fraud detection as the top four fields where data mining is applied. It is also commonly used in finance, direct marketing, insurance, and manufacturing. In fact, it has become common practice in almost every industry to discern new knowledge from data; only the extent of penetration varies across industries.

This is, of course, in addition to the vast quantities of data collected in the non-business world. It has found application in disciplines as varied as astronomy, genetics, healthcare, and education, just to name a few. The U.S. Department of Homeland Security applies data mining for a variety of purposes, including the comparison of "traveler, cargo, and conveyance information against intelligence and other enforcement data by incorporating risk-based targeting scenarios and assessments," and "to improve the collection, use, analysis, and dissemination of information that is gathered for the primary purpose of targeting, identifying, and preventing potential terrorists and terrorist weapons from entering the United States" (DHS 2009).

The availability of new types of data has opened up additional opportunities for selective extraction of useful information. Data originating from the Web can be mined based on content, network structure, or usage (e.g., when was a page used and by whom). There has been considerable interest in the mining of text from a variety of perspectives – to filter e-mail, to gain intelligence about competitors, to analyze the opinions of movie viewers to better understand movie reviews, as well as the mining of social network data both in terms of user behaviors and networks, including text mining of comments. The analysis of audio and video files is another difficult but promising

avenue for data mining. Speech recognition technologies have improved significantly. But, audio mining goes much further by providing users the ability to search and index the digitized audio content in a variety of contexts like news and webcasts, recorded telephone conversations, office meetings, and archives in libraries and museums.

How Does Data Mining Work?

Most of the general ideas applicable to modeling of any kind hold true for data mining as well. To work effectively, data mining requires clearly stated objectives and evaluation criteria. The process (often referred to as the Knowledge Discovery in Databases – or KDD – process) entails various critical steps. All data need to be cleaned to eliminate noise and correct errors. As data usually come from multiple, heterogeneous sources, there has to be a logical process of data integration. Once an objective has been identified for analysis, all appropriate data needs to be retrieved from the storage warehouse(s). If necessary, extracted data may need to be transformed into a form amenable for mining. Once all these preprocessing steps are completed, relevant data mining techniques can be applied. As with any analysis technique, the output from the mining process usually needs to be interpreted by the analyst after imposing as much domain knowledge as possible to intelligently glean useful information. Any model that is built should be tested and validated before putting to full use. Additionally, the KDD process has to be iterative for it to be beneficial. The knowledge discovered through mining can be used to obtain feedback from the user which in turn can be used to improve the mining process.

Data mining tasks fall into two main groups – descriptive tasks that characterize properties of the data being analyzed, and predictive tasks which make predictions about new data points based on inferences made from existing data. Data mining algorithms traditionally fall into one of three categories — classification and prediction, clustering, and association discovery. Other functionalities like data characterization and outlier analysis are also common, as are applications that form key components of recommender systems. Data visualization plays an important role in many of these techniques by guiding the users in the right direction. Some of these techniques are described briefly below.

Classification. Classification, or supervised induction, is perhaps the most common of all data mining activities. The objective of classification is to analyze the historical data stored in a database and to automatically generate a model that can predict future behavior. This induced model consists of generalizations over the records of a training data set, which help distinguish predefined classes. The hope is that this model can then be used to predict the classes of other unclassified records. When the output variable of interest is categorical, the models are referred to as classifiers, while models where the output variable is numerical are called prediction models.

Tools commonly used for classification include neural networks, decision trees, and if-then-else rules that need not have a tree structure. Statistical tools like logistic regression are also commonly used. Neural networks involve the development of mathematical structures with the ability to learn. They tend to be most effective where the number of variables involved is large and the relationships between them too complex and imprecise. It can easily be implemented in a parallel environment, with each node of the network doing its calculations on a different processor. There are disadvantages as well. It is usually very difficult to provide a good rationale for the predictions made by a neural network. Also, training time on neural networks tends to be considerable. Further, the time needed for training tends to increase as the volume of data increases, and in general, such training cannot be done on very large databases. These and other factors have limited the acceptability of neural networks for data mining.

Decision trees (DTs) classify data into a finite number of classes, based on the values of the variables. DTs are comprised of essentially a hierarchy of if-then statements and are thus significantly faster than neural nets. Logistic regression models are used for binary classification, with multinomial logistic models being used if there are more than two output categories.

Clustering. Most clustering algorithms partition the records of a database into segments where members of a segment share similar qualities. In fact, clustering is sometimes referred to as unsupervised classification. Unlike in classification, however, the clusters are unknown when the algorithm starts. Consequently,

before the results of clustering techniques are put to actual use, it might be necessary for an expert to interpret and potentially modify the suggested clusters. Once reasonable clusters have been identified, they could be used to classify new data. Not surprisingly, clustering techniques include optimization; we want to create groups, which have maximum similarity among members within each group and minimum similarity among members across the groups. Another common application is market basket analysis.

Association Discovery. A special case of association rule mining looks at sequences in the data. Sequence discovery has many applications, and is a significant sub-field in itself. It can be to conduct temporal analysis to identify customer behavior over time, to identify interesting genetic sequences, for website re-design, and even for intrusion detection.

Visualization. The insights to be gained from visualizing the data cannot be over-emphasized. This holds true for most data analysis techniques, but is of special relevance to data mining. Given the sheer volume of data in the databases being considered, visualization in general is a difficult endeavor. It can be used, however, in conjunction with data mining to gain a clearer understanding of many underlying relationships.

Recommender Systems. Many companies claim that a substantial portion of their revenues are a result of effective recommendations. Among the better known examples are Amazon.com, which was one of the earlier proponents of recommender systems, and Netflix, which claims that "roughly two-thirds of the films rented were recommended to subscribers by the site" (Flynn 2006). The impact and importance of a well implemented recommendation system is exemplified by the fact that Netflix offered a million-dollar prize for anyone who could improve their recommendation accuracy by at least 10%. A variety of techniques exist for making recommendations, with user and item based collaborative filtering being the most common.

Other Relevant Aspects

Software. There are many large vendors of data mining software. Some of the key commercial packages include SAS Enterprise Miner, IBM SPSS Modeler (Formerly SPSS Clementine), Oracle, DigiMine, Microsoft SQL Server, SAP Business Objects. Weka is a well reputed freeware out of The University of Waikato in New Zealand. Another open source data mining software is Rapid Miner.

Privacy. Data mining has been restricted in its impact due to privacy concerns. In particular, in privacy concerns when applying data mining to healthcare data. A contested court case concerns the mining of physicians' prescription history to increase drug sales; some states are trying to limit access to this information (Field 2010). The fundamental issue underlying these concerns relate to the intent behind data collection. For example, while consumers explicitly agree to the use of data collected for bill payment for that specific purpose, they may not know or want to agree to the use of their data for mining – that would go beyond the original intent for which the data were acquired.

Another area of data mining privacy concerns counterterrorist information Claburn (2008). A report dealing with the balance between privacy and security by the National Research Council recommends that the U.S. government rethink its approach to counterterrorism in light of the privacy risks posed by data mining.

Although some work has been done to incorporate privacy concerns explicitly into the mining process, this is still a developing field. In all likelihood, the matter of privacy in the context of data mining will be an issue for some time. A simple solution is unlikely. These issues will probably be resolved only through a blend of legislation and additional research into privacy preserving data mining.

The Role of Operations Research

Data mining algorithms are a heterogeneous group, loosely tied together by the common goal of generating better information. Operations research is concerned with making the best use of available information. By selecting the appropriate definition of information, operations research has been playing a significant role on both sides of the data mining engine. Formulations for clustering and classification were introduced in the 1960s and 70s (Ólafsson 2006). Nonlinear programming solution techniques have been adapted for faster training in neural network applications. Scalability, the ability to deal with

large amounts of data, is a difficult and important issue in data mining, one in which OR could play a significant role.

The lack of reliable data (or of the data itself) is a common problem faced by operations researchers trying to get a good model to work in the real world. This problem becomes more acute when data needs to be deciphered from terabytes of stored information. Data mining tools make accessing and processing the data easier and may provide more reliable data to the OR modeler. There are opportunities for operations research to be applied at a more fundamental level as well. Ultimately, as with any analysis tool, the outputs of data mining models are only as good as the inferences the analyst can make from them. OR techniques can be of assistance in making the best use of the outputs obtained. For example, research has been conducted to improve recommendations by combining information from multiple association rules, and to provide the best set of recommendations to maximize the likelihood of purchase. Similarly, combining information on prior purchase histories and revenue optimization models enables a new blend of practical business decision making. As noted, this integration of data mining and optimization has been labeled business analytics. IBM and other major vendors are developing new business groups focused on analytics that arise from combinations of organizations in optimization and data mining (Turban et al. 2010, pp. 78).

Concluding Remarks

By detecting patterns hitherto unknown, data mining techniques could suggest new modes to pursue old objectives. They could even allow the formulation of better, more sophisticated models in the wake of new information. In general, the gains to be made from exploiting newly discovered information are significantly higher than the marginal improvements that can be made by improving existing solution procedures. As the volume and types of data being collected increase, so will the need for better tools to analyze the data. Consequently, the future of data mining seems to be full of possibilities. The enthusiasm for discovering new information, however, needs to be tempered with the need to address privacy concerns, as not doing so could have long term repercussions on the parties involved.

See

- ▶ Artificial Intelligence
- ▶ Cluster Analysis
- ▶ Computer Science and Operations Research Interfaces
- ▶ Decision Trees
- ▶ Neural Networks
- ▶ Nonlinear Programming
- ▶ Visualization

References

Babcock, C. (2006, January 9). Data, data, everywhere. *InformationWeek*.

Claburn, T. (2008, October 7). Counterterrorist data mining needs privacy protection. *InformationWeek*.

Department of Homeland Security. (2009, December). *DHS Privacy Office: 2009 data mining report to Congress*.

Field, A. (2010, April 3). Legal briefing: Will drugmakers' prescription data mining be undermined? *Daily Finance*.

Flynn, L. (2006, January 23). Like this? You'll hate that. (not all web recommendations are welcome.). *The New York Times*.

Hand, D., Mannila, M., & Smyth, P. (2001). *Principles of data mining*. Cambridge, MA: MIT Press.

Lai, E. (2008, October 14). Teradata creates elite club for petabyte-plus data warehouse customers. *Computerworld*.

Ólafsson, S. (2006, November 2006). Editorial: introduction to operations research and data mining. *Computers and Operations Research, 33*(11).

Turban, E., Sharda, R., & Delen, D. (2010). *Decision support and business intelligence systems* (9th ed.). Upper Saddle River, NJ: Pearson Prentice Hall.

Data Warehousing

Paul Gray
Claremont Graduate University, Claremont, CA, USA

Introduction

The data warehouse is one of the key information infrastructure resources for Operations Researchers. Its difference from the conventional transactional database, which is used to keep track of individual events, is shown in Table 1.

The typical transaction database contains details about individual transactions such as the purchase of merchandise or individual invoices sent or paid.

Data Warehousing, Table 1 Data warehouse vs. transaction database

Data Warehouse	Subject oriented	Integrated	Time-variant	Non-volatile
Transaction Database	Transaction oriented	Un-integrated	Current status	Changes as trans- actions occur

Transactional databases are concerned with operations while data warehouses are organized by subject. For example, operational data in a bank focuses on transactions involving loans, savings, credit cards, and trust accounts, while the data warehouse is organized around customer, vendor, product, and activity history.

The continually changing transactional data is not in the form needed for planning, managing, and analyzing. That is where the data warehouse comes in.

The classic data warehouse is defined as "a subject oriented, integrated, non-volatile, time variant, collection of data to support management's decisions" (Inmon 1992, p. 29).

The characteristics of the data warehouse that were summarized in Table 1 are given in more detail in Table 2.

In addition, the characteristics of the data itself are different, as shown in Table 3.

Data warehouses are really databases that provide both aggregated and detailed data for decision making. They are usually physically separated from both the organization's transaction databases and its operational systems.

Note that data normalization, which is used in transactional databases, makes sure that an individual data point appears once and only once. Normalization is not required conceptually in data warehouses. Some data warehouse designs, however, do normalize their data.

Data Warehousing, Table 2 Data warehouse characteristics

Subject orientation	Data are organized by how users refer to it, not by client
Data Integration	Data are organized around a common identifier, consistent names, and the same values throughout. Inconsistencies are removed.
Time	Data provide time series and focus on history, rather than current status.
Non-volatile	Data can be changed only by the upload process, not by the user.

Data Warehousing, Table 3 Characteristics of data in the warehouse

Summarized	In addition to current operational data when needed, data summaries used for decision making are also stored.
Larger database	Time series implies much more data is included.
Not normalized	Data can be redundant.
Metadata	Includes data about how the data is organized and what it means.
Sources of input data	Data comes from operational systems

Flow of Data

The flow of data into and out of the data warehouse follows these steps:

1. Obtain inputs
2. Clean inputs
3. Store in the warehouse
4. Provide output for analysis

Inputs to the data warehouse are the first step in what is called the extract, transform, and load process (ETL). Data sources, often from what are called legacy systems, push data to the warehouse rather than the warehouse pulling data from the sources. The sources send updates to the data warehouses at pre-specified intervals. This operation is performed on a fixed schedule where the interval between updates can range from nearly real time to once a day or longer, depending on the source.

Each source may have its own convention for what to call things and may even use different names and/or different metrics. For example, different transactional databases may store gender as (m, f), (1, 0), (x, y), (male, female) or may have different names for the same person (e.g., S. Smith, Sam Smith, and S. E. Smith). To overcome inconsistencies and to make

sure that users see only one version of the truth, data cleansing is performed by the warehouse on the input.

Data cleansing involves changing the input data so that it meets the warehouse's standards. Specialized software (usually referred to as ETL) makes the input data extracted from the sources consistent (e.g., in format, scaling, and naming) with the way data is stored in the warehouse. For example, the warehouse standardizes on one of the formats for gender and translates all other versions to the standard. Transformation uses metadata (i.e., data about the data) to accomplish this. The data are loaded (i.e., stored) in the warehouse only after they are cleansed. The goal is to establish a single value of the truth within the warehouse.

The data warehouse is used for analytics and routine reporting. Both create information useful to managers and professionals. Analytics refers to using models and performing computations on the data. Routine reporting refers to creating, documents, tables, and graphics, usually on a repetitive schedule. Routine outputs include dashboards (which mostly present status), scorecards (which show how well goals are being met), and alerts (which notify managers when current values are outside prescribed limits).

What is in the Data Warehouse

The data warehouse contains not only the current detail data that was transferred from the legacy systems, but also lightly summarized or highly summarized data, as well as old detail data. Metadata are usually also stored in the data warehouse.

The current detail data reflects the most recent happenings and is usually stored on disk. Detail data is voluminous and is stored at higher levels of granularity. Granularity refers to the level of detail provided in the data warehouse. The more detail provided, the higher the level of granularity. The highest level is transaction data such as is required for data mining. For decision support, analysis, and planning, the level of granularity can be much lower. Granularity is an important trade-off because the higher the level of granularity, the more data must be stored, the greater the level of detail available, and the more computing needs to be done, even for problems that do not use that level of granularity. For example, if a gasoline company records every motorist's stop at its stations, it can use the credit transaction to understand its customers detailed buying patterns. For total sales by station, that level of granularity is not needed.

Lightly summarized data is generally used at the analyst level, whereas highly summarized data (which is compact and easily accessible) is used by senior managers. The choice of summarization level involves tradeoffs because the more highly summarized the data, the more the data is actually accessed and used, the quicker it is to retrieve, but the less detail is available for understanding it. One way to speed query response time is to pre-calculate aggregates which are referred to often, such as annual sales data.

To keep storage requirements within reason, older data are moved to lower cost storage with much slower data retrieval. An aging process within the data warehouse is used to decide when to move data to mass storage.

Metadata contains two types of information:

1. What the user needs to know to be able to access the data in the warehouse. It tells the user what is stored in the warehouse and where to find it.
2. What information systems personnel need to know about how data is mapped from operational form to warehouse form, i.e., what transformations occurred during input and the rules used for summarization.

Metadata keeps track of changes made converting, filtering, and summarizing data, as well as changes made in the warehouse over time, e.g., data added, data no longer collected, and format changes.

Warehouse Data Retrieval and Analysis

The data stored in the data warehouse are optimized for speedy retrieval through on-line analytical processing (OLAP). The retrieval methods depend on the data format. The three most common are:

- Relational OLAP (ROLAP), which works with relational databases
- Multidimensional OLAP (MOLAP) for data stored in multi-dimensional arrays
- Hybrid OLAP (HOLAP) which works with both relational and multidimensional databases.

OLAP involves answering multidimensional questions such as the number of units of Product

A sold in California at a discount to resellers in November (i.e., product, state, terms of sale, customer class, time).

To enable relational databases (that store data in two dimensions) to deal with multidimensionality, two types of tables are introduced: fact tables that contain numerical facts, or dimension tables that contain pointers to the fact tables and show where the information can be found. A separate dimension table is provided for each dimension (e.g., market, product, time). Fact tables tend to be long and thin and the dimension tables tend to be small, short, and wide. Because a single fact table is pointed to by several dimension tables, the visualization of this arrangement looks like a star and hence is called a star schema. A variant, used when the number of dimensions is large and multiple fact tables share some of the same dimension tables, is called a snowflake schema.

Multidimensionality allows analysts to slice and dice the data, i.e., to systematically reduce a body of data into smaller parts or views that yield more information. Slice and dice is also used to refer to the presentation of warehouse information in a variety of different and useful ways.

Why a Separate Warehouse?

A fundamental tenet of data warehouses is that their data are separate from operational data. The reasons for this separation are:

Performance. Requests for data for analysis are not uniform. At some times, for example, when a proposal is being written or a new product is being considered, huge amounts of data are required. At other times, the demand may be small. The demand peaks create havoc with conventional on-line transaction systems because they slow them down considerably, keeping users (and often customers) waiting.

Data Access. Analysis requires data from multiple sources. These sources are captured and integrated by the warehouse.

Data Formats. The data warehouse contains summary and time-based data as well as transaction data. Because the data are integrated, the information in the warehouse is kept in a single, standard format.

Data Quality. The data cleansing process of ETL creates a single version of the truth.

Other Forms of Data Warehouses

As organizations found new ways of using the warehouse, they created specialized forms for specific uses. Among these are:

- Data marts
- Operational data stores
- Real-time warehouses
- Data warehouse appliances
- Data warehouses in the cloud
- Separate data warehouses for casual and power users

Data marts are a small-scale version of a data warehouse that include all the characteristics of an enterprise data warehouse, but are much smaller in size and cost. Data marts can be independent or dependent.

- Independent data marts are typically stand-alone units used by departments or small strategic business units that often support only specific subject areas. A data mart is appropriate if it is the only data warehouse for a small or medium sized firm. Multiple independent data marts become a problem rather than a solution if they differ from department to department. Integrating them so that there is only a single value of the truth throughout the organization is difficult, particularly if a comprehensive data warehouse is later attempted.
- Dependent data marts, such as those used by analytics groups, contain a subset of the warehouse data needed by a particular set of users. To maintain a single value of the truth, care is taken that the dependent data mart does not change the data from the warehouse.

An Operational Data Store (ODS) is a data warehouse for transaction data. It is a form of data warehouse for operational use. The ODS is used where some decisions need to be made in near real-time and require the characteristics of a warehouse (e.g., clean data). The ODS is subject oriented and integrated like the warehouse but, unlike the data warehouse, information in an ODS can be changed and updated rather than retained forever. Thus, an ODS contains current and near-current information, but not much historical data.

When data moves from legacy systems to the ODS, the data are re-created in the same form as in the warehouse. Thus, the ODS converts data, selects among sources, may contain simple summaries of the

current situation for management use, alters the key structures and the physical structure of the data, as well as its internal representation. Loading data into a data warehouse from an ODS is easier than loading from individual legacy systems, because most of the work on the data has been performed. It contains much less data than a data warehouse but also includes some that is not stored in the data warehouse. The ODS is usually loaded more frequently by data sources than the warehouse to keep it much more current. For example, the Walmart ODS receives information every 15 minutes.

The real-time data warehouse is used to support ongoing analysis and actions. A form of operational data store, real time data warehouses are closely tied to operational systems. They hold detailed, current data and try to use even shorter times between successive loadings than operational data stores. With these data warehouses, enterprises can respond to customer interactions and changing conditions in real time. For example, credit card companies use it to detect and stop fraud as it happens, a transportation company uses it to reroute its vehicles, and online retailers use it to communicate special offers based on a customer's Web surfing or mobile phone behavior. The real-time data warehouse is an integral part of both short-term (tactical) and long-term (strategic) decisions.

The real-time data warehouse changes the decision support paradigm, which has long been associated with strategic decision making. It supplies support for operational decision making such as customer-facing (direct interactions or communications with customers) and supply chain applications.

A data warehouse appliance is similar in concept to an all-in-one PC, i.e., it integrates the physical components of a data warehouse (servers, storage, operating system) with a database management system and software optimized for the data warehouse. These low-cost appliances are designed to provide terabyte to petabyte capacity warehouses.

Cloud computing refers to using the networked, on-demand, shared resources available through the Internet for virtual computing. Typically, rather than each firm owning its own warehouse, a third-party vendor provides a centralized service to multiple clients based on hardware and software usage. Although, as of 2010 - no data warehouse in the cloud exists, some inferences can be drawn. Agosta (2008) argues that in cloud computing the data in a warehouse will have to be location independent and transparent rather than being a centralized, non-volatile repository. Furthermore, the focus will be on distributed data marts and analytics rather than large data stores because of the problems and costs in moving the huge amounts of data in a warehouse to the cloud.

Data warehouses attract two types of users (Eckerson 2010):

- Casual users. These users are executives and other knowledge workers who consume information but do not usually create it. Their use is mostly static. They check dashboards, monitor regular reports, respond to alerts, and only occasionally dig deeper into the warehouse to create ad hoc reports.
- Power users. These users explore the data and build models. Conventional reports are insufficient for their needs. They model data in unique ways and supplement warehouse contents with data obtained from other sources.

In most organizations, the conventional data warehouse is used by both types of users despite their different needs. Some organizations, however, are moving to separate warehouses, one for each type of user. The conventional data warehouse feeds its data to the one for the power users, so that there is still only one version of the truth. In these organizations, conventional data warehouses continue to serve casual users whose requirements are mostly static. The idea is that performance gains are achieved by creating a separate warehouse customized to power users. Over the years, the special warehouses for power users have operated under a variety of names such as exploration data warehouse (for number crunching) (Inmon 1998), prototype data warehouse (for new approaches to warehouse design), and data warehouse sandbox. Eckerson (2010) describes to three types of sandbox architectures for analytics: physical, virtual, and desktop.

The physical sandbox is built around a data warehouse appliance or a specialized database with rapid access (e.g., columnar or massively parallel processing) that contains a copy of the data in the warehouse. Complex queries from the data warehouse are offloaded and used, together with data not stored in the warehouse. The result is that runaway queries (so large that they overload the warehouse) do

not slow the warehouse and analysts can safely and easily explore large amounts of data.

The virtual sandbox is created inside the warehouse by using workload management utilities. Again, data can be added to that available in the warehouse. The advantage is that warehouse data does not need to be replicated. The disadvantage is that care must be taken to keep processing for casual and power users separate.

In desktop sandboxes, analysts are provided with powerful in-memory desktop databases that can be downloaded from the warehouse. Analysts gain local control and fast performance but much less data scalability than in physical or virtual sandboxes.

Applications

Data warehousing is central to data mining and business intelligence. Other applications include:

- Customer churn prediction
- Decision support
- Financial forecasting
- Insurance fraud analysis
- Logistics and inventory management
- Trend analysis

See

- ▶ Business Intelligence
- ▶ Data Mining
- ▶ Decision Support Systems (DSS)
- ▶ Information Systems and Database Design in OR/MS
- ▶ Visualization

References

Agosta, L. (2008). Data warehousing in the clouds: Making sense of the cloud computing market. *Beye Network*, 9 October 2008.

Eckerson, W. W. (2010). Dual BI architectures: The time has come. *The Data Warehousing Institute*, 18 Nov 2010.

Gray, P., & Watson, H. J. (1998). *Decision support in the data warehouse*. Upper Saddle River, NJ: Prentice-Hall.

Inmon, W. H. (1992). *Building the data warehouse*. New York: Wiley.

Inmon, W. H. (1998). *The exploration warehouse*. DM Review, June 1998.

Inmon, W. H. (2005). *Building the data warehouse* (4th ed.). Indianapolis, IN: Wiley.

Kimball, R., et al. (2009). *Kimball's data warehouse toolkit classics: The data warehouse toolkit, 2nd Edn; The data warehouse lifecycle, 2nd Edn; The data warehouse ETL toolkit*. New York: Wiley.

Sprague, R. H., & Carlson, E. D. (1982). *Building effective decision support systems*. Englewood Cliffs, NJ: Prentice Hall.

Database Design

▶ Information Systems and Database Design in OR/MS

DEA

▶ Data Envelopment Analysis

Decision Analysis

David A. Schum
George Mason University, Fairfax, VA, USA

Introduction

The term decision analysis identifies a collection of technologies for assisting individuals and organizations in the performance of difficult inferences and decisions. Probabilistic inference is a natural element of any choice made in the face of uncertainty. No single discipline can lay claim to all advancements made in support of these technologies. Operations research, probability theory, statistics, economics, psychology, artificial intelligence, and other disciplines have contributed valuable ideas now being exploited in various ways by individuals in many governmental, industrial, and military organizations. As the term decision analysis suggests, complex inference and choice tasks are decomposed into smaller and presumably more manageable elements, some of which are probabilistic and others preferential or value-related. The basic strategy employed in

decision analysis is divide and conquer. The presumption is that individuals or groups find it more difficult to make holistic or global judgments required in undecomposed inferences and decisions than to make specific judgments about identified elements of these tasks. In many cases we may easily suppose that decision makers are quite unaware of all of the ingredients that can be identified in the choices they face. Indeed, one reason why a choice may be perceived as difficult is that the person or group charged with making this choice may be quite uncertain about the kind and number of judgments this choice entails. One major task in decision analysis is to identify what are believed to be the necessary ingredients of particular decision tasks.

The label decision analysis does not in fact provide a complete description of the activities of persons who employ various methods for assisting others in the performance of inference and choice tasks. This term suggests that the only thing accomplished is the decomposition of an inference or a choice into smaller elements requiring specific judgments or information. It is, of course, necessary to have some process by which these elements can be reassembled or aggregated so that a conclusion or a choice can be made. In other words, we require some method of synthesis of the decomposed elements of inference and choice. A more precise term for describing the emerging technologies for assistance in inference and choice would be the term decision analysis and synthesis. This fact has been noted in an account of progress in the field of decision analysis (Watson and Buede 1987). As it happens, the same formal methods that suggest how to decompose an inference or choice into more specific elements can also suggest how to reassemble these elements in drawing a conclusion or selecting an action.

Processes and Stages of Decision Analysis

Human inference and choice are very rich intellectual activities that resist easy categorization. Human inferences made in natural settings (as opposed to contrived classroom examples) involve various mixtures of the three forms of reasoning that have been identified: (1) deduction (showing that some conclusion is necessary), (2) induction (showing that some conclusion is probable), and (3) abduction (showing that something is possibly or plausibly true). There are many varieties of choice situations that can be discerned. Some involve the selection of an action or option such as where to locate a nuclear power plant or a toxic waste disposal site. Quite often one choice immediately entails the need for another and so we must consider entire sequences of decisions. It is frequently difficult to specify when a decision task actually terminates. Other decisions involve determining how limited resources may best be allocated among various demands for these resources. Some human choice situations involve episodes of bargaining or negotiation in which there are individuals or groups in some competitive or adversarial posture. Given the richness of inference and choice, analytic and synthetic methods differ from one situation to another as observed in several surveys of the field of decision analysis (von Winterfeldt and Edwards 1986; Watson and Buede 1987; Clemen 1991; Shanteau et al. 1999). Some general decision analytic processes can, however, be identified.

Most decision analyses begin with careful attempts to define and structure an inference and/or decision problem. This will typically involve consideration of the nature of the decision problem and the individual or group objectives to be served by the required decision(s). A thorough assessment of objectives is required since it is not possible to assist a person or group in making a wise choice in the absence of information about what objectives are to be served. It has been argued that the two central problems in decision analysis concern uncertainty and multiple conflicting objectives (von Winterfeldt and Edwards 1986, pp. 4–6). A major complication arises when, as usually observed, a person or a group will assert objectives that are in conflict. Decisions in many situations involve multiple stakeholders and it is natural to expect that their stated objectives will often be in conflict. Conflicting objectives signal the need for various tradeoffs that can be identified. Problem structuring also involves the generation of options, actions, or possible choices. Assuming that there is some element of uncertainty, it is also necessary to generate hypotheses representing relevant alternative states of the world that act to produce possibly different consequences of each option being considered. The result is that when an action is selected we are not certain about which consequence or outcome will occur.

Another important structuring task involves the identification of decision consequences and their attributes. The attributes of a consequence are measurable characteristics of a consequence that are related to a decision maker's asserted objectives. Identified attributes of a consequence allow us to express how well a consequence measures up to the objectives asserted in some decision task. Stated in other words, attributes form value dimensions in terms of which the relative preferability of consequences can be assessed. There are various procedures for generating attributes of consequences from stated objectives (e.g., Keeney and Raiffa 1976, pp. 31–65). Particularly challenging are situations in which we have multiattribute or vector consequences. Any conflict involving objectives is reflected in conflicts among attributes and signals the need for examining possible tradeoffs. Suppose, for some action A_i and hypothesis H_j, vector consequence $\boldsymbol{C}\mathrm{v}_{ij}$ has attributes $\{A_1, A_2, \ldots, A_r, \ldots, A_s, \ldots, A_t\}$. The decision maker may have to judge how much of A_r to give up in order to get more of A_s; various procedures facilitate such judgments. Additional structuring is necessary regarding the inferential element of choice under uncertainty. Given some exhaustive set of mutually exclusive hypotheses or action-relevant states of the world, the decision maker will ordinarily use any evidence that can be discovered that is relevant in determining how probable are each of these hypotheses at the time a choice is required. No evidence comes with already-established relevance, credibility, and inferential force credentials, these credentials have to be established by argument. The structuring of complex probabilistic arguments is a task that has received considerable attention (e.g., see Pearl 1988; Neapolitan 1990; Schum 1990, 1994).

At the structural stage just discussed, the process of decomposing a decision is initiated. On some occasions such decomposition proceeds according to formal theories of probability and value taken to be normative. It may even happen that the decision of interest can be represented in terms of some existing mathematical programming or other formal technique common in operations research. In some cases the construction of a model for a decision problem proceeds in an iterative fashion until the decision maker is satisfied that all ingredients necessary for a decision have been identified. When no new problem ingredients can be identified the model that results is said to be a requisite model (Phillips 1982, 1984). During the process of decomposing the probability and value dimensions of a decision problem it may easily happen that the number of identified elements quickly outstrips a decision maker's time and inclination to provide judgments or other information regarding each of these elements. The question is: how far should the process of divide and conquer be carried out? In situations in which there is not unlimited time to identify all conceivable elements of a decision problem, simpler or approximate decompositions at coarser levels of granularity have to be adopted.

In most decision analyses there is a need for a variety of subjective judgments on the part of persons involved in the decision whose knowledge and experience entitles them to make such judgments. Some judgments concern probabilities and some concern the value of consequences in terms of identified attributes. Other judgments may involve assessment of the relative importance of consequence attributes. The study of methods for obtaining dependable quantitative judgments from people represents one of the most important contributions of psychology to decision analysis (for a survey of these judgmental contributions, see von Winterfeldt and Edwards 1986). After a decision has been structured and subjective ingredients elicited, the synthetic process in decision analysis is then exercised in order to identify the best conclusion and/or choice. In many cases such synthesis is accomplished by an algorithmic process taken as appropriate to the situation at hand. Modern computer facilities allow decision makers to use these algorithms to test the consequences of various possible patterns of their subjective beliefs by means of sensitivity analyses. The means for defending the wisdom of conclusions or choices made by such algorithmic methods re-quires consideration of the formal tools used for decision analysis and synthesis.

Theories of Analysis and Synthesis

Two major pillars upon which most of modern decision analysis rests are theories of probabilistic reasoning and theories of value or preference. A very informative summary of the roots of decision theory has been provided by Fishburn (1999). It is safe to say that the conventional view of probability, in which Bayes' rule appears as a canon for coherent or

rational probabilistic inference, dominates current decision analysis. For some body of evidence $\boldsymbol{E}$v, Bayes' rule is employed in determining a distribution of posterior probabilities $P(H_k|\boldsymbol{E}\text{v})$, for each hypothesis H_k in an exhaustive collection of mutually exclusive decision-relevant hypotheses. The ingredients Bayes' rule requires, prior probabilities (or prior odds) and likelihoods (or likelihood ratios), are in most cases assumed to be assessed subjectively by knowledgeable persons. In some situations, however, appropriate relative frequencies may be available. The subjectivist view of probability, stemming from the work of Ramsey and de Finetti, has had a very sympathetic hearing in decision analysis (see Mellor 1990, and de Finetti 1972, for collections of the works of Ramsey and de Finetti).

Theories of coherent or rational expression of values or preferences stem from the work of von Neumann and Morgenstern (1947). In this work appears the first attempt to put the task of stating preferences on an axiomatic footing. Adherence to the von Neumann and Morgenstern axioms places judgments of value on a cardinal or equal-interval scale and are often then called judgments of utility. These axioms also suggest methods for eliciting utility judgments and they imply that a coherent synthesis of utilities and probabilities in reaching a decision consists of applying the principle of expected utility maximization. This idea was extended in the later work of Savage (1954), who adopted the view that the requisite probabilities are subjective in nature. The canon for rational choice emerging from the work of Savage is that the decision maker should choose from among alternative actions by determining which one has the highest subjective expected utility (SEU). Required aggregation of probabilities is assumed to be performed according to Bayes' rule. In some works, this view of action-selection is called Bayesian decision theory (Winkler 1972; Smith 1988).

Early works by Edwards (1954, 1961) stimulated interest among psychologists in developing methods for probability and utility elicitation; these works also led to many behavioral assessments of the adequacy of SEU as a description of actual human choice mechanisms. In a later work, Edwards (1962) proposed the first system for providing computer assistance in the performance of complex probabilistic inference tasks. Interest in the very difficult problems associated with assessing the utility of multiattribute consequences stems from the work of Raiffa (1968). But credit for announcing the existence of the applied discipline now called decision analysis belongs to Howard (1966, 1968).

Decision Analytic Strategies

There are now many individuals and organizations employed in the business of decision analysis. The inference and decision problems they encounter are many and varied. A strategy successful in one context may not be so successful in another. In most decision-analytic encounters, an analyst plays the role of a facilitator, also termed high priests (von Winterfeldt and Edwards 1986, p. 573). The essential task for the facilitator is to draw out the experience and wisdom of decision makers while guiding the analytic process toward some form of synthesis. In spite of the diversity of decision contexts and decision analysts, Watson and Buede (1987, pp. 123–162) were able to identify the following five general decision analytic strategies in current use. They make no claim that these strategies are mutually exclusive.

1. Modeling. In some instances decision analysts will focus upon efforts to construct a conceptual model of the process underlying the decision problem at hand. In such a strategy, the decision maker(s) being served not only provide the probability and value ingredients their decision requires but are also asked to participate in constructing a model of the context in which this decision is embedded. In the process of constructing these often-complex models, important value and uncertainty variables are identified.
2. Introspection. In some decision analytic encounters, a role played by the facilitator is one of assisting decision makers in careful introspective efforts to determine relevant preference and probability assessments necessary for a synthesis in terms of subjective expected utility maximization. Such a process places great emphasis upon the reasonableness and consistency of the often large number of value and probability ingredients of action selection.
3. Rating. In some situations, especially those involving multiple stakeholders and multiattribute consequences, any full-scale task decomposition would be paralytic or, in any case, would not provide the timely decisions so often required.

In order to facilitate decision making under such circumstances, models involving simpler probability and value assessments are often introduced by the analyst. In some forms of decision analysis, many of the difficult multiattribute utility assessments are made simpler through the use of various rating techniques and by the assumption of independence of the attributes involved.

4. Conferencing. In a decision conference the role of the decision analyst as facilitator (or high priest) assumes special importance. In such encounters, often involving a group of persons participating to various degrees in a decision, the analyst promotes a structured dialogue and debate among participants in the generation of decision ingredients such as options, hypotheses and their probability, and consequences and their relative value. The analyst further assists in the process of synthesis of these ingredients in the choice of an action. The subject matter of a decision conference can involve action selection, resource allocation, or negotiation.
5. Developing. In some instances, the role of the decision analyst is to assist in the development of strategies for recurrent choices or resource allocations. These strategies will usually involve computer-based decision support systems or some other computer-assisted facility whose development is justified by the recurrent nature of the choices. The study and development of decision support systems has itself achieved the status of a discipline (Sage 1991). An active and exciting developmental effort concerns computer-implemented influence diagrams stemming from the work of Howard and Matheson (1981). Influence diagram systems can be used to structure and assist in the performance of inference and/or decision problems and have built-in algorithms necessary for the synthesis of probability and value ingredients (e.g., Shachter 1986; Shachter and Heckerman 1987; Breese and Heckerman 1999). Such systems are equally suitable for recurrent and nonrecurrent inference and choice tasks.

Controversies

As an applied discipline, decision analysis inherits any controversies associated with theories upon which it is based. There is now a substantial literature challenging the view that the canon for probabilistic inference is Bayes' rule (e.g., Cohen 1977, 1989; Shafer 1976). Regarding preference axioms, Shafer (1986) has argued that no normative theories of preference have in fact been established and that existing theories rest upon an incomplete set of assumptions about basic human judgmental capabilities. Others have argued that the probabilistic and value-related ingredients required in Bayesian decision theory often reflect a degree of precision that cannot be taken seriously given the imprecise or fuzzy nature of the evidence and other information upon which such judgments are based (Watson et al. 1979). Philosophers have recently been critical of contemporary decision analysis. Agreeing with Cohen and Shafer, Tocher (1977) argued against the presumed normative status of Bayes' rule. Rescher (1988) argued that decision analysis can easily show people how to decide in ways that are entirely consistent with objectives that turn out not to be in their best interests. Keeney's work (1992) took some of the sting out of this criticism. Others (e.g., Dreyfus 1984) question whether or not decomposed inference and choice is always to be preferred over holistic inference and choice; this same concern is reflected in other contexts such as law (Twining 1990, pp. 238–242). So, the probabilistic and value-related bases of modern decision analysis involve matters about which there will be continuing dialogue and, perhaps, no final resolution. This acknowledged, decision makers in many contexts continue to employ the emerging technologies of decision analysis and find, in the process, that very complex inferences and choices can be made tractable and far less intimidating.

See

- ▶ Choice Theory
- ▶ Decision Analysis in Practice
- ▶ Decision Making and Decision Analysis
- ▶ Decision Support Systems (DSS)
- ▶ Decision Trees
- ▶ Group Decision Making
- ▶ Influence Diagrams
- ▶ Multi-attribute Utility Theory
- ▶ Utility Theory

References

Breese, J., & Heckerman, D. (1999). Decision-theoretic troubleshooting: A framework for repair and experiment. In J. Shanteau, B. Mellers, & D. Schum (Eds.), *Decision science and technology: Reflections on the contributions of ward Edwards* (pp. 271–287). Boston: Kluwer Academic.

Clemen, R. T. (1991). *Making hard decisions: An introduction to decision analysis*. Boston: PWS-Kent.

Cohen, L. J. (1977). *The probable and the provable*. Oxford: Clarendon Press.

Cohen, L. J. (1989). *An introduction to the philosophy of induction and probability*. Oxford: Clarendon Press.

De Finetti, B. (1972). *Probability, induction, and statistics: The art of guessing*. New York: Wiley.

Dreyfus, S. (1984). The risks ! and benefits ? Of risk-benefit analysis. *Omega, 12*, 335–340.

Edwards, W. (1954). The theory of decision making. *Psychological Bulletin, 41*, 380–417.

Edwards, W. (1961). Behavioral decision theory. *Annual Review of Psychology, 12*, 473–498.

Edwards, W. (1962). Dynamic decision theory and probabilistic information processing. *Human Factors, 4*, 59–73.

Fishburn, P. (1999). The making of decision theory. In J. Shanteau, B. Mellers, & D. Schum (Eds.), *Decision science and technology: Reflections on the contributions of Ward Edwards* (pp. 369–388). Boston: Kluwer Academic.

Hammond, J., Keeney, R., & Raiffa, H. (2002). *Smart choices: A practical guide to making better decisions*. New York: Random House.

Howard, R. (1966). Decision analysis: Applied decision theory. In D. B. Hertz & J. Melese (Eds.), *Proceedings fourth international conference on operational research*. New York: Wiley-Interscience.

Howard, R. (1968). The foundations of decision analysis. *IEEE Transactions on Systems Science and Cybernetics, SSC-4*, 211–219.

Howard, R., & Matheson, J. (1981). Influence diagrams. In R. Howard & J. Matheson (Eds.), *The principles and applications of decision analysis* (Vol. 2). Menlo Park, CA: Strategic Decisions Group, 1984.

Keeney, R. (1992). *Value-focused thinking*. Cambridge, MA: Harvard University Press.

Keeney, R., & Raiffa, H. (1976). *Decision with multiple objectives: Preferences and value tradeoffs*. New York: Wiley.

Mellor, D. H. (1990). *F.P. Ramsey: Philosophical papers*. Cambridge, UK: Cambridge University Press.

Neapolitan, R. (1990). *Probabilistic reasoning in expert systems: Theory and algorithms*. New York: Wiley.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible reasoning*. San Mateo, CA: Morgan Kaufmann.

Phillips, L. (1982). Requisite decision modeling: A case study. *Journal of the Operational Research Society, 33*, 303–311.

Phillips, L. (1984). A theory of requisite decision models. *Acta Psychologica, 56*, 29–48.

Raiffa, H. (1968). *Decision analysis: Introductory lectures on choices under uncertainty*. Reading, MA: Addison-Wesley.

Rescher, N. (1988). *Rationality: A philosophical inquiry into the nature and rationale of reason*. Oxford: Clarendon.

Sage, A. (1991). *Decision support systems engineering*. New York: Wiley.

Savage, L. J. (1954). *The foundations of statistics*. New York: Wiley.

Schum, D. (1990). Inference networks and their many subtle properties. *Information and Decision Technologies, 16*, 69–98.

Schum, D. (1994). *Evidential foundations of probabilistic reasoning*. New York: Wiley.

Shachter, R. (1986). Evaluating influence diagrams. *Operations Research, 34*, 871–882.

Shachter, R., & Heckerman, D. (1987). Thinking backward for knowledge acquisition. *AI Magazine, Fall, 8*, 55–61.

Shafer, G. (1976). *A mathematical theory of evidence*. Princeton, NJ: Princeton University Press.

Shafer, G. (1986). *"Savage revisited," statistical science* (Vol. 1, pp. 463–501). Hayward, CA: Institute of Mathematical Statistics (with comments).

Shanteau, J., Mellers, B., & Schum, D. (1999). *Decision science and technology: Reflections on the contributions of Ward Edwards*. Boston: Kluwer Academic.

Smith, J. Q. (1988). *Decision analysis: A Bayesian approach*. London: Chapman and Hall.

Tocher, K. (1977). Planning systems. *Philosophical Transactions of the Royal Society of London, A287*, 425–441.

Twining, W. (1990). *Rethinking evidence: Exploratory essays*. Oxford: Basil Blackwell.

von Neumann, J., & Morgenstern, O. (1947). *Theory of games and economic behavior*. Princeton, NJ: Princeton University Press.

von Winterfeldt, D., & Edwards, W. (1986). *Decision analysis and behavioral research*. Cambridge, UK: Cambridge University Press.

Watson, S. R., & Buede, D. (1987). *Decision synthesis: The principles and practice of decision analysis*. Cambridge, UK: Cambridge University Press.

Watson, S. R., Weiss, J. J., & Donnell, M. L. (1979). Fuzzy decision analysis. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-9*(1), 1–9.

Winkler, R. L. (1972). *Introduction to Bayesian inference and decision*. New York: Holt, Rinehart, and Winston.

Decision Analysis in Practice

James E. Matheson
SmartOrg, Inc., Menlo Park, CA, USA

Introduction

Decision analysis (DA) is all about practice, as the title of Ronald Howard's defining paper (Howard 1966; presented in 1965) was "Decision Analysis: Applied Decision Theory." He went on to elaborate: "Decision analysis is a logical procedure for the balancing of the factors that influence a decision. The procedure

incorporates uncertainties, values, and preferences in a basic structure that models a decision. Typically it includes technical, marketing, competitive, and environmental factors. The essence of the procedure is the construction of a structural model of the decision in a form suitable for computation and manipulation; the realization of this model is often a set of computer programs."

In about 1968, a program of DA was begun at Stanford Research Institute. This group rapidly grew into a major department called the Decision Analysis Group dedicated to helping decision makers in organizations, both industry and government, reach good decisions, while also consolidating these experiences and doing research on DA methodology (Howard and Matheson 1983). This group was the most intensive DA consulting group through the early 1980s. One of the powerful new methodological tools invented by this group was the Influence Diagram (see entry). DA practice has always developed new tools and approaches based on the challenges of real problems.

At the end of the next decade, with this experience behind him, Professor Howard goes on to say (Howard 1980), "Decision Analysis, as I have described it, is, as a formalism, a logical procedure for decision making. When Decision Analysis is practiced as an applied art the formalism interacts with the intuitive and creative facilities to provide understanding of the nature of the decision problem and therefore guidance in selecting a desirable course of action. I know of no other formal-artistic approach that has been so effective in guiding decision-makers."

In this sense there is no real theory of DA. Its philosophy is grounded in decision theory and systems engineering, with more recent contributions from psychology, but in the end it is an applied art. This Decision Engineering approach is discussed in depth in an INFORMS tutorial (Matheson 2005). This article describes some of the keys to good application and the kinds of positive changes DA promotes in the organizations that adopt it.

A Decision: The Defining Element

A decision is defined as an irrevocable allocation of resources. Exactly what is meant by irrevocable depends on the context. If a single individual—the decision maker (DM)—makes and executes a decision, then the decision and the irrevocable action are one – the individual might decide to take one path versus another along a road. Traveling down the new path is an irrevocable decision in the sense that changing the decision would require going back to the junction and taking the second path, but at a later time. However, when an organizational DM takes a big strategic decision, the DM asks many other people to take later irrevocable actions, which might not even be fully specified at that time of the original decision (for example, asking someone to find an appropriate company and acquire it). In these settings, a decision is often defined as a commitment to allocate resources, which opens new questions of possible execution failure and nested or sequential decisions. In any case, the decisions at hand provide the focus for DA, which distinguishes DA from all kinds of studies and statistical analyses that are not directly serving decisions. This means that, once the decision arena has been defined, the DM can guide all subsequent activity, such as modeling and information gathering, on its ability to inform better decisions. Issues that might make a great deal of difference to the outcome, but do not have the potential to change the decision taken, are unimportant, while issues of less impact but that do inform the decision are of greater importance. The DM uses this sort of decision sensitivity to intuitively and analytically guide the whole process, and to do what is most important to making a decision in the limited time and resources available to make it.

Framing: The Perceived Situation

Perhaps the biggest decision failure is a careful analysis of the wrong problem. Often a decision arises in an organization as just another tactical decision, when actually new strategies are called for – but strategy is not the prerogative or in the comfort zone of those considering the decisions. Thus, old products and whole companies are displaced by competitors who perceived the situation differently, and who were able to act in new ways. Also, executives spend most of their time and energy operating efficiently and find it difficult to "waste time" on strategy or to get into a strategic mind set. The beginning of a DA should review the decision frame, possibly bringing in outside perspectives

and new team members, often expanding the frame, and then reviewing that frame at key points during the process. When a DA process gets stuck, reframing maybe in order (Matheson 1990).

Outcomes: What are the Results

In the face of uncertainty, the decision maker (DM) is forced to distinguish between decisions – what can be done, outcomes – what happens, and preferences – what is wanted. The DM wants good outcomes, but can only control the quality of the decisions, not the outcomes. For example, the DM may invest $10,000 in a venture having only a 10% chance of returning $10,000,000, and considers that a good investment. Quite likely, however, the bad outcome may occur. Clearly, the quality of this decision cannot be judged by its outcome; a bad outcome should not dissuade the DM from looking for similar good investments later. Given this distinction between decision quality and outcome quality, there is a need for a definition of a good decision – DA itself is that definition!

In many organizational cultures, champions are asked to claim that investment proposals are sure things and guarantee that they will succeed. On course, many of these investments fail, but inconsistency does not stop this irrational culture from persisting. However, organizations that can overcome a culture of hiding from uncertainty and instead actually search for the hidden uncertainties in their investments often outperform those that do not. Good DA vets these uncertainties, assesses their probabilities and impacts, and determines what to do about them, such as information gathering and hedging, or even creating new alternatives, before proceeding to recommend the primary decision – a principle called embracing uncertainty (Matheson and Matheson 1998).

There are well established procedures for assessing uncertainties and avoiding well-known biases, such as the work on probability assessment processes by Spetzler and Staël von Holstein (1975). Most practical decision analyses, however, do not require such careful assessment; three points, say 10-50-90 percentiles, are so much better than one single and often biased point. It is essential that those three points not be biased. Most of the de-biasing techniques of Spetzler and Staël von Holstein (1975) are useful preparation before assessing even a three-point distribution. Perhaps the most useful technique is backcasting, as it simultaneously eliminates all sorts of biases.

Preferences: What is Wanted

Because only one thing can be maximized, a good or optimal decision cannot be defined without being clear about value trade-offs that create a single measure to maximize. In most commercial decision analyses, it is best to reduce all values to monetary ones. In fact, seeking a monetary value scale is always a good practice, because money can often be spent to create better alternatives or seek better information, and, without a monetary scale, the DM cannot evaluate those efforts. There is a story about a Swedish executive who had promised the residents of a town that he would never close their factory, but, under hard times, he was facing heavy losses by keeping it open. He was asked by a decision analyst if he would close it if he were losing a million dollars a year, to which he quickly answered, "of course not – this is Sweden where we owe that much to the community." He was then asked if he would close the plant if it were losing a hundred million dollars a year, to which he replied, "it would be our duty to close it as the country and our company cannot sustain such heavy losses." After haggling over the price, he realized that the high monetary value he had just made explicit allowed him to visualize new alternatives, where he would close the plant, pay some additional closing costs to the community and guarantee workers jobs in other factories. He ultimately took these actions and saved his company from financial ruin. Being forced to make a monetary value tradeoff enabled him to invent to better alternatives. He was not valuing things like higher employment on an absolute scale. He was only assessing a tradeoff value in the context of his specific decision – this value is personal and subjective, just like probability, in this case not his own, but one he expresses as a fiduciary of the company he represents. Converting values to monetary equivalents is an excellent practice, because it establishes how much money could be afforded to build new alternatives – money is a common denominator to translate disparate values.

Decision Analysis in Practice, Fig. 1 Risk tolerance in millions of dollars as measured from *top* executives of three publically traded companies, A, B, and C

Size Measure	A	B	C	Approximate Ratio to Risk Tolerance
Net Sales	2,300	16,000	31,000	15:1
Net Income	120	700	1,900	1:1
Equity	1,000	6,500	12,000	6:1
Market Value	940	4,600	9,900	5:1
Risk Tolerance	150	1,000	2,000	

What about value over time? In a simple case, a highly rated company regularly adjusts or rebalances its financial capital at a weighted cost of capital of R%. If the company has opportunities (or preferences) that imply a value other than R%, the company should rearrange its investments using its banking relationships until its needs are exactly in line with the financial rate of R%. At that point, the company's own time preferences are exactly the same as the financial rate. Because of this harmonization process, this cost of capital becomes the company's own time value of money. Another way to state this observation is that the company should invest to maximize net present value (NPV) at its cost of capital, and then spread that NPV over time optimally using financial transactions at the same rate, separating investment funding and usage decisions.

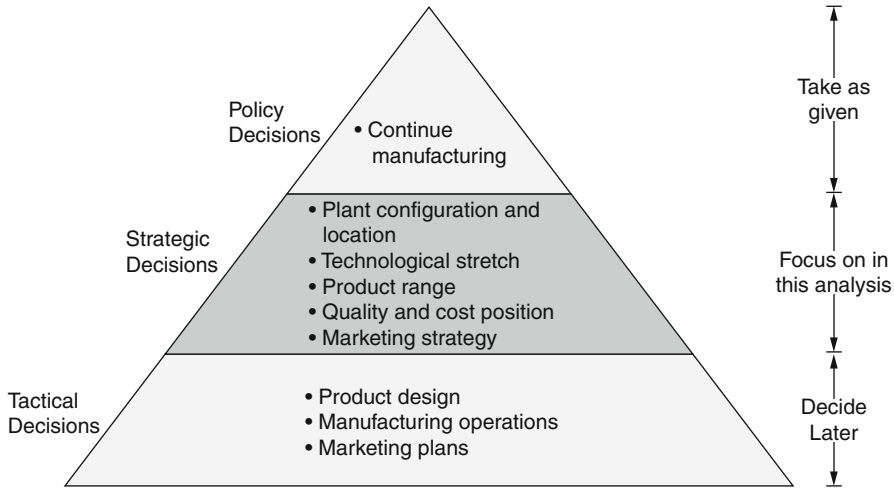
How should preferences under uncertainty be treated? Assuming that each uncertainty has been characterized satisfactorily in the form of probability distributions over NPV, which investment should be picked? If the company is large enough to undertake many investments of this size during each year, then maximizing the expected value is a reasonable way to maximize long-term economic-value creation. However, if the range of the uncertainties could impact the financial structure or soundness of the company, it would be wise for it to be risk averse. Some financial pundits argue that companies traded on the stock market should not be risk averse as the shareholders can diversify. There are many arguments against this position, including the actual behavior of most companies, the cost of bankruptcy or other financial distress, the inability of the shareholder to gain information and change positions quickly (lack of liquidity), but, perhaps most significantly, are the availability of risk hedging options to the company that are not available to shareholders. The risk attitude of the company is assessed by asking series of questions about which of several hypothetical investments they would undertake or reject. This attitude is almost always captured as the risk tolerance, say expressed in millions of dollars, which is the parameter of an exponential utility function:

$$U(x) = -ae^{(-x/\rho)} \text{ where } a > 0 \text{ and } \rho = \text{risk tolerance}$$

One test question to determine the risk tolerance is considering a hypothetical but typical investment, in terms of complexity and time duration, where there is a 0.5 probability of winning the risk tolerance and a 0.5 probability of losing one-half that amount. The risk tolerance is then adjusted until the DM is indifferent between taking and rejecting this investment.

There are good arguments that risk tolerance should be set for the total organization and not for a division or a project. One advantage of being a division of a large organization is to be able to use the corporate risk tolerance, which a similar stand-alone organization could not do. Figure 1 compares the measured risk tolerances of three large corporations, which were all engaged in a joint venture. This chart can be used to get an initial approximation for other public companies, commonly by estimating risk tolerance as 1/6 of shareholders' equity or 1/5 of the market value of outstanding shares of stock.

Investments with a range of outcomes on the order of the risk tolerance need explicit treatment using utility theory. Investments with a range of outcomes less than of 10% of the risk tolerance should usually be evaluated using expected values, and investments with a range of outcomes larger than the risk tolerance should be avoided, partnered, or treated by a very



Decision Analysis in Practice, Fig. 2 Decision hierarchy for a plant modernization decision

experienced decision analyst. The author has seen one such case in a lifetime of professional practice. If the exponential utility will not suffice, the analysis is in very deep water indeed! In dealing with uncertainties large enough to require risk aversion, there is a need to beware of dependencies among uncertainties in other investments or the background cash flow of the organization. Hedging and diversification impacts are likely to overshadow other considerations.

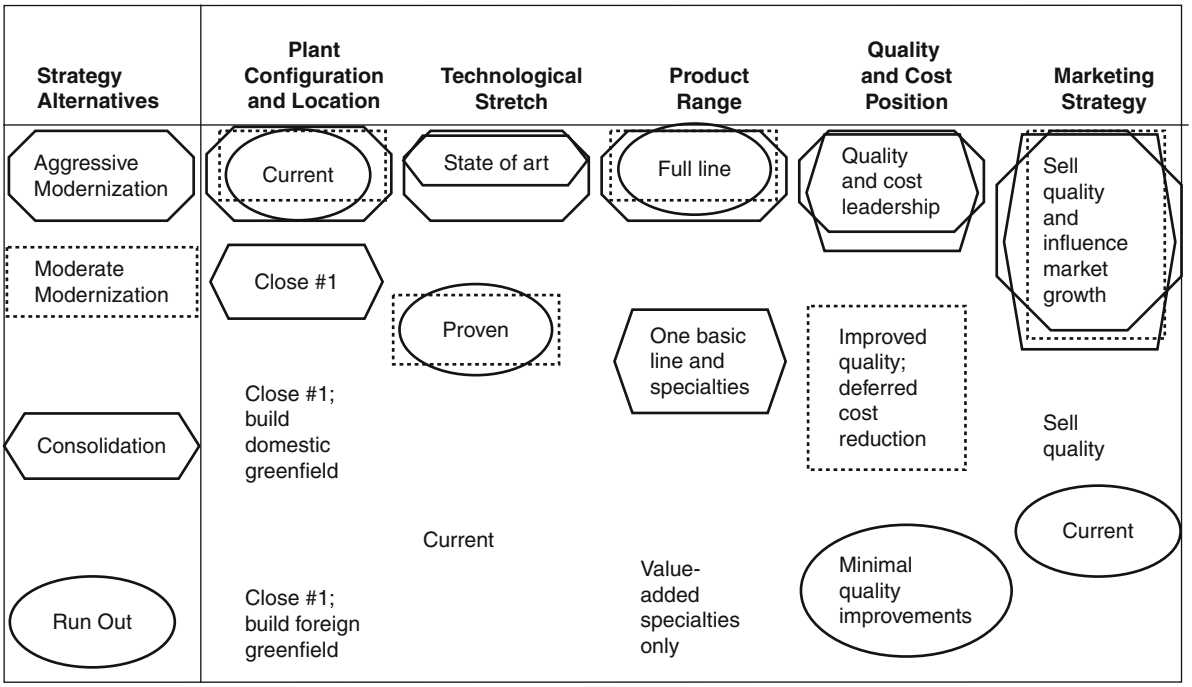
Alternatives: What Can be Done

In simple decisions problems, such as classroom examples, a limited number of well-specified alternatives are given. In most real situations, however, new alternatives can and should be created to uncover more valuable ones. Part of the natural reluctance of organizations to generate and consider new alternatives is that the decision problems arise out of situations where natural alternatives are evident. In addition, those product or investment champions and others who have made an emotional investment by picking winners prematurely, see alternative generation as a waste of time or even a direct threat. There are many ways to create new alternatives, but a simple one is to use the project team itself in a session with a ground rule that at least five new significantly different alternatives must be developed. There are many tools to stimulate creativity, most requiring that a wealth of information and new possibilities be put on the table before evaluating them; such as examples of what others have been done, what competitors are saying, what consumers are asking for. After the analysis enters the financial modeling stage; sensitivity analysis should also be used to drive the discussion of alternatives that minimize risk (hedge or diversify) or take advantage of uncertainties.

For situations with complex multidimensional alternatives, decision hierarchies and strategy tables are extremely useful. The decision hierarchy for a plant modernization decision (Fig. 2) identifies the strategic decisions under consideration, the policy decisions that are not currently being questioned, and the tactical or implementation decisions which will be made or optimized after the strategy is selected. The list of identified strategic decisions are further specified in the columns of the strategy table, illustrated in Fig. 3. The columns list specific mutually-exclusive alternatives for each strategy variable. Thus, a selection of one item from each column constitutes a well-specified strategic alternative. The special column at the left gives names and symbols for each alternative, which is read by following its symbol across the columns. Further descriptions of these tools can be found in Matheson and Matheson (1998) and McNamee and Celona (2007).

Decision Modeling: Analyzing as Simply as Possible

The process of DA uses the decision to be made as a guide to cut through many complex modeling issues. Often details, such as numerous market segments or



Decision Analysis in Practice, Fig. 3 Strategy table for a plant modernization decision

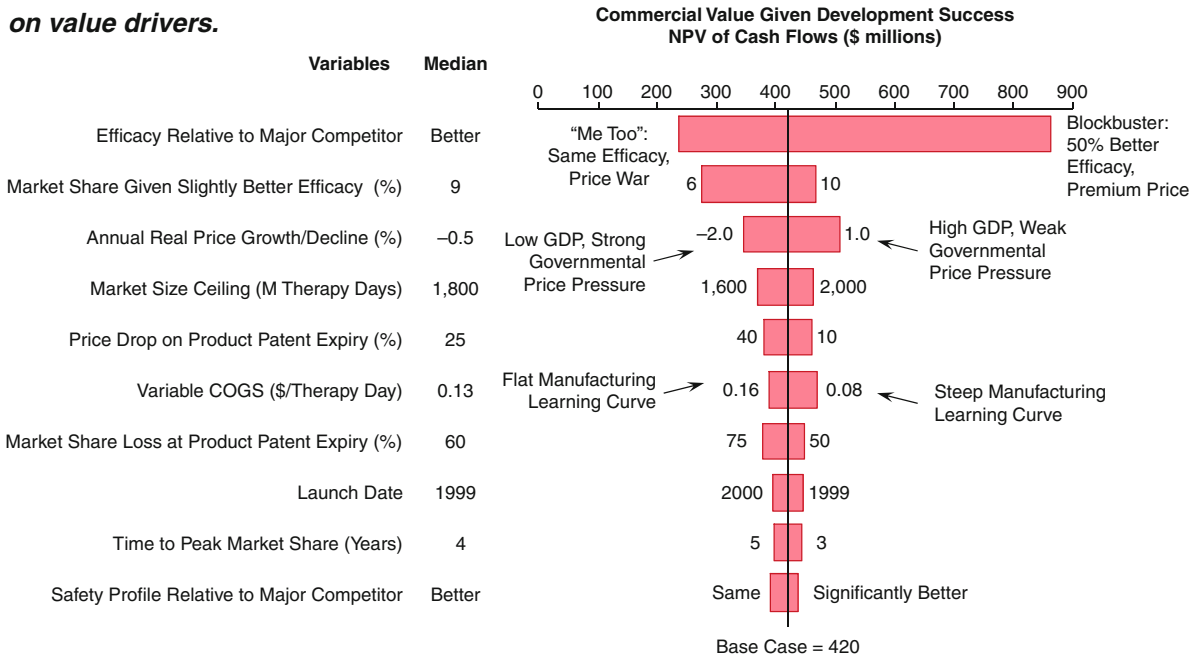
multiple product generations, can be treated with multipliers, followed by sensitivity analysis to the value of those multipliers, to determine if something important was missed. Verisimilitude is unimportant, only the impact on gaining clarity of action. Good modeling for decision making is an important professional task, see McNamee and Celona, (2007).

A special kind of sensitivity analysis called a tornado chart (Fig. 4) is a key tool for checking the model and gaining new insights. Each uncertain variable is varied one at a time over the range of the low (10 percentile) and high (90 percentile) assessments, to determine the range of (deterministic) NPV resulting from different runs of the model, usually while holding the other values at their medians. Notice that output ranges of each variable correspond to the same range of uncertainty on their inputs, so if the results are arranged in a decreasing order of the output ranges, they are also in order of the impact of each uncertainty on value, as in Fig. 4. Since for independent variables, the uncertainty ranges should add as the square root of the sum of the squares, only the first several results are big contributors, which often produces insight into which factors are driving risk, as well as ideas for how to reduce that risk. More sophisticated tornado diagrams overlay results for multiple alternatives to give insight into which uncertainties could actually cause a decision switch, as these would be the most critical to learn more about.

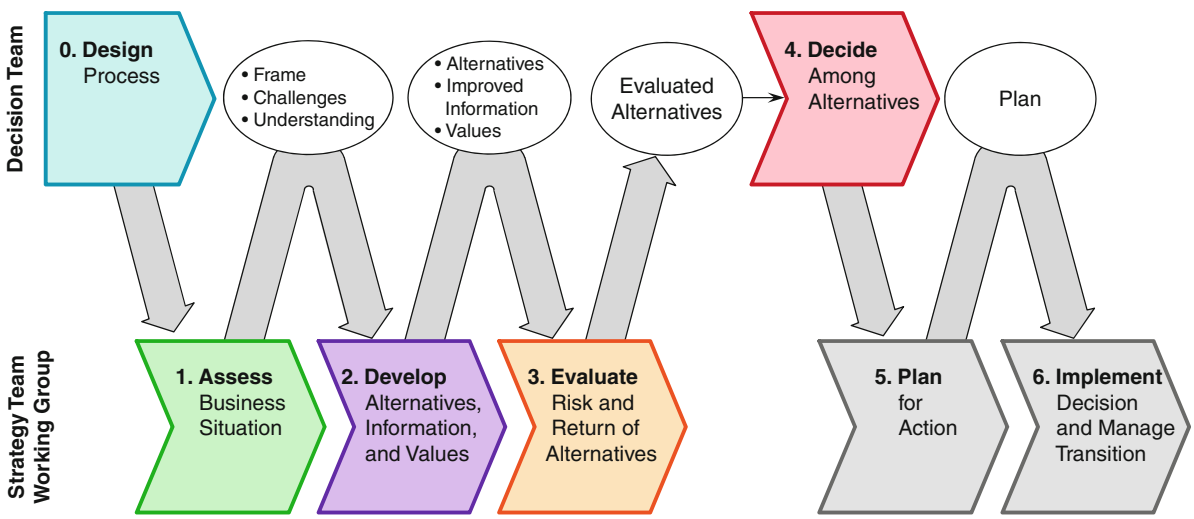
Commitment to Action: Getting It Done

The author has decided to diet many times, without actually following through. And that is only dealing with himself! It is much more difficult to align an organization to carry out the chosen action. A good analysis sets the stage for implementation success at the beginning by the choice of individuals involved in reaching the decision. It is natural not to put the potential naysayers on the decision making or the decision analyzing team, but if they are not chosen, they will often veto the result, overtly if they have the power and covertly if not. It is best to put any skeptical person with veto power on the team, even if only in a review board role, and require that they raise their issues during the analysis process rather than objecting later – speak up or forever hold your peace. In this way they have the opportunity to inform the team of their

Dealing effectively with uncertainty builds trust in the evaluation framework and helps focus attention on value drivers.



Decision Analysis in Practice, Fig. 4 Tornado chart



Decision Analysis in Practice, Fig. 5 Dialog decision process

important issues, which can be taken into account during the analysis, and they acquire a deeper understanding of the decision situation by participating, giving them a much better chance of ultimately buying in to the conclusions. It gives them needed psychological time and space to reconsider and revise long held convictions. Also, put key implementers on the team so they understand and buy

Decision Analysis in Practice, Fig. 6 Decision quality chain

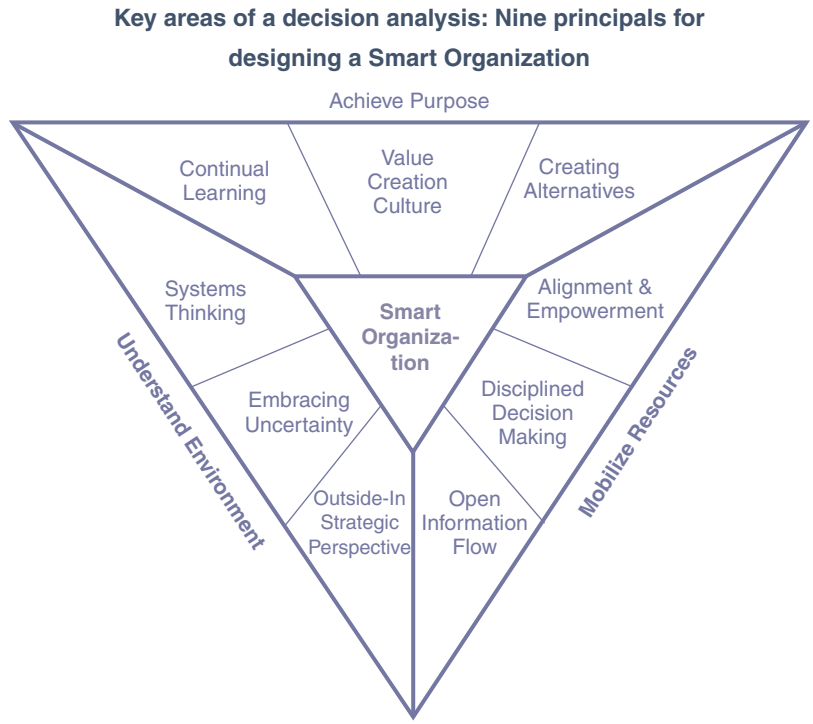
into what they are asked to implement. The Dialog Decision Process (Fig. 5) was devised to organize all of these actors into a highly workable project structure.

The Decision Quality Chain

The key elements described above are often arranged in a decision quality chain (Fig. 6), originally proposed by Carl Spetzler (Keelin and Spetzler 1992). The metaphor of a chain is used to express that the chain is only as good as its weakest link – that is the most important one; the weakest link changes as the DA proceeds. Decision analysts sometimes use a spider diagram to score progress at each team review (Keelin et al. 2009).

Embedding Good Decision-Making Skills into Organizations

The book, *The Smart Organization*, (Matheson and Matheson 1998), describes "Nine Principles of a Smart Organization" that characterizes a set of habits and a mindset conducive to good decisions, Fig. 7. This book also presents an organizational IQ test to measure compliance with these norms. These tests have been administered to thousands of organizations. The payoff for being a smart organization was striking – organizations in the top quartile of IQ were over five times more likely to be in the top quartile of financial performance, as reported in Matheson and Matheson (2001). Organizations with high scores have patterns of behavior that enable them to spontaneously see the need for decisions, request and frame appropriate decision analyses, and conduct and participate in decision analyses more efficiently and effectively. A few organizations are leading the way by integrating DA into their organizational DNA. Among them, most notably, has been Chevron, which won the annual Decision Analysis Society's Practice Award (2010) for "The implementation of Decision Analysis Practice at Chevron: 20 years of building a DA culture." Matheson and Matheson (2007) discuss how DA principles can become the basis of the Decision Organization.



Decision Analysis in Practice, Fig. 7 Nine principals (Matheson and Matheson 1998)

Concluding Remarks

DA has evolved from specialized high-level consulting to changing culture and embedding processes into organizational routines. The various roles that a DA professional might be called upon to play include:

1. Decision Analyst - responsible for processing numbers,
2. Decision Facilitator - responsible for meetings,
3. Decision Consultant - responsible for attaining commitment,
4. Decision Engineer - responsible for process, systems and organizational design,
5. Decision Change Agent - responsible for personal, organizational, and cultural change necessary for routine, high quality decision making.

See

- ▶ Decision Analysis
- ▶ Decision Making and Decision Analysis
- ▶ Decision Trees
- ▶ Influence Diagrams

References

Howard, R. A. (1966). Decision analysis: Applied decision theory. In Hertz, D. B., & Melese, J. (Eds.), *Proceedings of the Fourth International Conference on Operational Research* (pp. 55–71).

Howard, R. A. (1980). An assessment of decision analysis. *Operations Research, 28*(1), 4–27.

Howard, R. A., & Matheson, J. E. (Eds.). (1983). *Readings on the principles and applications of decision analysis*. Menlo Park: Strategic Decisions Group.

Keelin, T., Schoemaker, P., & Spetzler, C. (2009). *Decision quality – The fundamentals of making good decisions*. Palo Alto, CA: Decision Education Foundation.

Keelin, T., & Spetzler, C. (1992). *Decision quality: Opportunity for leadership in total quality management*. Palo Alto, CA: Strategic Decision Group.

Matheson, D. (1990). *When should you reexamine your frame?* Ph.D. dissertation, Stanford University.

Matheson, J. (2005). *Decision analysis = decision engineering*, Ch.7 (pp. 195–212). Tutorials in Operations Research INFORMS 2005.

Matheson, D. & Matheson, J. (1998). *The smart organization, creating value through strategic R&D*. Harvard Business School Press.

Matheson, D., & Matheson, J. (2001). *Smart organizations perform better*, *Research-Technology Management*, Industrial Research Institute, July-August.

Matheson, D., & Matheson, J. (2007). From decision analysis to the decision organization. In W. Edwards, R. Miles Jr., & D. von Winterfeldt (Eds.), *Advances in decision analysis:*

From foundations to applications. Cambridge: Cambridge University Press.
McNamee, P., & Celona, J. (2007). *Decision analysis for the professional*. Menlo Park, CA: SmartOrg.
Spetzler, C., & Staël von Holstein, C.-A. (1975). Probability encoding in decision analysis. *Management Science, 22*, 340–358.

Decision Maker (DM)

An individual (or group) who is dissatisfied with some existing situation or with the prospect of a future situation and who possesses the desire and authority to initiate actions designed to alter the situation. In the literature, the letters DM are often used to denote decision maker.

See

▶ Decision Analysis
▶ Decision Analysis in Practice
▶ Decision Making and Decision Analysis

Decision Making and Decision Analysis

Dennis M. Buede
Innovative Decisions, Inc., Vienna, VA, USA

Introduction

Decision making is a process undertaken by an individual or organization. The intent of this process is to improve the future position of the individual or organization, relative to current projections of that future position, in terms of one or more criteria. Most scholars of decision making define this process as one that culminates in an irrevocable allocation of resources to affect some chosen change or the continuance of the status quo. The most commonly allocated resource is money, but other scarce resources are goods and services, and the time and energy of talented people.

Once the concept of making a decision is accepted as a human action, an immediate question is "what is the difference between a good and a bad decision?" The common tendency is to attribute good decisions to situations in which good outcomes were obtained. This approach, however, implies that good decisions cannot be recognized when they are made, but only after the outcomes are observed (which may be seconds or decades later). This common tendency also implies that good decisions have nothing to do with the decision-making process; throwing a dart at a chart of alternatives may lead, on occasion, to good outcomes, while long, hard thought about values and uncertainties does not always yield good outcomes. So leaders in the decision analysis field have defined a good decision as one that is consistent with the values and uncertainties of the decision maker (DM) after considering as many relevant alternatives as possible within the appropriate time frame and with the available information. The belief is that better outcomes will be more likely, but are not guaranteed, with a sound decision making process than throwing darts at a chart of alternatives.

Three primary decision modes have been identified by Watson and Buede (1987): (1) choosing one option from a list, (2) allocating a scarce resource(s) among competing projects, and (3) negotiating an agreement with one or more adversaries. Decision analysis is the common analytical approach for the first mode, optimization using decision analysis concepts of value objectives for the second, and a host of techniques have been applied to negotiation decisions.

The three major elements of a decision that cause decision making to be troublesome are the creative generation of options; the identification and quantification of multiple conflicting criteria, as well as time and risk preference; and the assessment and analysis of uncertainty associated with the causal linkage between alternatives and objectives. To claim to have made a good decision, the DM must be able to defend how these three elements were addressed.

Many DMs claim to be troubled by the feeling that there is an, as yet unidentified, alternative that must surely be better than those so far considered. The development of techniques for identifying such alternatives has received considerable attention (Keller and Ho 1988; Keeney 1992). Additional research has been undertaken to identify the pitfalls in assessing probability distributions that represent the uncertainty of a DM (Edwards et al. 2007). Research has also focused on the identification of the most appropriate preference assessment techniques (Edwards et al. 2007). Keeney (1992) has advanced

concepts for the development and structuring of a value hierarchy for key decisions. Very little research has been done on the issue of causal linkages between alternatives and the objectives.

The making of a good decision requires a sound decision making process. However, doing research on competing decision processes, with sound validation using ground truth, is not possible. It is not possible to create multiple versions of reality so that the DM can try the preferred alternative from competing decision processes to identify which would have turned out best. Researchers have proposed multi-phased processes for decision making, e.g., (Howard 1968; Witte 1972; Mintzberg et al. 1976). The common phases include: intelligence or problem definition, design or analysis, choice, and implementation. A weakness in one phase in the decision making process often cannot be compensated for by strengths in the other phases. In general, the decision-making process must address the development of a reasoned set of objectives and associated preference structure; decision alternatives; and the facts, data, opinions, and judgments needed to relate the alternatives to the value model. Then, of course, the logic of evaluating the alternatives in light of the value structure must be sound.

Decision Analysis

The field of decision analysis involves both analysis and synthesis. Analysis is a process for dividing a problem into parts and performing some quantitative assessment of those parts. Synthesis then combines those assessments into a macro assessment. Decision analysis provides an integrating framework for doing this assessment, as well as the theory and techniques for doing the analyses of the parts. These parts are traditionally values (objectives for improving the DM's situation), alternatives (resources the DM can expend to change the world), and the linkage between the alternatives and the values (the facts and uncertainties within the DM's world). Nonetheless, experienced decision analysts often ask the DM for a holistic assessment of the alternatives prior to showing the analysis results (as part of the synthesis process) so that the analysis results can be compared to this holistic standard and the differences noted and examined. Often this comparison to the holistic assessment identifies some issues that were missed in the analysis.

Decision analysis has its roots in many fields. Some of the most obvious are operations research, engineering, business, psychology, probability and statistics, and logic. Fishburn (1999) provides a well-documented summary of these roots of decision analysis. Von Neumann and Morgenstern (1947) provided the first axiomatic structure for decision making, incorporating both probabilistic and value preferences into a principle of expected utility maximization. Savage (1954) recognized the need for subjective probabilities to be combined with subjective utility judgments, leading to subjective expected utility (SEU). Since decision making involves trying to predict how the future world will evolve, the subjectivist approach to uncertainty is the primary perspective adopted in decision analysis. De Finetti (1972) provides a detailed review of the subjectivist approach. Bayes' rule is often required in the computation of expected utility, i.e., Bayesian decision theory is used to describe the decision analysis process (Smith 1988). Interestingly, Bayesian probability and subjectivist probability are used interchangeably. Howard (1966, 1968), Raiffa (1968), and Edwards (1962) all made important contributions in transforming an academic theory into a practical discipline to guide DMs through the difficulties of real world decision making.

Values represent what the DM wants to improve in the future. As an example, when considering the purchase of a new car, the DM may be weighing reduced cost in the future against improved safety, comfort, prestige, and performance. The context of this decision and, therefore, the values, is the likely uses of a car for commuting, long distance travel, errands, etc. Keeney (1992) provides a structure for thinking about how to separate out the ends (or fundamental) objectives from the means objectives. Several authors have defined the mathematics behind the quantification of a value structure for the analysis of alternatives, see Keeney and Raiffa (1976), French (1986), and Kirkwood (1997). In general, the quantification of preferences must deal with tradeoffs among objectives, risk preference introduced by uncertainty, and time preference introduced by achieving payoffs across the objectives at different points in time. Besides having complex issues to quantify, the DM must deal with subjective

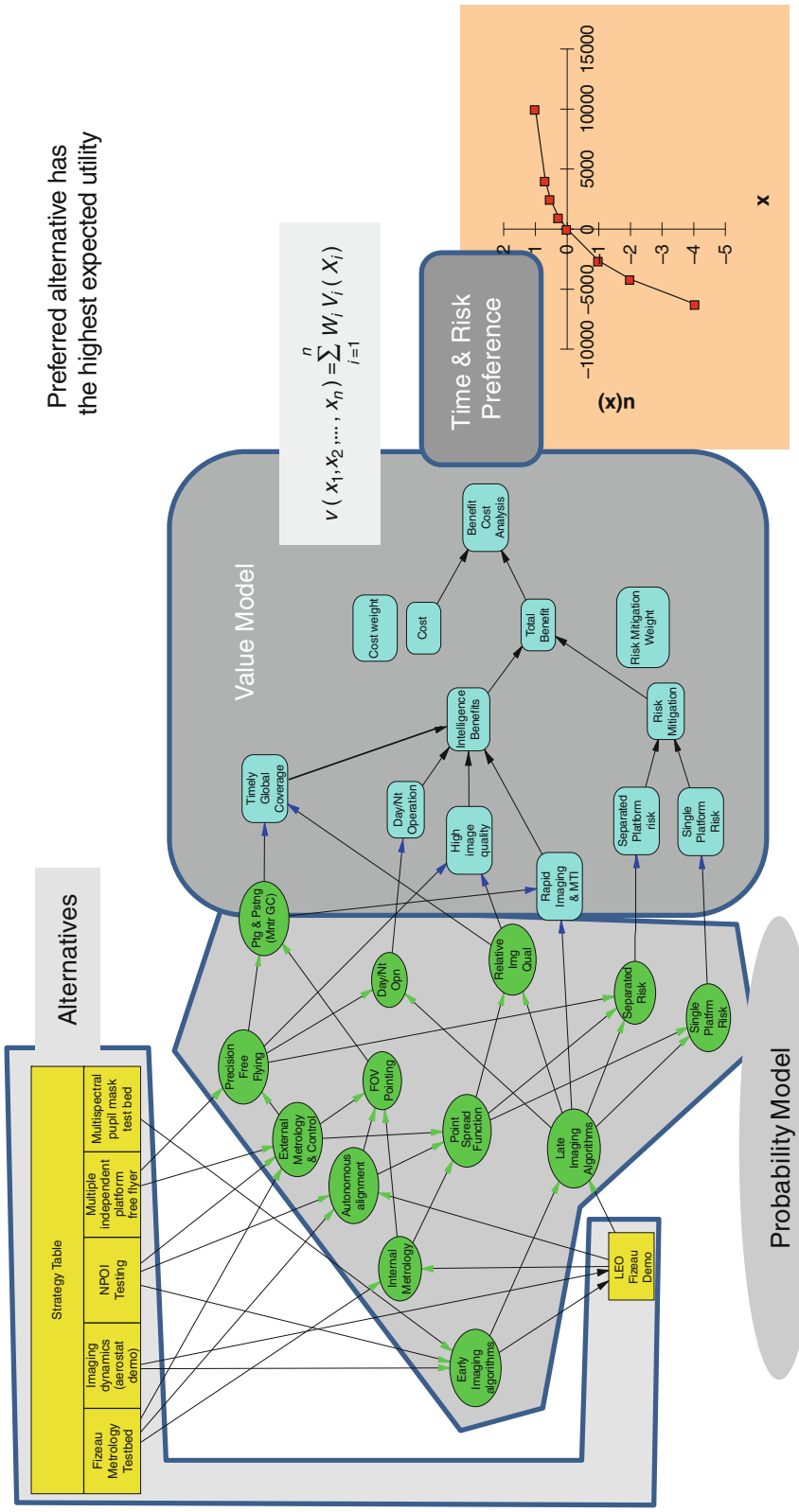
judgments, because there can be no source of preference information other than human judgment. Those approaches that attempt to avoid human judgment are throwing the proverbial baby out with the bath water.

Alternatives are the actions (expenditures of resources) that the DM can take now and into the future. In general, the set of alternatives also includes what are termed options or delayed actions that the DM can decide to take in the future if certain events occur between now and the time associated with the option. The space of alternatives is commonly defined over a discrete set, though there is nothing in the theory of decision analysis that precludes a continuous selection set. Various processes have been used to define this set of alternatives, including brainstorming activities. The most commonly discussed approach is called a strategy table or morphological box (Buede 2009). The strategy table divides the alternative space (including any options) into a discrete number of elements or components. For each element, multiple possible selections are defined. The combination of elements and choices within each element are analogous to a buffet dinner during which each diner selects zero, one or more choices from each element and places them onto a plate. If we require each diner to take one and only one selection from each of N elements of the dinner, there are (n_1 x n_2 x ... x n_N) possible dinners that could be selected. When the choice process is broaden to include no selection or several selections from each element, the number of possible dinners grows. (Note: it is also possible that some of these combinations are impossible or very negatively valued.) Typically, members of the decision-making team are asked to pick five to fifteen representative and interesting selections from the large number of possible selections for the analysis to consider. Often, the evaluation of the initial selection of alternatives from the strategy table will be followed by a second selection of alternatives from the strategy table, with a second round of analysis for this new set. The second set (and possibly a third set) would examine alternatives more like those that did well in the first evaluation and less like those that did poorly.

The linkage between the alternatives and values (both certain and uncertain) is the third element of analytic decomposition of decision analysis. Some parts of this linkage may be well known and deterministic, such as a specific cost of a car, a defined amount of money to purchase. Other parts of this linkage may not be well known, thus requiring the development of a probability distribution; for example, the same car with a known purchase price may not have such a well-known operating cost over the next five to ten years. In some cases, we can develop a probability distribution for this intermediate variable which has a known relationship to a measure for the relevant objective. In other cases, the relationship to one of the objectives may also be probabilistic, requiring the development of an influence diagram with chance nodes separating some or all of the alternatives from the objectives, see Fig. 1.

Once the analytical structure has been built by decomposing the decision problem into such constructs as alternatives, value models, and uncertainties, there is a need to compute (or synthesize) the expected utility of each possible alternative, and to answer additional questions that the DM may have. Examples of common questions are: there is some disagreement about what the risk preference (or time preference or value trade-offs or probabilities) should be, does this make any difference?; alternatives 1 and 2 are much better than the rest, but are very close in expected utility, what are the major differences between these two alternatives?; if one cannot be sure about some parameter's value in the model, will changing it by x% change the order of the alternatives in terms of expected utility? This whole process of computing the results and posing/answering questions regarding the meaning of the analysis and the robustness of the parameters in the analysis is called synthesis. This is exactly why a quantitative model is so much more helpful than a qualitative model. A qualitative model cannot provide these answers without a great deal of fuzziness, leading to continued discussion and argument.

A common criticism of decision analysis is that those involved cannot provide the preference and probabilistic numbers reliably and consistently. Many years of research has demonstrated this conclusively (Edwards et al. 2007; von Winterfeldt and Edwards 1986; Watson and Buede 1987). The real question, however, is not whether humans can provide these judgments accurately, but whether inaccurate judgments for a specified quantitative model leads to a better conversation about the decision being made than does a meandering, fuzzy conversation that starts and stops many times without having such a model or



Decision Making and Decision Analysis, Fig. 1 Representative Influence Diagram

any other anchor guiding it. Those who have participated in such meandering, fuzzy conversations have been often left with an empty feeling that there is no real agreement or understanding about the implications of the decision. As long as the key DMs have been involved in the quantitative modeling and understand the results of the synthesis, it is possible to argue that the quantitative analysis, with all of it flaws, has produced useful insights into the decision and provides an accurate audit trail about what was known and not known at the time of decision. The quantitative model is, however, a model and thus subject to the famous quote: "Essentially, all models are wrong, but some are useful" (Box and Draper 1987, p 424).

Decison Analytic Strategies

Many individuals and consulting companies have aided DMs and their organizations to arrive at better decisions. Watson and Buede (1987, pp. 123-159) identified five strategies: (1) modeling, (2) introspection, (3) rating, (4) conferencing, and (5) developing. A sixth strategy that is added here is aggregating mathematically.

1. **Modeling**. The modeling strategy involves building complex representations (models) that link the selection of specific options or alternatives to the values of the DM so that the expected utility across time of each option can be calculated. These models may be decision trees, influence diagrams (Shachter 1986) or simulation models. This approach runs the risk that the DM cannot understand the modeling and, therefore, does not gain the important insights from the model nor trust the results.
2. **Introspection**. The introspection strategy requires deep thought about (i.) the multiple-objective utility function across competing objectives, and (ii.) the joint probability distribution that relates the alternatives to these objectives. This approach is characterized by a question and answer process involving the decision analyst and a single DM (Keeney 1977). This approach does not benefit from additional opinions and expertise beyond the single DM.
3. **Rating**. The rating strategy is the simplest and most used. This strategy typically involves the assumption of an additive value model across multiple objectives, while ignoring time and risk preference, and a deterministic relationship between each alternative, the set of objectives, and their measures. Edwards (1971) introduced this approach under the acronym SMART, but later changed it to SMARTS to reflect the importance of using swing weights rather than importance weights. This approach ignores the complexities of value issues and uncertainty relating the alternatives to the objectives, and uses an ad hoc approach towards gathering information from other participants and experts.
4. **Conferencing**. The conferencing strategy employs simple models as used in Rating with a carefully constructed group (Phillips 2007). The advantage of the simple model is that it is transparent enough to the group to be trusted, and can then focus group discussions across the spectrum of concerns characterized by the objectives, allowing the appropriate experts to weigh in on their topics of expertise. This approach assumes the complexity of the problem is being addressed by the collection of individuals in their reasoning processes, but always runs the risks that the collective reasoning process has interpreted the complexity incorrectly. This alternative reasoning process is difficult to document and scrutinize. Other conferencing approaches exist that utilize computer technology extensively (Nunamaker et al. 1993). These technological approaches to conferencing emphasize giving every participant a chance to enter their inputs via keypads, often limiting discussion. The critical issue is information transfer via open discussion versus group domination by a few individuals. The collective reasoning process is even harder to assess when individuals are communicating via key pads.
5. **Developing.** The developing strategy involves the development of a decision support system that will be used by an individual or collection of individuals for a specific class of decisions over time. This approach usually adopts either a modeling or rating approach to be embed inside the decision support system, along with access to a changing database (see Sauter (1997) for a summary). There continues to be a wide variety of software implementations that serve as a basis for these decision support systems.
6. **Aggregating mathematically**. There are a number of academics and some practitioners who believe a group is best supported by analyzing the decision

from each individual's perspective, and then creating a mathematical aggregation of those individual perspectives. These approaches have been categorized as: social choice theory, group utility analysis, group consensus, and game theory.

See

- Computational Organization Theory
- Corporate Strategy
- Decision Analysis
- Decision Analysis in Practice
- Decision Support Systems (DSS)
- Influence Diagrams
- Multi-attribute Utility Theory
- Multiple Criteria Decision Making
- Simulation of Stochastic Discrete-Event Systems
- Utility Theory

References

Box, G., & Draper, N. (1987). *Empirical model-building and response surfaces*. New York: John Wiley.

Buede, D. M. (2009). *The engineering design of systems: Models and methods*. New York: John Wiley.

De Finetti, B. (1972). *Probability induction, and statistics: The art of guessing*. New York: John Wiley.

Edwards, W. (1962). Dynamic decision theory and probabilistic information processing. *Human Factors, 4*, 59–73.

Edwards, W. (1971). Social utilities. *The Engineering Economist, 6*, 119–129.

Edwards, W., Miles, R. F., Jr., & von Winterfeldt, D. (Eds.). (2007). *Advances in decision analysis: From foundations to applications*. New York: Cambridge University Press.

Fishburn, P. (1999). The making of decision theory. In J. Shanteau, B. Mellers, & D. Schum (Eds.), *Decision science and technology: Reflections on the contributions of Ward Edwards* (pp. 369–388). Boston, MA: Kluwer.

French, S. (1986). *Decision theory: An introduction to the mathematics of rationality*. Chichester, UK: John Wiley.

Hammond, F. S., Keeney, R. L., & Raiffa, H. (1999). *Smart choices: A practical guide to making better decisions*. Cambridge, MA: Harvard Business School.

Howard, R. (1966). Decision analysis: Applied decision theory. In Hertz, D.B., & Melese, J. (eds), *Proceedings fourth international conference on operational research*. New York: Wiley-Interscience.

Howard, R. (1968). The foundations of decision analysis. *IEEE Transactions on Systems, Science, and Cybernetics, SSC-4*, 211–219.

Keeney, R. L. (1977). The art of assessing multiattribute utility functions. *Organizational Behavior and Human Performance, 19*, 267–310.

Keeney, R. (1992). *Value-focused thinking*. Boston: Harvard University Press.

Keeney, R. A., & Raiffa, H. (1976). *Decisions with multiple objectives: Preferences and value tradeoffs*. New York: John Wiley.

Keller, L., & Ho, J. (1988). Decision problem structuring: Generating options. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-15*, 715–728.

Kirkwood, C. W. (1997). *Strategic decision making: Multiple objective decision analysis with spreadsheets*. Belmont, CA: Duxbury Press.

Mintzberg, H., Raisinghani, D., & Theoret, A. (1976). The structure of 'unstructured' decision processes. *Administrative Sciences Quarterly, 21*, 246–275.

Nunamaker, J., Dennis, A., Valacich, J., Vogel, D., & George, J. (1993). Group support systems research: Experience from the lab and field. In L. Jessup & J. Valacich (Eds.), *Group support systems*. New York: Macmillan.

Phillips, L. D. (2007). Decision conferencing. In W. Edwards et al. (Eds.), *Advances in decision analysis*. New York: Cambridge University Press.

Raiffa, H. (1968). *Decision analysis: Introductory lectures on choices under uncertainty*. Reading, MA: Addison-Wesley.

Sauter, V. L. (1997). *Decision support systems: An applied managerial approach*. New York: John Wiley.

Savage, L. J. (1954). *The foundations of statistics*. New York: John Wiley.

Shachter, R. D. (1986). Evaluating influence diagrams. *Operations Research, 34*, 871–882.

Smith, J. Q. (1988). *Decision analysis: A Bayesian approach*. London: Chapman and Hall.

von Neumann, J., & Morgenstern, O. (1947). *Theory of games and economic behavior*. Princeton, NJ: Princeton University Press.

von Winterfeldt, D., & Edwards, W. (1986). *Decision analysis and behavioral research*. New York: Cambridge University Press.

Watson, S., & Buede, D. (1987). *Decision synthesis: The principles and practice of decision analysis*. Chichester, UK: Cambridge University Press.

Witte, E. (1972). Field research on complex decision-making processes–The phase theorem. *International Studies of Management and Organization*, 156–182.

Decision Problem

The basic decision problem is as follows: Given a set of r alternative actions $A = \{a_1,\ldots, a_r\}$, a set of q states of nature $S = \{s_1,\ldots, s_q\}$, a set of rq outcomes $O = \{o_1, \ldots, o_{rq}\}$, a corresponding set of rq payoffs $P = \{p_1,\ldots, p_{rq}\}$, and a decision criterion to be optimized, $f(a_j)$, where f is a real-valued function defined on A, choose an alternative action a_j that optimizes the decision criterion $f(a_j)$.

See

- ▶ Decision Analysis
- ▶ Decision Analysis in Practice
- ▶ Decision Making and Decision Analysis
- ▶ Group Decision Making
- ▶ Multi-Criteria Decision Making (MCDM)
- ▶ Utility Theory

Decision Support Systems (DSS)

Andrew Vazsonyi
University of San Francisco, San Francisco, CA, USA

Introduction

Throughout history there has been a deeply embedded conviction that, under the proper conditions, some people are capable of helping others come to grips with problems in daily life. Such professional helpers are called counselors, psychiatrists, psychologists, social workers, and the like. In addition to these professional helpers, there are less formal helpers, such as ministers, lawyers, teachers, or even bartenders, hairdressers, and cab drivers.

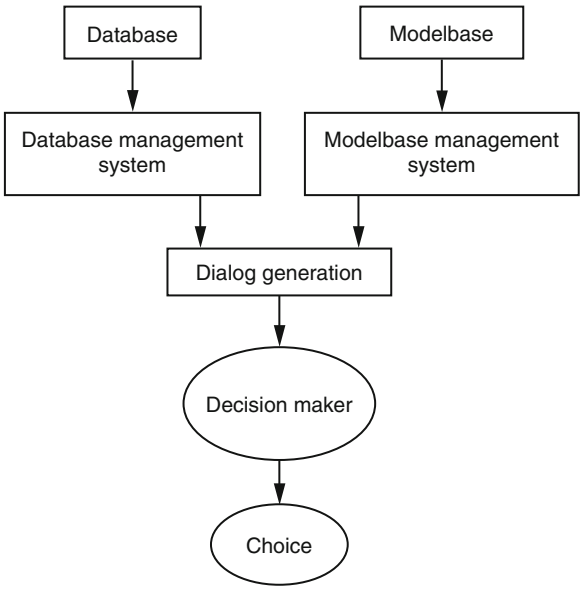
The proposition that science and quantitative methods, such as those used in OR/MS, may help people is relatively new, and is still received by many with deep skepticism. There are some disciplines overlapping and augmenting OR/MS. One important one is called decision support systems (DSS).

Before discussion of DSS, it is to be stressed that the expression is used in a different manner by different people, and there is no general agreement of what DSS really is. Moreover, the benefits claimed by DSS are in no way different from the benefits claimed by OR/MS. To appreciate DSS, a pluralistic view must be taken of the various disciplines offered to help managerial decision making.

Features of Decision Support Systems

During the early 1970s, under the impact of new developments in computer systems, a new perspective about decision making appeared. Keen and Morton (1973) coined the expression decision support systems, to designate their approach to the solution of managerial problems. They postulated a number of distinctive characteristics of DSS, especially the five listed below:

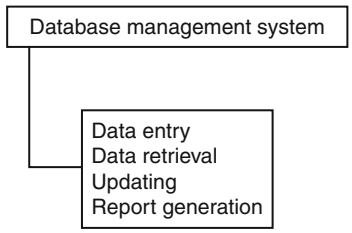
- A DSS is designed for specific decision makers and their decision tasks,
- A DSS is developed by cycling between design and implementation,
- A DSS is developed with a high degree of user involvement,
- A DSS includes both data and models, and
- Design of the user-machine interface is a critical task in the development of a DSS.



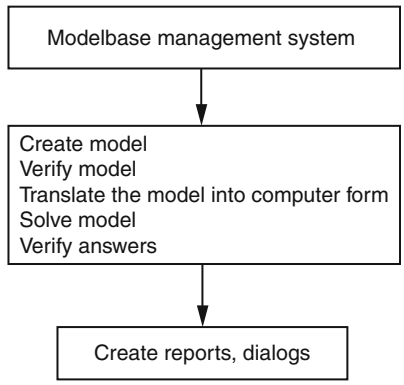
Decision Support Systems (DSS), Fig. 1 Components of a DSS

Figure 1 shows the structure and major components of a DSS. The database holds all the relevant facts of the problem, whether they pertain to the firm or to the environment. The database management system (Fig. 2) takes care of the entry, retrieval, updating, and deletion of data. It also responds to inquiries and generates reports.

The modelbase holds all the models required to work the problem. The modelbase management system (Fig. 3) assists in creating the mathematical model, and in translating the human prepared mathematical model into computer understandable form. The critical process of the modelbase management system is finding the solution to the mathematical model. The system also generates



Decision Support Systems (DSS), Fig. 2 Database management system



Decision Support Systems (DSS), Fig. 3 Modelbase management system

reports and assists in the preparation of computer-human dialogs.

While OR/MS stresses the model, DSS stresses the computer-based database. DSS emphasizes the importance of the user-machine interface, and the design of dialog generation and management software.

Advocates of DSS assert that by combining the power of the human mind and the computer, DSS is capable of enhancing decision making, and that DSS can grapple with problems not subject to the traditional approach of OR/MS.

Note that DSS stresses the role of humans in decision making, and explicitly factors human capabilities into decision making. A decision support system accepts the human as an essential subsystem. DSS does not usually try to optimize in a mathematical sense, and bounded rationality and satisficing provide guidance to the designers of DSS.

Designing Decision Support Systems

The design phases of DSS are quite similar to the phases of the design, implementation, and testing of other systems. It is customary to distinguish six phases, although not all six phases are required for every DSS.

1. During the systems analysis and design phase, existing systems are reviewed and analyzed with the objective of establishing requirements and needs of the new system. Then it is established whether meeting the specifications is feasible from the technical, economical, psychological, and social points of view. Is it possible to overcome the difficulties, and are opportunities commensurate with costs? If the answers are affirmative, meetings with management are held to obtain support. This phase produces a conceptual design and master plan.
2. During the design phase, input, processing, and output requirements are developed and a logical (not physical) design of the system is prepared. After the logical design is completed and found to be acceptable, the design of the hardware and software is undertaken.
3. During the construction and testing phase, the software is completed and tested on the hardware system. Testing includes user participation to assure that the system will be acceptable both from the points of view of the user and management, if they are different.
4. During the implementation phase, the system is retested, debugged, and put into use. To assure final user acceptance, no effort is spared in training and educating users. Management is kept up-to-date on the progress of the project.
5. Operation and maintenance is a continued effort during the life of the DSS. User satisfaction is monitored, errors are uncovered and corrected, and the method of operating the system is fine-tuned.
6. Evaluation and control is a continued effort to assure the viability of the system and the maintenance of management support.

A Forecasting System

Connoisseur Foods is a diversified food company with several autonomous subdivisions and subsidiaries (adapted from Alter 1980, and Turban 1990). Several of the division managers were old-line managers relying on experience and judgment to make major decisions. Top management installed a DSS to provide quantitative help to establish and monitor levels of such marketing

efforts as advertising, pricing, and promotion. The DSS model was based on S-shaped response functions of marketing conditions to such decision functions as advertising. The curves were derived by using both historical data and marketing experts. The databases for the farm products division contained about 20 million data items on sales both in dollars and number of units for 400 items sold in 300 branches.

The DSS assisted management in developing better marketing strategies and more competitive positions. Top management, however, stated that the real benefit of the DSS was not so much the installation of isolated systems and models, but the assimilation of new approaches in corporate decision making.

A Portfolio Management System

The trust division of Great Eastern Bank employed 50 portfolio managers in several departments (adapted from Alter 1980 and Turban 1990). The portfolio managers controlled many small accounts, large pension funds, and provided advice to investors in large accounts. The on-line DSS portfolio management system provided information to the portfolio managers.

The DSS includes lists of stocks from which the portfolio managers could buy stocks, information, and analysis on particular industries. It is basically a data retrieval system that could display portfolios, as well as specific information on securities.

The heart of the system is the database that allowed portfolio managers to generate reports with the following functions:

- Directory by accounts,
- Table to scan accounts,
- Graphic display of breakdown by industry and security for an account,
- Tabular listing of all securities within an account,
- Scatter diagrams between data items,
- Summaries of accounts,
- Distribution of data on securities,
- Evaluation of hypothetical portfolios,
- Performance monitoring of portfolios,
- Warnings if deviations from guidelines occur; and
- Tax implications.

The benefits of the systems were better investment performance, improved information, improved presentation formats, less clerical work, better communication, improved bank image, and enhanced marketing capability.

Concluding Remarks

Advocates of DSS claim that DSS deals with unstructured or semistructured problems, while OR/MS is restricted to structured problems. Few workers in OR/MS would agree.

At the onset, it is frequently the case that a particular business situation is confusing, and, to straighten it out, a problem must be instituted and the problem must be structured. Thus, whether OR/ MS or DSS or both are involved, attempts will be made to structure as much of the situation as possible.

The problem will be structured by OR/MS or DSS to the point that some part of the problem can be taken care of by quantitative methods and computers, and some others are left to human judgment, intuition, and opinion. There may be a degree of difference between OR/MS and DSS: OR/MS may stress optimization, the model base; DSS the database.

Attempts to draw the line between DSS and OR/MS are counterproductive. Those who are dedicated to help management in solving hard problems need to be concerned with any and all theories, practices, and principles that can help. To counsel management in the most productive manner requires that no holds be barred when a task is undertaken.

The principles of DSS are often used without mention in simulation programs. Moreover, as in the spirit of DSS, the user-machine interface is often visual, given the animation capability of modern computers. Thus, managerial decisions may be influenced not only by using traditional quantitative measures, but also by judging customer perceptions.

See

- ▶ Bounded Rationality
- ▶ Choice Theory
- ▶ Decision Analysis
- ▶ Decision Analysis in Practice
- ▶ Decision Problem
- ▶ Information Systems and Database Design in OR/MS
- ▶ Satisficing
- ▶ Soft Systems Methodology

References

Alter, S. L. (1980). *Decision support systems: Current practice and continuing challenges*. Reading, MA: Addison-Wesley.

Bennett, J. L. (1983). *Building decision support systems*. Reading, MA: Addison-Wesley.

Burstein, F., & Holsapple, C. (2008). *Handbook on decision support systems 2: Variations*. New York: Springer.

Holsapple, C., & Whinston, A. (1996). *Decision support systesms: A knowledge-based approach*. Eagan, MN: West Publishing.

Keen, P. G. W., & Morton, S. (1973). *Decision support systems*. Reading, MA: Addison-Wesley.

Pritsker, A. A. B. (1996). Life & death decisions. *OR/ MS Today, 25*(4), 22–28.

Simon, H. A. (1992). Methods and bounds of economics. In Praxiologies and the philosophy of economics. New Brunswick and London: Transaction Publishers.

Turban, E. (1990). *Decision support and expert systems* (2nd ed.). New York: Macmillan.

Decision Trees

Stuart Eriksen[1], Candice H. Huynh[2] and L. Robin Keller[2]

[1]Santa Ana, CA, USA

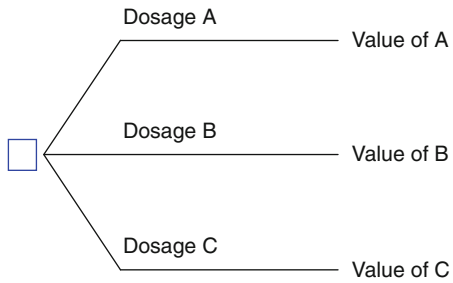
[2]University of California, Irvine, CA, USA

Introduction

A decision tree is a pictorial description of a well-defined decision problem. It is a graphical representation consisting of nodes (where decisions are made or chance events occur) and arcs (which connect nodes). Decision trees are useful because they provide a clear, documentable, and discussible model of either how the decision was made or how it will be made.

The tree provides a framework for the calculation of the expected value of each available alternative. The alternative with the maximum expected value is the best choice path based on the information and mind-set of the decision makers at the time the decision is made. This best choice path indicates the best overall alternative, including the best subsidiary decisions at future decision steps, when uncertainties have been resolved.

The decision tree should be arranged, for convenience, from left to right in the temporal order in which the events and decisions will occur. Therefore, the steps on the left occur earlier in time than those on the right.



Decision Trees, Fig. 1 The choice of drug dosage

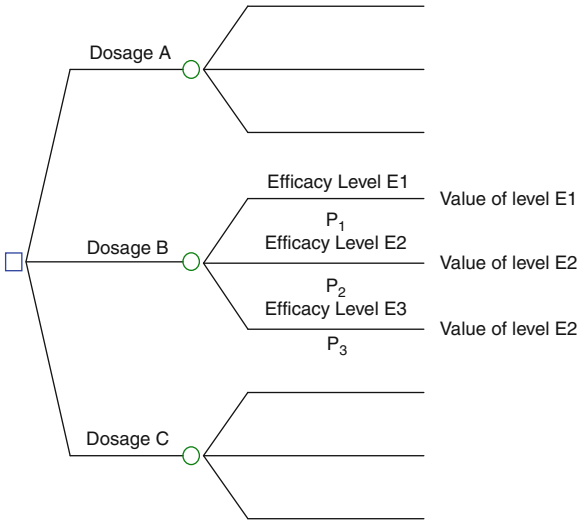
Decision Nodes

Steps in the decision process involving decisions between several choice alternatives are indicated by decision nodes, drawn as square boxes. Each available choice is shown as one arc (or path) leading away from its decision node toward the right. When a planned decision has been made at such a node, the result of that decision is recorded by drawing an arrow in the box pointing toward the chosen option. As an example of the process, consider a pharmaceutical company president's choice of which drug dosage to market. The basic dosage choice decision tree is shown in Fig. 1. Note that the values of the eventual outcomes (on the far right) will be expressed as some measure of value to the eventual user (for example, the patient or the physician).

Chance Nodes

Steps in the process which involve uncertainties are indicated by circles (called chance nodes), and the possible outcomes of these probabilistic events are again shown as arcs or paths leading away from the node toward the right. The results of these uncertain factors are out of the hands of the decision maker; chance or some other group or person (uncontrolled by the decision maker) will determine the outcome of this node. Each of the potential outcomes of a chance node is labeled with its probability of occurrence.

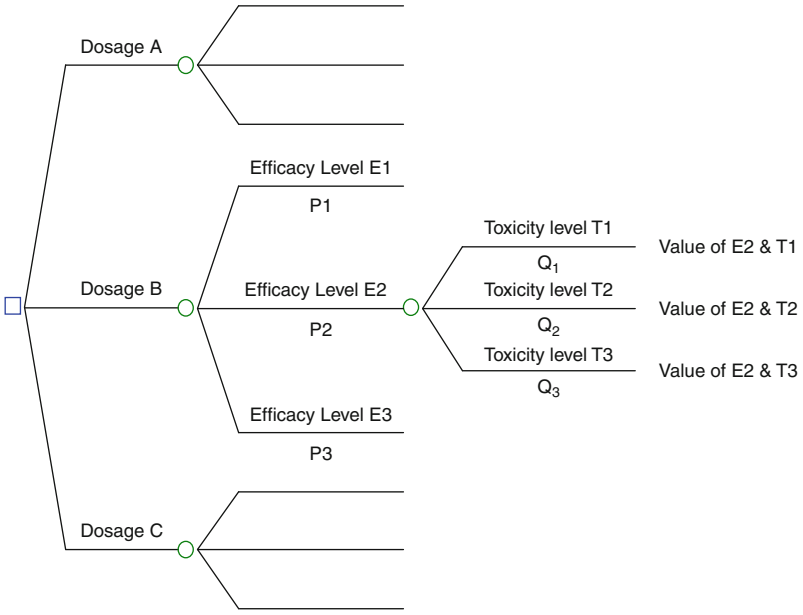
All possible outcomes must be indicated, so the sum of the potential outcome probabilities of a chance node must equal 1.0. Using the drug dose selection problem noted above, the best choice of dose depends on at least one probabilistic event: the level of performance of the drug in clinical trials, which is a proxy measure of the efficacy of the drug. A simplified decision tree for that part of the firm's decision is shown in Fig. 2. Note that each dosage choice has a subsequent efficacy chance node similar to the one shown, so the expanded tree would have nine outcomes. The probabilities (p_1, p_2, and p_3) associated with the outcomes are expected to differ for each dosage.



Decision Trees, Fig. 2 The choice of drug dosage based on efficacy outcome

There are often several nodes in a decision tree; in the case of the drug dosage decision, the decision will also depend on the toxicity as demonstrated by both animal study data and human toxicity study data, as well as on the efficacy data. The basic structure of this more complex decision is shown in Fig. 3. The completely expanded tree has 27 eventual outcomes and associated values. Notice that although not always the case, here the probabilities (q_1, q_2, and q_3) of the toxicity levels are independent of the efficacy level.

One use of a decision tree is to clearly display the factors and assumptions involved in a decision. If the decision outcomes are quantified and the probabilities of chance events are specified, the tree can also be analyzed by calculating the expected value of each alternative. If several decisions are involved in the problem being considered, the strategy best suited to each specific set of chance outcomes can be planned in advance.



Decision Trees, Fig. 3 The choice of dosage based on uncertain efficacy and toxicity

Probabilities

Estimates of the probabilities for each of the outcomes of the chance nodes must be made. In the simplified case of the drug dose decision above, the later chance node outcome probabilities are modeled as being independent of the earlier chance nodes. While not intuitively obvious, careful thought should show that the physiological factors involved in clinical efficacy must be different from those involved in toxicity, even if the drug is being used to treat that toxicity. Therefore, with most drugs, the probability of high human toxicity is likely independent of the level of human efficacy. In the more general non-drug situations, however, for sequential steps, the latter probabilities are often dependent conditional probabilities, since their value depends on the earlier chance outcomes.

For example, consider the problem in Fig. 4, where the outcome being used for the drug dose decision is based on the eventual sales of it. The values of the eventual outcomes now are expressed as sales for the firm.

The probability of high sales depends on the efficacy as well as on the toxicity, so the dependent conditional probability of high sales is the probability of high sales given that the efficacy is level 2 and toxicity is level 2, which can be written as $p_{(\text{High Sales}|\text{E2\&T2})}$.

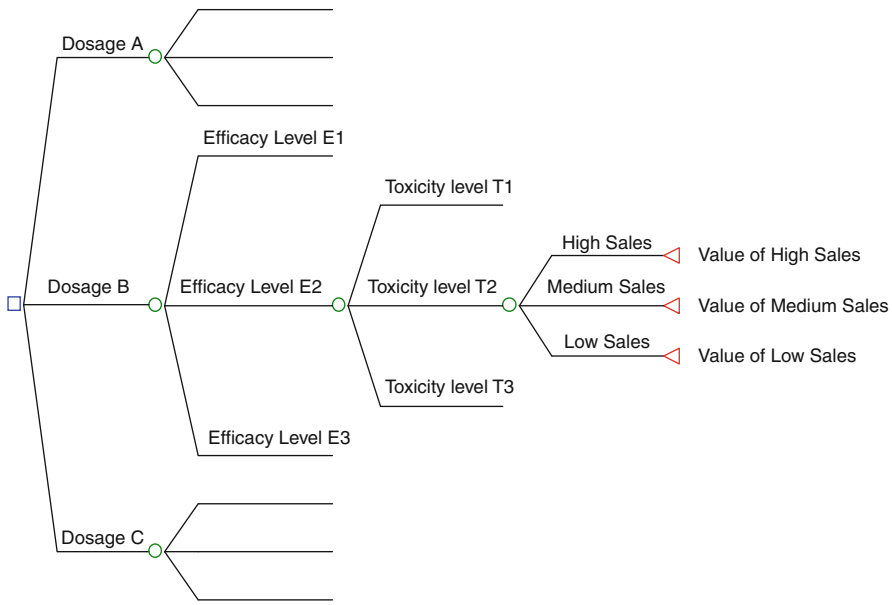
Outcome Measures

At the far right of the tree, the possible outcomes are listed at the end of each branch. To calculate numerical expected values for alternative choices, outcomes must be measured numerically and often monetary measures will be used. More generally, the utility of the outcomes can be calculated. Single or multiple attribute utility functions have been elicited in many decision situations to represent decision makers' preferences for different outcomes on a numerical (although not monetary) scale.

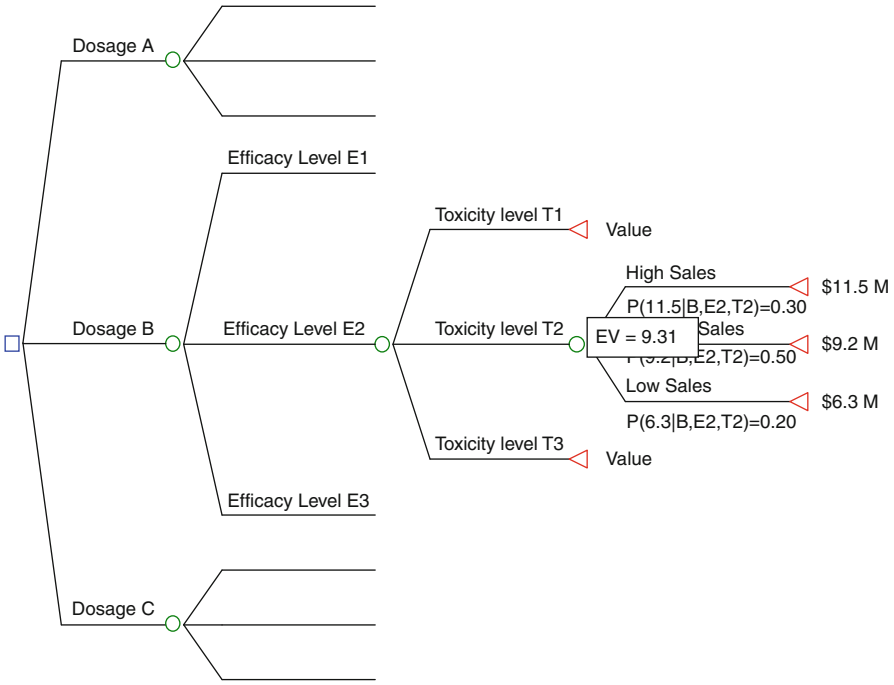
The Tree as an Aid in Decision Making

The decision tree analysis method is called fold-back and prune. Beginning at a far right chance node of the tree, the expected value of the outcome measure is calculated and recorded for each chance node by summing, over all the outcomes, the product of the probability of the outcome times the measured value of the outcome. Figure 5 shows this calculation for the first step in the analysis of the drug-dose decision tree.

This step is called folding back the tree since the branches emanating from the chance node are folded



Decision Trees, Fig. 4 The choice of dosage based on efficacy and toxicity and their eventual effect on sales

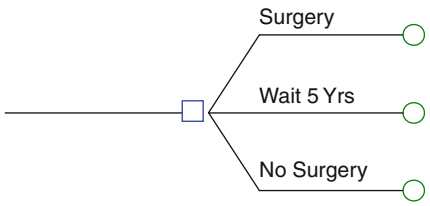


Decision Trees, Fig. 5 The first step, calculating the expected value of the chance node for sales: EV = 0.3(11.5) + 0.5(9.2) + 0.2(6.3) = 9.31

up or collapsed, so that the chance node is now represented by its expected value. This is continued until all the chance nodes on the far right have been evaluated. These expected values then become the values for the outcomes of the chance or decision nodes further to the left in the diagram. At a decision node, the best of the alternatives is the one with the maximum expected value, which is then recorded by drawing an arrow towards that choice in the decision node box and writing down the expected value associated with the chosen option. This is referred to as pruning the tree, as the less valuable choices are eliminated from further consideration. The process continues from right to left, by calculating the expected value at each chance node and pruning at each decision node. Finally the best choice for the overall decision is found when the last decision node at the far left has been evaluated.

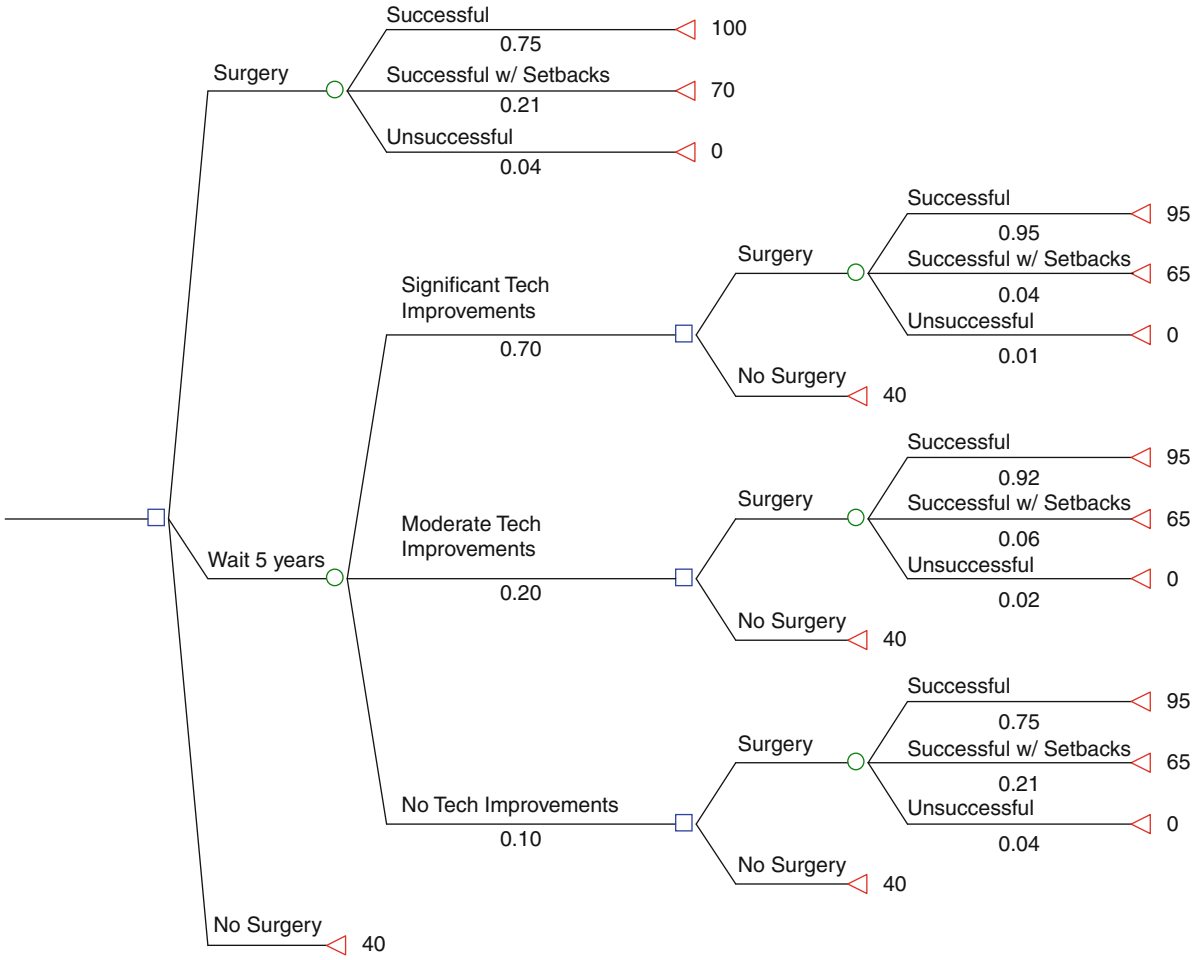
Example

In this example, a decision faced by a patient who is considering laser eye surgery to improve her vision will be considered. The basic decision process is shown in Fig. 6. The initial decision a patient encounters is whether to: have the surgery, wait for more technological advances, or not have the surgery at all.



Decision Trees, Fig. 6 The initial decision point

Suppose that if a patient chooses to wait at the first decision node, she will observe the outcome of possible technological advances at the first chance node, and then will have to make the decision of whether to have the surgery or not. Figure 7 shows a detailed decision tree of this patient's decision process. The entries at the end of the branches can be seen as a measure of health utility to the patient, on a 0-100 scale, where 100 is the best level of health utility. Other patients can customize this tree to their personal circumstances using a combination of chance and decision nodes.



Decision Trees, Fig. 7 Complete mapping of the decision process of whether or not to have lasik surgery

Following the method of folding back the tree, the expected health utility of having the surgery immediately is 89.70, waiting 5 years is 91.74, and not having the surgery at all is 40.00, where the calculation of each chance node is the expected health utility. And so waiting 5 years is the optimal decision for the patient in this example.

See

- ▶ Bayesian Decision Theory, Subjective Probability, and Utility
- ▶ Decision Analysis
- ▶ Decision Analysis in Practice
- ▶ Decision Making and Decision Analysis
- ▶ Multi-attribute Utility Theory
- ▶ Preference Theory
- ▶ Utility Theory

References

Clemen, R., & Reilly, T. (2004). *Making hard decisions with decision tools*. Belmont, CA: Duxbury Press.

Eriksen, S. P., & Keller, L. R. (1993). A multi-attribute approach to weighing the risks and benefits of pharmaceutical agents. *Medical Decision Making, 13*, 118–125.

Keeney, R. L., & Raiffa, H. (1976). *Decisions with multiple objectives: Preferences and value tradeoffs*. Wiley, New York.

Kirkwood, C. (1997). *Strategic decision making: Multiobjective decision analysis with spreadsheets*. Belmont, CA: Duxbury Press.

Raiffa, H. (1968). *Decision analysis*. Reading, MA: Addison-Wesley.

Decision Variables

The variables in a given model that are subject to manipulation by the specified decision rule.

See

► Controllable Variables

Decomposition Algorithms

► Benders Decomposition Method
► Block-Angular System
► Dantzig-Wolfe Decomposition Algorithm
► Large-Scale Systems

References

Dantzig, G. B., & Thapa, M. N. (2003). *Linear programming 2: Theory and extensions*. New York: Springer.

Deep Uncertainty

Warren E. Walker[1], Robert J. Lempert[2] and Jan H. Kwakkel[1]
[1]Delft University of Technology, Delft, The Netherlands
[2]RAND Corporation, Santa Monica, CA, USA

Introduction

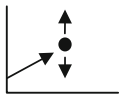
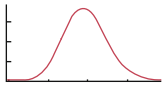
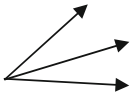
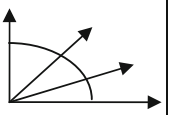
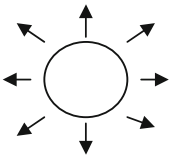
The notion of uncertainty has taken different meanings and emphases in various fields, including the physical sciences, engineering, statistics, economics, finance, insurance, philosophy, and psychology. Analyzing the notion in each discipline can provide a specific historical context and scope in terms of problem domain, relevant theory, methods, and tools for handling uncertainty. Such analyses are given by Agusdinata (2008), van Asselt (2000), Morgan and Henrion (1990), and Smithson (1989).

In general, uncertainty can be defined as limited knowledge about future, past, or current events. With respect to policy making, the extent of uncertainty clearly involves subjectivity, since it is related to the satisfaction with existing knowledge, which is colored by the underlying values and perspectives of the policymaker and the various actors involved in the policy-making process, and the decision options available to them.

Shannon (1948) formalized the relationship between the uncertainty about an event and information in "A Mathematical Theory of Communication." He defined a concept he called entropy as a measure of the average information content associated with a random outcome. Roughly speaking, the concept of entropy in information theory describes how much information there is in a signal or event and relates this to the degree of uncertainty about a given event having some probability distribution.

Uncertainty is not simply the absence of knowledge. Funtowicz and Ravetz (1990) describe uncertainty as a situation of inadequate information, which can be of three sorts: inexactness, unreliability, and border with ignorance. However, uncertainty can prevail in situations in which ample information is available (Van Asselt and Rotmans 2002). Furthermore, new information can either decrease or increase uncertainty. New knowledge on complex processes may reveal the presence of uncertainties that were previously unknown or were understated. In this way, more knowledge illuminates that one's understanding is more limited or that the processes are more complex than previously thought (van der Sluijs 1997).

Uncertainty as inadequacy of knowledge has a very long history, dating back to philosophical questions debated among the ancient Greeks about the certainty of knowledge, and perhaps even further. Its modern history begins around 1921, when Knight made a distinction between risk and uncertainty (Knight 1921). According to Knight, risk denotes the calculable and thus controllable part of all that is unknowable. The remainder is the uncertain – incalculable and uncontrollable. Luce and Raiffa (1957) adopted these labels to distinguish between decision making under risk and decision making under uncertainty. Similarly, Quade (1989) makes a distinction between stochastic uncertainty and real uncertainty. According to Quade, stochastic

LOCATION	LEVEL	Complete Certainty	Level 1	Level 2	Level 3	Level 4	Level 5	Total ignorance
	Context		A clear enough future 	Alternate futures (with probabilities) 	Alternate futures with ranking 	A multiplicity of plausible futures 	An unknown future 	
	System model		A single (deterministic) system model	A single (stochastic) system model	Several system models, one of which is most likely	Several system models, with different structures	Unknown system model; know we don't know	
	System outcomes		A point estimate for each outcome	A confidence interval for each outcome	Several sets of point estimates, ranked according to their perceived likelihood	A known range of outcomes	Unknown outcomes; know we don't know	
	Weights on outcomes		A single set of weights	Several sets of weights, with a probability attached to each set	Several sets of weights, ranked according to their perceived likelihood	A known range of weights	Unknown weights; know we don't know	

Deep Uncertainty, Fig. 1 The progressive transition of levels of uncertainty from complete certainty to total ignorance

uncertainty includes frequency-based probabilities and subjective (Bayesian) probabilities. Real uncertainty covers the future state of the world and the uncertainty resulting from the strategic behavior of other actors. Often, attempts to express the degree of certainty and uncertainty have been linked to whether or not to use probabilities, as exemplified by Morgan and Henrion (1990), who make a distinction between uncertainties that can be treated through probabilities and uncertainties that cannot. Uncertainties that cannot be treated probabilistically include model structure uncertainty and situations in which experts cannot agree upon the probabilities. These are the more important and hardest to handle types of uncertainties (Morgan 2003). As Quade (1989, p. 160) wrote: "Stochastic uncertainties are therefore among the least of our worries; their effects are swamped by uncertainties about the state of the world and human factors for which we know absolutely nothing about probability distributions and little more about the possible outcomes." These kinds of uncertainties are now referred to as deep uncertainty (Lempert et al. 2003), or severe uncertainty (Ben-Haim 2006).

Levels of Uncertainty

Walker et al. (2003) define uncertainty to be "any departure from the (unachievable) ideal of complete determinism."

For purposes of determining ways of dealing with uncertainty in developing public policies or business strategies, one can distinguish two extreme levels of uncertainty—complete certainty and total ignorance—and five intermediate levels (e.g. Courtney 2001; Walker et al. 2003; Makridakis et al. 2009; Kwakkel et al. 2010d). In Fig. 1, the five levels are defined with respect to the knowledge assumed about the various aspects of a policy problem: (a) the future world, (b) the model of the relevant system for that future world, (c) the outcomes from the system, and (d) the weights that the various stakeholders will put on the outcomes. The levels of uncertainty are briefly discussed below.

Complete certainty is the situation in which everything is known precisely. It is not attainable, but acts as a limiting characteristic at one end of the spectrum.

Level 1 uncertainty (A clear enough future) represents the situation in which one admits that one is not absolutely certain, but one is not willing or able to measure the degree of uncertainty in any explicit way (Hillier and Lieberman 2001, p. 43). Level 1 uncertainty is often treated through a simple sensitivity analysis of model parameters, where the impacts of small perturbations of model input parameters on the outcomes of a model are assessed.

Level 2 uncertainty (Alternate futures with probabilities) is any uncertainty that can be described adequately in statistical terms. In the case of uncertainty about the future, Level 2 uncertainty is often captured in the form of either a (single) forecast (usually trend based) with a confidence interval or multiple forecasts (scenarios) with associated probabilities.

Level 3 uncertainty (Alternate futures with ranking) represents the situation in which one is able to enumerate multiple alternatives and is able to rank the alternatives in terms of perceived likelihood. That is, in light of the available knowledge and information there are several different parameterizations of the system model, alternative sets of outcomes, and/or different conceivable sets of weights. These possibilities can be ranked according to their perceived likelihood (e.g. virtually certain, very likely, likely, etc.). In the case of uncertainty about the future, Level 3 uncertainty about the future world is often captured in the form of a few trend-based scenarios based on alternative assumptions about the driving forces (e.g., three trend-based scenarios for air transport demand, based on three different assumptions about GDP growth). The scenarios are then ranked according to their perceived likelihood, but no probabilities are assigned, see Patt and Schrag (2003) and Patt and Dessai (2004).

Level 4 uncertainty (Multiplicity of futures) represents the situation in which one is able to enumerate multiple plausible alternatives without being able to rank the alternatives in terms of perceived likelihood. This inability can be due to a lack of knowledge or data about the mechanism or functional relationships being studied; but this inability can also arise due to the fact that the decision makers cannot agree on the rankings. As a result, analysts struggle to specify the appropriate models to describe interactions among the system's variables, to select the probability distributions to represent uncertainty about key parameters in the models, and/or how to value the desirability of alternative outcomes (Lempert et al. 2003).

Level 5 uncertainty (Unknown future) represents the deepest level of recognized uncertainty; in this case, what is known is only that we do not know. This ignorance is recognized. Recognized ignorance is increasingly becoming a common feature of life, because catastrophic, unpredicted, surprising, but painful events seem to be occurring more often. Taleb (2007) calls these events "Black Swans." He defines a Black Swan event as one that lies outside the realm of regular expectations (i.e., "nothing in the past can convincingly point to its possibility"), carries an extreme impact, and is explainable only after the fact (i.e., through retrospective, not prospective, predictability). One of the most dramatic recent Black Swans is the concatenation of events following the 2007 subprime mortgage crisis in the U.S. The mortgage crisis (which some had forecast) led to a credit crunch, which led to bank failures, which led to a deep global recession in 2009, which was outside the realm of most expectations. Another recent Black Swan was the level 9.0 earthquake in Japan in 2011, which led to a tsunami and a nuclear catastrophe, which led to supply chain disruptions (e.g., for automobile parts) around the world.

Total ignorance is the other extreme on the scale of uncertainty. As with complete certainty, total ignorance acts as a limiting case.

Lempert et al. (2003) have defined deep uncertainty as "the condition in which analysts do not know or the parties to a decision cannot agree upon (1) the appropriate models to describe interactions among a system's variables, (2) the probability distributions to represent uncertainty about key parameters in the models, and/or (3) how to value the desirability of alternative outcomes. They use the language 'do not know' and 'do not agree upon' to refer to individual and group decision making, respectively. This article includes both individual and group decision making in all five of the levels, referring to Level 4 and Level 5 uncertainties as 'deep uncertainty', and assigning the 'do not know' portion of the definition to Level 5 uncertainties and the 'cannot agree upon' portion of the definition to Level 4 uncertainties.

Decision Making Under Deep Uncertainty

There are many quantitative analytical approaches to deal with Level 1, Level 2, and Level 3 uncertainties. In fact, most of the traditional applied scientific work in the engineering, social, and natural sciences has been built upon the supposition that the uncertainties result from either a lack of information, which "has led to an emphasis on uncertainty reduction through ever-increasing information seeking and processing" (McDaniel and Driebe 2005), or from random variation, which has concentrated efforts on stochastic processes and statistical analysis. However, most of the important policy problems faced by policymakers are characterized by the higher levels of uncertainty, which cannot be dealt with through the use of probabilities and cannot be reduced by gathering more information, but are basically unknowable and unpredictable at the present time. And these high levels of uncertainty can involve uncertainties about all aspects of a policy problem — external or internal developments, the appropriate (future) system model, the parameterization of the model, the model outcomes, and the valuation of the outcomes by (future) stakeholders.

For centuries, people have used many methods to grapple with the uncertainty shrouding the long-term future, each with its own particular strengths. Literary narratives, generally created by one or a few individuals, have an unparalleled ability to capture people's imagination. More recently, group processes, such as the Delphi technique (Quade 1989), have helped large groups of experts combine their expertise into narratives of the future. Statistical and computer simulation modeling helps capture quantitative information about the extrapolation of current trends and the implications of new driving forces. Formal decision analysis helps to systematically assess the consequences of such information. Scenario-based planning helps individuals and groups accept the fundamental uncertainty surrounding the long-term future and consider a range of potential paths, including those that may be inconvenient or disturbing for organizational, ideological, or political reasons.

Despite this rich legacy, these traditional methods all founder on the same shoals: an inability to grapple with the long term's multiplicity of plausible futures. Any single guess about the future will likely prove wrong. Policies optimized for a most likely future may fail in the face of surprise. Even analyzing a well-crafted handful of scenarios will miss most of the future's richness and provides no systematic means to examine their implications. This is particularly true for methods based on detailed models. Such models that look sufficiently far into the future should raise troubling questions in the minds of both the model builders and the consumers of model output. Yet the root of the problem lies not in the models themselves, but in the way in which models are used. Too often, analysts ask what will happen, thus trapping themselves in a losing game of prediction, instead of the question they really would like to have answered: Given that one cannot predict, which actions available today are likely to serve best in the future?

Broadly speaking, although there are differences in definitions, and ambiguities in meanings, the literature offers four (overlapping, not mutually exclusive) ways for dealing with deep uncertainty in making policies, see van Drunen et al. (2009).

Resistance: plan for the worst conceivable case or future situation,

- *Resilience*: whatever happens in the future, make sure that you have a policy that will result in the system recovering quickly,
- *Static robustness*: implement a (static) policy that will perform reasonably well in practically all conceivable situations,
- *Adaptive robustness*: prepare to change the policy, in case conditions change.

The first approach is likely to be very costly and might not produce a policy that works well because of Black Swans. The second approach accepts short-term pain (negative system performance), but focuses on recovery.

The third and fourth approaches do not use models to produce forecasts. Instead of determining the best predictive model and solving for the policy that is optimal (but fragilely dependent on assumptions), in the face of deep uncertainty it may be wiser to seek among the alternatives those actions that are most robust — that achieve a given level of goodness across the myriad models and assumptions consistent with known facts (Rosenhead and Mingers 2001). This is the heart of any robust decision method. A robust policy is defined to be one that yields outcomes that are deemed to be satisfactory according to some selected

assessment criteria across a wide range of future plausible states of the world. This is in contrast to an optimal policy that may achieve the best results among all possible plans but carries no guarantee of doing so beyond a narrowly defined set of circumstances. An analytical policy based on the concept of robustness is also closer to the actual policy reasoning process employed by senior planners and executive decision makers. As shown by Lempert and Collins (2007), analytic approaches that seek robust strategies are often appropriate both when uncertainty is deep and a rich array of options is available to decision makers.

Identifying static robust policies requires reversing the usual approach to uncertainty. Rather than seeking to characterize uncertainties in terms of probabilities, a task rendered impossible by definition for Level 4 and Level 5 uncertainties, one can instead explore how different assumptions about the future values of these uncertain variables would affect the decisions actually being faced. Scenario planning is one approach to identifying static robust policies, see van der Heijden (1996). This approach assumes that, although the likelihood of the future worlds is unknown, a range of plausible futures can be specified well enough to identify a (static) policy that will produce acceptable outcomes in most of them. It works best when dealing with Level 4 uncertainties. Another approach is to ask what one would need to believe was true to discard one possible policy in favor of another. This is the essence of Exploratory Modeling and Analysis (EMA).

Long-term robust policies for dealing with Level 5 uncertainties will generally be dynamic adaptive policies—policies that can adapt to changing conditions over time. A dynamic adaptive policy is developed with an awareness of the range of plausible futures that lie ahead, is designed to be changed over time as new information becomes available, and leverages autonomous response to surprise. Eriksson and Weber (2008) call this approach to dealing with deep uncertainty Adaptive Foresight. Walker et al. (2001) have specified a generic, structured approach for developing dynamic adaptive policies for practically any policy domain. This approach allows implementation to begin prior to the resolution of all major uncertainties, with the policy being adapted over time based on new knowledge. It is a way to proceed with the implementation of long-term policies despite the presence of uncertainties. The adaptive policy approach makes dynamic adaptation explicit at the outset of policy formulation. Thus, the inevitable policy changes become part of a larger, recognized process and are not forced to be made repeatedly on an ad hoc basis. Under this approach, significant changes in the system would be based on an analytic and deliberative effort that first clarifies system goals, and then identifies policies designed to achieve those goals and ways of modifying those policies as conditions change. Within the adaptive policy framework, individual actors would carry out their activities as they would under normal policy conditions. But policymakers and stakeholders, through monitoring and corrective actions, would try to keep the system headed toward the original goals. McCray et al. (2010) describe it succinctly as keeping policy "yoked to an evolving knowledge base." Lempert et al. (2003, 2006) propose an approach called Robust Decision Making (RDM), which conducts a vulnerability and response option analysis using EMA to identify and compare (static or dynamic) robust policies. Walker et al. (2001) propose a similar approach for developing adaptive policies, called Dynamic Adaptive Policymaking (DAP).

Some Applications of Robust Decision Making (RDM) and Dynamic Adaptive Policymaking (DAP)

RDM has been applied in a wide range of decision applications, including the development of both static and adaptive policies. The study of Dixon et al. (2007) evaluated alternative (static) policies considered by the U.S. Congress while debating reauthorization of the Terrorism Risk Insurance Act (TRIA). TRIA provides a federal guarantee to compensate insurers for losses due to very large terrorist attacks in return for insurers providing insurance against attacks of all sizes. Congress was particularly interested in the cost to taxpayers of alternative versions of the program. The RDM analysis used a simulation model to project these costs for various TRIA options for each of several thousand cases, each representing a different combination of 17 deeply uncertain assumptions about the type of terrorist attack, the factors influencing the pre-attack distribution of insurance coverage, and any post-attack compensation

decisions by the U.S. Federal government. The RDM analysis demonstrated that the expected cost to taxpayers of the existing TRIA program would prove the same or less than any of the proposed alternatives except under two conditions: the probability of a large terrorist attack (greater than $40 billion in losses) significantly exceeded current estimates and future Congresses did not compensate uninsured property owners in the aftermath of any such attack. This RDM analysis appeared to help resolve a divisive Congressional debate by suggesting that the existing (static) TRIA program was robust over a wide range of assumptions, except for a combination that many policymakers regarded as unlikely. The analysis demonstrates two important features of RDM: (1) its ability to systematically include imprecise probabilistic information (in this case, estimates of the likelihood of a large terrorist attack) in a formal decision analysis, and (2) its ability to incorporate very different types of uncertain information (in this case, quantitative estimates of attack likelihood and qualitative judgments about the propensity of future Congresses to compensate the uninsured).

RDM has also been used to develop adaptive policies, including policies to address climate change (Lempert et al. 1996), economic policy (Seong et al. 2005), complex systems (Lempert 2002), and health policy (Lakdawalla et al. 2009). An example that illustrates RDM's ability to support practical adaptive policy making is discussed in Groves et al. (2008) and Lempert and Groves (2010). In 2005, Southern California's Inland Empire Utilities Agency (IEUA), that supplies water to a fast growing population in an arid region, completed a legally mandated (static) plan for ensuring reliable water supplies for the next twenty-five years. This plan did not, however, consider the potential impacts of future climate change. An RDM analysis used a simulation model to project the present value cost of implementing IEUA's current plans, including any penalties for future shortages, in several hundred cases contingent on a wide range of assumptions about six parameters representing climate impacts, IEUA's ability to implement its plan, and the availability of imported water. A scenario discovery analysis identified three key factors — an 8% or larger decrease in precipitation, any drop larger than 4% in the rain captured as groundwater, and meeting or missing the plan's specific goals for recycled waste water — that, if they occurred simultaneously, would cause IEUA's overall plan to fail (defined as producing costs exceeding by 20% or more those envisioned in the baseline plan). Having identified this vulnerability of IEUA's current plan, the RDM analysis allowed the agency managers to identify and evaluate alternative adaptive plans, each of which combined near-term actions, monitoring of key supply and demand indicators in the region, and taking specific additional actions if certain indicators were observed. The analysis suggested that IEUA could eliminate most of its vulnerabilities by committing to updating its plan over time and by making relative low-cost near-term enhancements in two current programs. Overall, the analysis allowed IEUA's managers, constituents, and elected officials, who did not all agree on the likelihood of climate impacts, to understand in detail vulnerabilities to their original plan and to identify and reach consensus on adaptive plans that could ameliorate those vulnerabilities.

An example of DAP comes from the field of airport strategic planning. Airports increasingly operate in a privatized and liberalized environment. Moreover, this change in regulations has changed the public's perception of the air transport sector. As a result of this privatization and liberalization, the air transport industry has undergone unprecedented changes, exemplified by the rise of airline alliances and low cost carriers, an increasing environmental awareness, and, since 9/11, increased safety and security concerns. These developments pose a major challenge for airports. They have to make investment decisions that will shape the future of the airport for many years to come, taking into consideration the many uncertainties that are present. DAP has been put forward as a way to plan the long-term development of an airport under these conditions (Kwakkel et al. 2010a). As an illustration, a case based on the current challenges of Amsterdam Airport Schiphol has been pursued. Using a simulation model that calculates key airport performance metrics such as capacity, noise, and external safety, the performance of an adaptive policy and a competing traditional policy across a wide range of uncertainties was explored. This comparison revealed that the traditional plan would have preferable performance only in the narrow bandwidth of future developments for which it was optimized. Outside this bandwidth, the adaptive policy had superior performance. The analysis further revealed

that the range of expected outcomes for the adaptive policy is significantly smaller than for the traditional policy. That is, an adaptive policy will reduce the uncertainty about the expected outcomes, despite various deep uncertainties about the future. This analysis strongly suggested that airports operating in an ever increasing uncertain environment could significantly improve the adequacy of their long-term development if they planned for adaptation (Kwakkel et al. 2010b, 2010c).

Another policy area to which DAP has been applied is the expansion of the port of Rotterdam. This expansion is very costly and the additional land and facilities need to match well with market demand as it evolves over the coming 30 years or more. DAP was used to modify the existing plan so that it can cope with a wide range of uncertainties. To do so, adaptive policy making was combined with Assumption-Based Planning (Dewar 2002). This combination resulted in the identification of the most important assumptions underlying the current plan. Through the adaptive policy making framework, these assumptions were categorized and actions for improving the likelihood that the assumptions will hold were specified (Taneja et al. 2010).

Various other areas of application of DAP have also been explored, including flood risk management in the Netherlands in light of climate change (Rahman et al. 2008), policies with respect to the implementation of innovative urban transport infrastructures (Marchau et al. 2008), congestion road pricing (Marchau et al. 2010), intelligent speed adaptation (Agusdinata et al. 2007), and magnetically levitated (Maglev) rail transport (Marchau et al. 2010).

See

► Exploratory Modeling and Analysis

References

Agusdinata, D. B. (2008). *Exploratory modeling and analysis: A promising method to deal with deep uncertainty*. Ph.D. dissertation, Delft University of Technology, The Netherlands.

Agusdinata, D. B., Marchau, V. A. W. J., & Walker, W. E. (2007). Adaptive policy approach to implementing intelligent speed adaptation. *IET Intelligent Transport Systems (ITS), 1*(3), 186–198.

Ben-Haim, Y. (2006). *Information-gap decision theory: Decisions under severe uncertainty* (2nd ed.). New York: Wiley.

Courtney, H. (2001). *20/20 foresight: Crafting strategy in an uncertain world*. Boston: Harvard Business School Press.

Dewar, J. A. (2002). *Assumption-based planning: A tool for reducing avoidable surprises*. Cambridge, UK: Cambridge University Press.

Dixon, L., Lempert, R.J., LaTourrette, T., & Reville, R.T. (2007). *The Federal role in terrorism insurance: Evaluating alternatives in an uncertain world (MG-679-CTRMP)*. Santa Monica, CA: RAND.

Eriksson, E. A., & Weber, K. M. (2008). Adaptive foresight: Navigating the complex landscape of policy strategies. *Technological Forecasting and Social Change, 75*, 462–482.

Funtowicz, S. O., & Ravetz, J. R. (1990). *Uncertainty and quality in science for policy*. Dordrecht, The Netherlands: Kluwer.

Groves, D. G., Davis, M., Wilkinson, R., Lempert, R. (2008). Planning for climate change in the inland empire: Southern California. *Water Resources IMPACT*, July 2008.

Hillier, F. S., & Lieberman, G. J. (2001). *Introduction to operations research*. New York: McGraw Hill.

Knight, F. H. (1921). *Risk, uncertainty and profit*. New York: Houghton Mifflin Company (republished in 2006 by Dover Publications, Mineola, NY).

Kwakkel, J. H., Walker, W. E., & Marchau, V. A. W. J. (2010a). Adaptive airport strategic planning. *European Journal of Transport and Infrastructure Research, 10*(3), 249–273.

Kwakkel, J. H., Walker, W. E., & Marchau, V. A. W. J. (2010b). From predictive modeling to exploratory modeling: How to use non-predictive models for decision-making under deep uncertainty. *25th Mini-EURO Conference on Uncertainty and Robustness in Planning and Decision Making (URPDM 2010)*, Coimbra, Portugal, 15–17 April 2010.

Kwakkel, J. H., Walker, W. E., & Marchau, V. A. W. J. (2010c). Assessing the efficacy of adaptive airport strategic planning: Results from computational experiments. *World Conference on Transport Research*, Porto, Portugal, 11–15 July 2010.

Kwakkel, J. H., Walker, W. E., & Marchau, V. A. W. J. (2010d). Classifying and communicating uncertainties in model-based policy analysis. *International Journal of Technology, Policy and Management, 10*(4), 299–315.

Lakdawalla, D. N., Goldman, D. P., Michaud, P.-C., Sood, N., Lempert, R., Cong, Z., de Vries, H., & Gutlerrez, I. (2009). US pharmaceutical policy in a global marketplace. *Health Affairs, 28*, 138–150.

Lempert, R. J. (2002, May 14). A new decision sciences for complex systems. *Proceedings of the National Academy of Sciences*, *99*(Suppl. 3), 7309–7313.

Lempert, R. J., & Collins, M. T. (2007). Managing the risk of uncertain threshold response: Comparison of robust, optimum, and precautionary approaches. *Risk Analysis, 27*(4), 1009–1026.

Lempert, R. J., & Groves, D. G. (2010). Identifying and evaluating robust adaptive policy responses to climate change for water management agencies in the American west. *Technological Forecasting and Social Change, 77*, 960–974.

Lempert, R. J., Groves, D. G., Popper, S. W., & Bankes, S. C. (2006). A general, analytic method for generating robust strategies and narrative scenarios. *Management Science, 52*(4), 514–528.

Lempert, R. J., Popper, S. W., & Bankes, S. C. (2003). *Shaping the next one hundred years: New methods for quantitative long-term strategy analysis (MR-1626-RPC)*. Santa Monica, CA: The RAND Pardee Center.

Lempert, R. J., Schlesinger, M. E., & Bankes, S. C. (1996). When we don't know the costs or the benefits: Adaptive strategies for abating climate change. *Climatic Change, 33*, 235–274.

Luce, R. D., & Raiffa, H. (1957). *Games and decisions*. New York: Wiley.

Makridakis, S., Hogarth, R. M., & Gaba, A. (2009). Forecasting and uncertainty in the economic and business world. *International Journal of Forecasting, 25*, 794–812.

Marchau, V., Walker, W., & van Duin, R. (2008). An adaptive approach to implementing innovative urban transport solutions. *Transport Policy, 15*(6), 405–412.

Marchau, V. A. W. J., Walker, W. E., & van Wee, G. P. (2010). Dynamic adaptive transport policies for handling deep uncertainty. *Technological Forecasting and Social Change, 77*(6), 940–950.

McCray, L. E., Oye, K. A., & Petersen, A. C. (2010). Planned adaptation in risk regulation: An initial survey of US environmental, health, and safety regulation. *Technological Forecasting and Social Change, 77*, 951–959.

McDaniel, R. R., & Driebe, D. J. (Eds.). (2005). *Uncertainty and surprise in complex systems: Questions on working with the unexpected*. Springer.

Morgan, M. G. (2003). Characterizing and dealing with uncertainty: Insights from the integrated assessment of climate change. *The Integrated Assessment Journal, 4*(1), 46–55.

Morgan, M. G., & Henrion, M. (1990). *Uncertainty: A guide to dealing with uncertainty in quantitative risk and policy analysis*. Cambridge, UK: Cambridge University Press.

Patt, A. G., & Dessai, S. (2004). Communicating uncertainty: Lessons learned and suggestions for climate change assessment. *Comptes Rendu Geosciences, 337*, 425–441.

Patt, A. G., & Schrag, D. (2003). Using specific language to describe risk and probability. *Climatic Change, 61*, 17–30.

Popper, S. W., Griffin, J., Berrebi, C., Light, T., & Min, E. Y. (2009). *Natural gas and Israel's energy future: A strategic analysis under conditions of deep uncertainty* (TR-747-YSNFF). Santa Monica, CA: RAND.

Quade, E. S. (1989). *Analysis for public decisions* (3rd ed.). New York: Elsevier Science.

Rahman, S. A., Walker, W. E., & Marchau, V. (2008). *Coping with uncertainties about climate change in infrastructure planning – An adaptive policymaking approach*. ECORYS Nederland BV, P.O. Box 4175, 3006 AD, Rotterdam, The Netherlands.

Rosenhead, J., & Mingers, J. (Eds.). (2001). *Rational analysis for a problematic world revisited: Problem structuring methods for complexity, uncertainty, and conflict*. Chichester, UK: Wiley.

Seong, S., Popper, S. W., & Zheng, K. (2005). *Strategic choices in science and technology Korea in the era of a rising China* (MG-320-KISTEP). Santa Monica, CA: RAND.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal, 27*, 379–423. 623–656, July, October.

Smithson, M. (1989). *Ignorance and uncertainty: Emerging paradigms*. New York: Springer.

Taleb, N. N. (2007). *The black swan: The impact of the highly improbable*. New York: Random House.

Taneja, P., Walker, W. E., Ligteringen, H., Van Schuylenburg, M., & van der Plas, R. (2010). Implications of an uncertain future for port planning. *Maritime Policy & Management, 37*(3), 221–245.

van Asselt, M. B. A. (2000). *Perspectives on uncertainty and risk*. Dordrecht, The Netherlands: Kluwer.

van Asselt, M. B. A., & Rotmans, J. (2002). Uncertainty in integrated assessment modelling: From positivism to pluralism. *Climatic Change, 54*, 75.

van der Heijden, K. (1996). *Scenarios: The art of strategic conversation*. Chichester, UK: Wiley.

van der Sluijs, J. P. (1997). *Anchoring amid uncertainty: On the management of uncertainties in risk assessment of anthropogenic climate change*. Ph.D. dissertation, University of Utrecht, The Netherlands.

van Drunen, M., Leusink, A., Lasage, R. (2009). Towards a climate-proof Netherlands. In A. K. Biswas, C. Tortajade, & R. Izquierdo (Eds.), *Water management in 2020 and beyond*. Springer.

Walker, W. E., Harremoës, P., Rotmans, J., van der Sluijs, J. P., van Asselt, M. B. A., Janssen, P., & Krayer von Krauss, M. P. (2003). Defining uncertainty: A conceptual basis for uncertainty management in model-based decision support. *Integrated Assessment, 4*(1), 5–17.

Walker, W. E., Rahman, S. A., & Cave, J. (2001). Adaptive policies, policy analysis, and policymaking. *European Journal of Operational Research, 128*(2), 282–289.

Degeneracy

The situation in which a linear-programming problem has a basic feasible solution with at least one basic variable equal to zero. If the problem is degenerate, then an extreme point of the convex set of solutions may correspond to several feasible bases. As a result, the simplex method may move through a sequence of bases with no improvement in the value of the objective function. In rare cases, the algorithm may cycle repeatedly through the same sequence of bases and never converge to an optimal solution. Anticycling rules, and perturbation and lexicographic techniques prevent this risk, but usually at some computational expense.

See

- ▶ Anticycling Rules
- ▶ Bland's Anticycling Rules
- ▶ Cycling
- ▶ Linear Programming
- ▶ Simplex Method (Algorithm)

Degeneracy Graphs

Tomas Gal
Fern Universität in Hagen, Hagen, Germany

Introduction

For a given linear-programming problem, primal degeneracy means that a basic feasible solution has at least one basic variable equal to zero. The problem is dual degenerate if a nonbasic variable has its reduced cost equal to zero (the condition for a multiple optimal solution to exist). Primal degeneracy may arise when there are some (weakly) redundant constraints (Karwan et al. 1983) or the structure of the corresponding convex polyhedral feasible set causes an extreme point to become overdetermined.

In nonlinear programming, such points are sometimes called singularities (Guddat et al. 1990). Here, constraint redundancy is equivalent to the failure of the linear independence constraint qualification of the binding constraint gradients, which, in general, leads to the nonuniqueness of optimal Lagrange multipliers (Fiacco and Liu 1993).

We focus here on primal degeneracy in the linear case: it is associated with multiple optimal bases and it allows for basis cycling to occur, that is, the nonconvergence of the simplex method due to the repeating of a sequence of nonoptimal feasible bases.

Let σ, called the degeneracy degree, be the number of zeros in a basic feasible solution. Also, let $U_{\min}$ and $U_{\max}$ be the minimal and the maximal number of possible bases associated with a degenerate vertex, respectively (Kruse 1986). To illustrate how many bases can be associated with a degenerate vertex, Table 1 shows, for some values for n, the number of (decision) variables, the associated values of σ, $U_{\min}$ and $U_{\max}$.

Historical Background

Soon after the simplex method had been invented by George Dantzig, he recognized that degeneracy in the primal problem could cause a cycle of bases to occur. In fact, Dantzig's original convergence proof of the simplex method assumed that all basic feasible solutions were nondegenerate. In the Fall of 1950, Dantzig made the first suggestion of a nondegeneracy procedure in a lecture on linear programming (LP) (Dantzig 1963). Charnes (1952) proposed a so-called perturbation method to prevent cycling. Since then, many variants of nondegeneracy and anticycling methods have been developed. For a review of degeneracy and its influence on computation, see Gal (1993).

In the end of the 1970s, a unifying approach to the analysis of degeneracy problems was proposed in terms of degeneracy graphs (Gal 1985). These graphs are used to define the connections among the bases associated with a degenerate vertex. From Table 1, it obvious that for real-world problems, with large numbers of constraints and variables, such systems of connections might have quite complex structures. It was felt that the language of graph theory could be applied to good advantage in explaining the relationships between degenerate bases.

Since they were first proposed, degeneracy graphs have become an important topic of research (Geue 1993; Kruse 1986; Niggemeier 1993; Zörnig 1993). In these works, the general theory of degeneracy graphs has been developed, the possibilities for their application to transportation, integer programming and other problems have been studied, and algorithmic aspects to solve various degeneracy problems have been investigated.

The main problem that led to the idea of using a graph theoretical representation was the so called

Degeneracy Graphs, Table 1 Values for σ, $U_{\min}$, $U_{\max}$

n	σ	$U_{\min}$	$U_{\max}$
5	3	16	56
10	5	12	3003
50	5	752	3.48×10^{6}
50	40	6.59×10^{12}	5.99×10^{25}
100	30	3.865×10^{10}	2.61×10^{39}
100	50	2.93×10^{16}	2.01×10^{40}
100	80	1.33×10^{25}	3×10^{52}

neighboring problem: Given a vertex of a convex polytope, find all neighboring vertices. This is not a problem if the given vertex is nondegenerate. It becomes a problem (Table 1) when the given vertex is degenerate.

Degeneracy Graphs

Given a σ-degenerate vertex $\boldsymbol{x}^{\mathrm{o}}$; to this vertex the set

$$B^{\mathrm{o}} = \{B | B \text{ feasible basis of } x^{\mathrm{o}}\}$$

is assigned. Denote

$$\begin{gathered}\text{by ‘‘} \leftarrow + \rightarrow \text{’’ a pivot} - \text{step with a positive} \\ \text{pivot } (\text{a positive pivot} - \text{step}) \\ \text{by ‘‘} \leftarrow - \rightarrow \text{’’ a pivot} - \text{step with a negative} \\ \text{pivot } (\text{a negative pivot} - \text{step}) \\ \text{by ‘‘} \leftarrow \rightarrow \text{’’ a pivot} - \text{step if any nonzero} \\ \text{pivot can be used } (\text{pivot} - \text{step}).\end{gathered}$$

The graph of a polytope X is the undirected graph

$$G(X) := G = (V, E),$$

where

$$V = \{B | B \text{ is a feasible basis of the corresponding system of equations}\}$$

and

$$E = \{\{B, B'\} \subseteq V | B \leftarrow + \rightarrow B'\}.$$

The degeneracy graph (DG) that is used to study various degeneracy problems with respect to a degenerate vertex is defined as follows. Let $\boldsymbol{x}^{\mathrm{o}} \in X \subset \Re^n$ be a σ-degenerate vertex. Then the (undirected) graph

$$G(x^{\mathrm{o}}) := G^{\mathrm{o}} = (B^{\mathrm{o}}, E^{\mathrm{o}})$$

where

$$E^{\mathrm{o}} = \{\{B_u, B_v\} \subseteq B^{\mathrm{o}} | B_u \longleftrightarrow B_v\}, u, v \in \{1, \ldots, U\}, U_{\min} \leq U \leq U_{\max} \qquad (1)$$

and U, the degeneracy power of $\boldsymbol{x}^{\mathrm{o}}$, is called the general $\sigma \times n - G$ of $\mathbf{x}^{\mathrm{o}}$. If, in (1), the operator is $\leftarrow + \rightarrow$ or $\leftarrow - \rightarrow$, then the corresponding graph is called the positive or negative DG of $\boldsymbol{x}^{\mathrm{o}}$, respectively.

These notions have been used to develop a theory of the DG. For example: the diameter, d, of a general DG satisfies $d \leq \min\{\sigma, n\}$; a general DG is always connected; a formula to determine the number of nodes of a DG has been developed; the connectivity of a DG is ≥ 2; every pair of nodes in any DG lies on a cycle (Zörnig 1993).

An interesting consequence of this theory is that every degenerate vertex can be exited in at most d (diameter) steps. Other theoretical properties of DGs help in explaining problems in, for example, sensitivity analysis with respect to a degenerate vertex (Gal 1997; Kruse 1993). Also, this theory helps to work out algorithms to solve the neighborhood problem and to determine all vertices of a convex polytope (Gal and Geue 1992; Geue 1993; Kruse 1986). With respect to a degenerate optimal vertex of an LP-problem, algorithms to perform sensitivity analysis and parametric programming have been developed (Gal 1995). Also, the connection between weakly redundant constraints, degeneracy and sensitivity analysis has been studied (Gal 1992).

Concluding Remarks

Degeneracy graphs have been applied to help solve the neighborhood problem, to explain why cycling in LP occurs, to develop algorithms to determine two-sided shadow prices, to determine all vertices of a (degenerate) convex polyhedron, and to perform sensitivity analysis under (primal) degeneracy. DGs can be used in any mathematical-programming problem that uses some version of the simplex method or, more generally, in any vertex searching method.

See

- Degeneracy
- Graph Theory
- Linear Programming

- ▶ Parametric Programming
- ▶ Redundant Constraint
- ▶ Sensitivity Analysis

References

Charnes, A. (1952). Optimality and degeneracy in linear programming. *Econometrica, 20*, 160–170.

Dantzig, G. B. (1963). *Linear programming and extensions*. Princeton, New Jersey: Princeton University Press.

Fiacco, A. V., & Liu, J. (1993). Degeneracy in NLP and the development of results motivated by its presence. In T. Gal (Ed.), *Degeneracy in optimization problems. Annals of OR*, *46/47*, 61–80

Gal, T. (1985). On the structure of the set bases of a degenerate point. *Journal of Optimization Theory and Applications, 45*, 577–589.

Gal, T. (1986). Shadow prices and sensitivity analysis in LP under degeneracy — state-of-the-art survey. *OR-Spektrum, 8*, 59–71.

Gal, T. (1992). Weakly redundant constraints and their impact on postoptimal analysis in LP. *European Journal of Operational Research, 60*, 315–336.

Gal, T. (1993). Selected bibliography on degeneracy. In: T. Gal (Ed.), *Degeneracy in optimization problems. Annals of OR, 46/47*, 1–7.

Gal, T. (1995). *Postoptimal analyses, parametric programming, and related topics*. Berlin, New York: W. de Gruyter.

Gal, T. (1997). Linear programming 2: Degeneracy graphs. In T. Gal & H. J. Greenberg (Eds.), *Advances in sensitivity analysis and parametric programming*. Dordrecht: Kluwer Academic Publishers.

Gal, T., & Geue, F. (1992). A new pivoting rule for solving various degeneracy problems. *Operations Research Letters, 11*, 23–32.

Geue, F. (1993). An improved N-tree algorithm for the enumeration of all neighbors of a degenerate vertex. In: T. Gal (Ed.), *Degeneracy in optimization problems. Annals of OR*, *46/47*, 361–392.

Guddat, J. F., Guerra Vasquez, F. & Jongen, Th. H. (1991). *Parametric Optimization: Singularities, Path Following and Jumps*. New York: R.G. Teubner and J. Wiley.

Karwan, M.H., Lotfi, F., Telgen, J.& Zionts, S. (Eds), (1983). *Redundancy in mathematical programming: A state-of-the-art survey*. Lecture Notes in Econ. and Math. Systems 206. Berlin: Springer Verlag.

Kruse, H. J. (1986). *Degeneracy graphs and the neighborhood problem*. Lecture Notes in Econ. and Math. Systems 260. Berlin: Springer Verlag.

Kruse, H. J. (1993). On some properties of σ-degeneracy graphs. In: T. Gal (Ed.), *Degeneracy in optimization problems. Annals of OR, 46/47*, 393–408.

Niggemeier, M. (1993). Degeneracy in integer linear optimization problems: A selected bibliography. In: T. Gal (Ed.), *Degeneracy in optimization problems. Annals of OR, 46/47*, 195–202.

Zörnig, P. (1993). A theory of degeneracy graphs. In: T. Gal (Ed.), *Degeneracy in optimization problems. Annals of OR, 46/47*, 541–556.

Degenerate Solution

A basic (feasible) solution in which some basic variables are zero.

See

- ▶ Anticycling Rules
- ▶ Cycling
- ▶ Degeneracy
- ▶ Degeneracy Graphs

Degree

The number of edges incident with a given node in a graph.

See

- ▶ Graph Theory

Delaunay Triangulation

- ▶ Computational Geometry
- ▶ Voronoi Constructs

Delay

The time spent by a customer in queue waiting to start service.

See

- ▶ Queueing Theory
- ▶ Waiting Time

Delphi Method

James A. Dewar and John A. Friel
RAND Corporation, Santa Monica, CA, USA

Introduction

The Delphi method was developed at the RAND Corporation from studies on decision making that began in 1948. The seminal work, "An Experimental Application of the Delphi Method to the Use of Experts," was written by Dalkey and Helmer (1963).

The primary rationale for the technique is the age-old adage "two heads are better than one," particularly when the issue is one where exact knowledge is not available. It was developed as an alternative to the traditional method of obtaining group opinions — face-to-face discussions. Experimental studies had demonstrated several serious difficulties with such discussions. Among them were: (1) influence of the dominant individual (the group is highly influenced by the person who talks the most or has most authority); (2) noise (studies found that much communication in such groups had to do with individual and group interests rather than problem solving); and (3) group pressure for conformity (studies demonstrated the distortions of individual judgment that can occur from group pressure).

The Delphi method was specifically developed to avoid these difficulties. In its original formulation it had three basic features: (1) anonymous response — opinions of the members of the group are obtained by formal questionnaire; (2) iteration and controlled feedback — interaction is effected by a systematic exercise conducted in several iterations, with carefully controlled feedback between rounds; and (3) statistical group response — the group opinion is defined as an appropriate aggregate of individual opinions on the final round.

Procedurally, the Delphi method begins by having a group of experts answer questionnaires on a subject of interest. Their responses are tabulated and fed back to the entire group in a way that protects the anonymity of their responses. They are asked to revise their own answers and comment on the group's responses. This constitutes a second round of the Delphi. Its results are tabulated and fed back to the group in a similar manner and the process continues until convergence of opinion, or a point of diminishing returns, is reached. The results are then compiled into a final statistical group response to assure that the opinion of every member of the group is represented.

In its earliest experiments, Delphi was used for technological forecasts. Expert judgments were obtained numerically (e.g., the date that a technological advance would be made), and in that case it is easy to show that the mean or median of such judgments is at least as close to the true answer as half of the group's individual answers. From this, the early proponents were able to demonstrate that the Delphi method produced generally better estimates than those from face-to-face discussions.

One of the surprising results of experiments with the technique was how quickly in the successive Delphi rounds that convergence or diminishing returns is achieved. This helped make the Delphi technique a fast, relatively efficient, and inexpensive tool for capturing expert opinion. It was also easy to understand and quite versatile in its variations. By 1975, there were several hundred applications of the Delphi method reported on in the literature. Many of these were applications of Delphi in a wide variety of judgmental settings, but there was also a growing academic interest in Delphi and its effectiveness.

Critique

Sackman (1975), also of the RAND Corporation, published the first serious critique of the Delphi method. His book, *Delphi Critique*, was very critical of the technique — particularly its numerical aspects — and ultimately recommended (p. 74) "that … Delphi be dropped from institutional, corporate, and government use until its principles, methods, and fundamental applications can be experimentally established as scientifically tenable."

Sackman's critique spurred both the development of new techniques for obtaining group judgments and a variety of studies comparing Delphi with other such techniques. The primary alternatives can be categorized as statistical group methods (where the answers of the group are tabulated statistically without any interaction); unstructured, direct interaction (another name for traditional, face-to-face

discussions); and structured, direct interaction (such as the Nominal Group Technique of Gustafson et al. 1973). In his comprehensive review, Woudenberg (1991) found no clear evidence in studies done for the superiority of any of the four methods over the others. Even after discounting several of the studies for methodological difficulties, he concludes that the original formulation of the quantitative Delphi is in no way superior to other (simpler, faster, and cheaper) judgment methods.

Another comprehensive evaluation of Delphi (Rowe et al. 1991) comes to much the same conclusion that Sackman and Woudenberg did, but puts much of the blame on studies that stray from the original precepts. Most of the negative studies use non-experts with similar backgrounds (usually undergraduate or graduate students) in simple tests involving almanac-type questions or short-range forecasts. Rowe et al. (1991) point out that these are poor tests of the effects that occur when a variety of experts from different disciplines iterate and feed back their expertise to each other. They conclude that Delphi does have potential in its original intent as a judgment-aiding technique, but that improvements are needed and those improvements require a better understanding of the mechanics of judgment change within groups and of the factors that influence the validity of statistical and nominal groups.

Applications

In the meantime, it is generally conceded that Delphi is extremely efficient in achieving consensus and it is in this direction that many subsequent Delphi evaluations have been used. Variations of the Delphi method, such as the policy Delphi and the decision Delphi, generally retain the anonymity of participants and iteration of responses. Many retain specific feedback as well, but these more qualitative variations generally drop the statistical group response. Delphi has been used in a wide variety of applications from its original purpose of technology forecasting (one report says that Delphi has been adopted in approximately 90% of the technological forecasts and studies of technological development strategy in China) to studying the future of medicine, examining possible shortages of strategic materials, regional planning of water and natural resources, analyzing national drug abuse policies, and identifying corporate business opportunities.

In addition, variations of Delphi continue to be developed to accommodate the growing understanding of its shortcomings. For example, a local area network (LAN) was constructed, composed of lap-top computers connected to a more capable workstation. Each participant had a dedicated spreadsheet available on a lap-top computer. The summary spreadsheet maintained by the workstation was displayed using a large-screen projector, and included the mean, media, standard deviation, and histogram of all the participants scores. In real-time, the issues were discussed, the various participants presented their interpretation of the situation, presented their analytic arguments for the scores they believed to be appropriate, and changed their scoring as the discussion developed. Each participant knew their scores, but not those of the other participants. When someone was convinced by the discussions to change a score they could do so anonymously. The score was transmitted to the workstation where a new mean, median, standard deviation, and histogram were computed and then displayed using a large screen projector. This technique retained all the dimensions of the traditional Delphi method and at the same time facilitated group discussion and real-time change substantially shortening the time typically required to complete a Delphi round.

See

- ▶ Decision Analysis
- ▶ Group Decision Computer Technology
- ▶ Group Decision Making

References

Dalkey, N., & Helmer, O. (1963). An experimental application of the delphi method to the use of experts. *Management Science, 9*, 458–467.

Gustafson, D. H., Shukla, R. K., Delbecq, A., & Walster, G. W. (1973). A comparison study of differences in subjective likelihood estimates made by individuals, interacting groups, delphi groups, and nominal groups. *Organizational Behavior and Human Performance, 9*, 280–291.

Keeney, S., & McKenna, H. (2011). *The delphi method in nursing and health research*. West Sussex, UK: John Wiley & Sons.

Rowe, G., Wright, G., & Bolger, F. (1991). Delphi: A reevaluation of research and theory. *Technological Forecasting and Social Change, 39*, 235–251.
Sackman, H. (1975). *Delphi critique*. Lexington, MA: Lexington Books.
Woudenberg, F. (1991). An evaluation of delphi. *Technological Forecasting and Social Change, 40*, 131–150.

Density

The proportion of the coefficients of a constraint matrix that are nonzero. For a given $(m \times n)$ matrix $\boldsymbol{A} = (a_{ij})$, if k is the number of nonzero a_{ij}, then the density is given by $k/(m \times n)$. Most large-scale linear-programming problems have a low density of the order of 0.01.

See

- Sparse Matrix
- Super-Sparsity

Density Function

When the derivative $f(x)$ of a cumulative probability distribution function $F(x)$ exists, it is called the density or probability density function (PDF).

See

- Probability Density Function (PDF)

Departure Process

Usually refers to the random sequence of customers leaving a queueing service center. More generally, it is the random point process or marked point process with marks representing aspects of the departure stream and/or the service center or node from which they are leaving. For example, the marked point process $(\boldsymbol{X}^d, \boldsymbol{T}^d)$ for departures from an M/G/1 queue takes $\boldsymbol{X}^d$ as the Markov process for the queue length process immediately after the departure time and $\boldsymbol{T}^d$ is the actual time of departure.

See

- Markov Chains
- Markov Processes
- Queueing Theory

Descriptive Model

A model that attempts to describe the actual relationships and behavior of a man/machine system. For a decision problem, such a model attempts to describe how individuals make decisions.

See

- Decision Problem
- Expert Systems
- Mathematical Model
- Model
- Normative Model
- Prescriptive Model

Design and Control

For a queueing system, design deals with the permanent, optimal setting of system parameters (such as service rate and/or number of servers), while control deals with adjusting system parameters as the system evolves to ensure certain performance levels are met. A typical example of a control rule is that a server is to be added when the queue size is greater than a certain number (say N_1) and when the queue size drops down to $N_2 < N_1$, the server goes to other duties.

See

- Dynamic Programming
- Markov Decision Processes
- Queueing Theory

Detailed Balance Equations

A set of equations balancing the expected, steady-state flow rates or probability flux between each pair of states or entities of a stochastic process (most typically a Markov chain or queueing problem), for example written as:

$$\pi_j q(j,k) = \pi_k q(k,j)$$

where π_m is the probability that the state is m and $q(m, n)$ is the flow rate from states m to n. The states may be broadly interpreted to be multi-dimensional, as in a network of queues, and the entities might be individual service centers or nodes. Contrast this with global balance equations, where the average flow into a single state is equated with the flow out.

See

- ▶ Markov Chains
- ▶ Networks of Queues
- ▶ Queueing Theory

Determinant

- ▶ Matrices and Matrix Algebra

Deterministic Model

A mathematical model in which it is assumed that all input data and parameters are known with certainty.

See

- ▶ Descriptive Model
- ▶ Mathematical Model
- ▶ Model
- ▶ Normative Model
- ▶ Prescriptive Model
- ▶ Stochastic Model

Developing Countries

Roberto Diéguez Galvão[1] and Graham K. Rand[2]
[1]Federal University of Rio de Janeiro, Brazil
[2]Lancaster University, Lancaster, UK

Introduction

OR started to establish itself in the developing countries in the 1950s, approximately one decade after its post-war inception in Great Britain and the United States. The main organizational basis of OR in the developing world are the national OR societies. These are in some cases well established, in other cases incipient. A number of them are members of the International Federation of Operational Research Societies (IFORS) and belong to regional groups within IFORS. In particular, ALIO, the Association of Latin American OR Societies, has the majority of its member societies belonging to developing countries. APORS, the Association of Asian-Pacific OR Societies within IFORS, also represents OR societies from developing countries. In 1989 a Developing Countries Committee was established as part of the organizational structure of IFORS, with the objective of coordinating OR activities in the developing countries and promoting OR in these countries.

The Social, Political, and Technological Environment

To speak of developing countries in general may lead to erroneous conclusions, since the conditions vary enormously from one country to another. First of all, how to characterize a developing country? Which countries may be classified as developing? The United Nations has, for some years now, started to distinguish between more and less developed countries in the developing world. It has adopted the term "less developed countries" (LDCs) to address those developing countries that fall below some threshold levels measured by social and economic indicators. But these questions are clearly well beyond the scope here.

The view here is that developing countries are those in which large strata of the population live at or below the subsistence level, where social services are practically nonexistent for the majority of the population, where the educational and cultural levels are in general very low. The political consequence of this state of affairs is a high degree of instability for the institutions of these countries, at all levels.

The economy is generally very dependent on the industrialized nations. Bureaucracy, economic dependence and serious problems of infrastructure conspire against economic growth. In the technical sphere there is again a high level of dependency on the industrialized world, with very little technological innovation produced locally. It is against this difficult background that one must consider the role OR can play and how OR can be used as a tool for development.

The Use of OR

Here the existence of three different emphases in the development of OR is considered: (i) development of theory, which takes place mostly in the universities; (ii) development of methods for specific problems, which occurs both in the universities and in the practical world; (iii) applications, which occur mostly in the practical world. The problems of OR are therefore a continuum, and both developing and industrialized nations share in all these three aspects of the continuum. The more important aspect for the developing countries tends, however, to be applications due to the nature of problems these nations have to face and their social, political, and technological environment discussed above. According to Rosenhead (1995), another important aspect is that existing theory and methods, grown in the developed world, are in many cases a poor fit for the problems facing the developing countries. Work on novel applications will be likely to throw up new methods and techniques of general interest.

The use of OR in the developing world is often seen as disconnected from the socio-economic needs of the respective countries, see Galvão (1988). Valuable theoretical contributions originate in these countries, but little is seen in terms of new theory and methods developed for the problems facing them.

A common situation in developing countries is a highly uncertain environment, which leads to the notion of wicked problems. These are, for example, problems for which there is often little or no data available, or where the accuracy of data is very poor. Complex decisions must nevertheless be made, against a background of competing interests and decision makers. There are not many tools available for solving these wicked problems, which are quite common in developing countries.

One of the main characteristics of applied OR projects in developing countries is that a large majority of them have not been implemented, see Löss (1981). This is due to a high degree of instability in institutions in these countries, to a lack of management education in OR, and to a tendency by OR analysts to attempt to use sophisticated OR techniques without paying due attention to the local environment and to the human factor in applied OR projects. These issues arise both in developed and developing countries, but experience indicates that they are more often overlooked in the latter.

A Special Issue of the European Journal of Operational Research (Bornstein et al. 1990) was dedicated to OR in Developing Countries. A review paper (White et al. 2011) provides an overall picture of the state of OR in the developing countries. In particular, it examines coverage in terms of countries and methods and highlights the contribution which OR is making towards the theme of poverty, the reduction of which is regarded as the key focus of development policy interventions as reflected in the Millennium Development Goals. Jaiswal (1985) and Rosenhead and Tripathy (1996) contain important contributions to the subject of OR in developing countries.

ICORD '92: The Ahmedabad Conference

Since the 1950s, there has been a controversy on the role of OR in developing countries. The central issue in this controversy is the following: Is there a separate OR for developing countries? If so, what makes it different from traditional OR? What steps could be taken to further OR in developing countries?

This issue has been discussed in different venues and several published papers have addressed it, see, for example, Bornstein and Rosenhead (1990). At one end of the scale there are those who think that there is nothing special about OR in developing countries, perhaps only less resources are available in these

countries to conduct theoretical/applied work. They argue that the problem should resolve itself when each country reaches appropriate levels of development, and not much time should be dedicated to this issue. At the other end there are those who think that because of a different material basis and due to problems of infrastructure, OR does have a different role to play in these countries. In the latter case, steps should be taken to ensure that OR plays a positive role in the development of their economies and societies.

Much changed in the latter part of the 1990s with the demise of communism in Europe and the emphasis on the globalization of the economy. The viewpoint that there is a separate OR for developing countries lost strength as a consequence. It had its high moment during ICORD '92, the first International Conference on Operational Research for Development, which took place in December 1992, at the Indian Institute of Management (IIM) in Ahmedabad. It was supported by IFORS, The British OR Society and the OR Society of India. It was partly funded by IIM itself, The Tata Iron and Steel Company (India) and (indirectly) by the Commonwealth Secretariat. Participants at the Conference numbered more than 60 and countries represented included Australia, Brazil, Eire, Great Britain, Greece, India, Kenya, Malaysia, Mexico, Nigeria, Peru, South Africa, Sri Lanka, United States, and Venezuela. Some 40 contributed papers were delivered and plenary speakers included the President of IFORS, Professor Brian Haley, Professor Kirit Parikh, Director of the Indira Ghandi Institute for Development in Bombay, and Dr. Francisco Sagasti of Peru, who had just spent five years in senior positions at the World Bank (Rosenhead 1993).

A series of plenary sessions were held, which resulted in a statement which has come to be known as the Ahmedabad Declaration, a political document drafted with the intention of strengthening the OR for Development movement, that called for a range of actions from IFORS to support and strengthen OR in developing countries, including a call for more space for discussion of OR for Development issues in OR departments in developed countries, for IFORS support for successor conferences to ICORD '92, and for IFORS increased economic support of OR activities in developing countries. It relied mainly on IFORS for its implementation. Despite IFORS' continued support of some OR activities in the developing countries, few of the main recommendations of the declaration were implemented. ICORD '96, the second Conference in the series, which took place in Rio de Janeiro, Brazil, in August 1996, was a disappointing sequel to the Ahmedabad Conference and signaled the decline of the movement.

Despite the perceived lack of commitment on the part of IFORS to implement these proposals (Rosenhead 1998), IFORS support of development related OR activities have continued, including the support of successor ICORDs, held in Manila, The Philippines (1997), Berg-en-Dal, South Africa (2001), Jamshedpur, India (2005), Fortaleza, Brazil (2007) and Djerba Island, Tunisia (2012). The IFORS Prize for OR in Development (known as the Third World Prize until 1993) competition has been held at every triennial conference since 1987. The Prize recognizes exemplary work in the application of OR to address issues of development. More recently, a particular focus has been encouraging the development of an OR infrastructure in Africa, and, with EURO, IFORS has sponsored conferences and scholarships in the African continent.

A fuller account of IFORS initiatives in promoting the use of OR for development is described in by Rand (2000). See also del Rosario and Rand (2010).

Is it safe to conclude, therefore, that those who advocate that there is nothing special about OR in developing countries had the better insight on the controversy? The hard facts of life show that little has changed in the social, political and technological environment in the developing countries. The decline of the OR for Development movement is a consequence of the new balance of power in global affairs since the Soviet Union ceased to exist. This decline did not occur because conditions in the developing world improved, or because OR has failed to contribute to the development of the respective economies and societies.

See

- ▶ IFORS
- ▶ Practice of Operations Research and Management Science
- ▶ Wicked Problems

References

Bornstein, C. T., & Rosenhead, J. (1990). The role of operational research in less developed countries: A critical approach. *European Journal of Operational Research, 49*, 156–178.

Bornstein, C. T., Rosenhead, J., & Vidal, R. V. V. (Eds.). (1990). Operational research in developing countries. *European Journal of Operational Research*, *49*(2), 155–294

del Rosario, E. A., & Rand, G. K. (2010). IFORS: 50 at 50. *Boletín de Estadística e Investigación Operativa, 26*(1), 84–96.

Galvão, R. D. (1988). Operational research in latin America: Historical background and future perspectives. In G. K. Rand (Ed.), *Operational research '87* (pp. 19–31). Amsterdam: North-Holland.

Jaiswal, N.K. (Ed.). (1985). *OR for developing countries*. Operational Research Society of India.

Löss, Z. E. (1981). O Desenvolvimento da Pesquisa Operacional no Brasil (The Development of OR in Brazil), M. Sc. Thesis, COPPE/Federal University of Rio de Janeiro.

Rand, G. K. (2000). IFORS and developing countries. In A. Tuson (Ed.), *Young OR 11: Tutorial & keynote papers* (pp. 75–86). Birmingham: Operational Research Society.

Rosenhead, J. (1993). ICORD '92–International Conference on operational research for development. *OR for Developing Countries Newsletter, 3*(3), 1–4.

Rosenhead, J. (1998). Ahmedabad 6 years on – has IFORS delivered? *OR for Developing Countries Newsletter, 6*(2), 5–8.

Rosenhead, J. (1995). Private communication.

Rosenhead, J., & Tripathy, A. (Eds.). (1996). *Operational research for development*. New Delhi: New Age International Limited.

White, L., Smith, H., & Currie, C. (2011). OR in developing countries: A review. *European Journal of Operational Research, 208*, 1–11.

Development Tool

Software used to facilitate the development of expert systems. The three types of tools are programming languages, shells, and integrated environments.

See

► Expert Systems

Devex Pricing

A criterion for selecting the variable entering the basis in the simplex method. Devex pricing chooses the incoming variable with the largest gradient in the space of the initial nonbasic variables. This is contrasted with the usual simplex method entering variable criterion that chooses the incoming variable based on the largest gradient in the space of the current nonbasic variables. The Devex criterion tends to reduce greatly the total number of simplex iteration on large problems.

See

► Linear Programming
► Simplex Method (Algorithm)

Deviation Variables

Variables used in goal programming models to represent deviation from desired goals or resource target levels.

See

► Goal Programming

DFR

Decreasing failure rate.

See

► Reliability of Stochastic Systems

Diameter

The maximum distance between any two nodes in a graph.

See

► Graph
► Graph Theory

Diet Problem

A linear program that determines a diet satisfying specified recommended daily allowance (RDAs) requirements at minimum cost. Stigler's diet problem was one of the first linear-programming problems solved by the simplex method.

See

- ▶ Linear Programming
- ▶ Simplex Method (Algorithm)
- ▶ Stigler's Diet Problem

References

Gass, S. I., & Garille, S. (2001). Stigler's diet problem revisited. *Operations Research, 49*(1), 1–13.
Stigler, G. J. (1945). The cost of subsistence. *Journal of Farm Economics, 27*(2), 303–314.

Differential Games

Gary M. Erickson
University of Washington, Seattle, WA, USA

Introduction

Differential games offer a valuable modeling approach for problems in operations research (OR) and management science (MS). Differential game models are useful because they combine the key aspects of dynamic optimization and game theory. As such, differential game modeling allows the analysis of a broad set of problems that involve decisions by multiple players over a time horizon. After a discussion of the essential concepts of differential games, applications from the literature are reviewed as examples of how differential game methodology has been used to study problems of interest to OR and MS.

Discussion

A differential game is a game with continuous-time dynamics. Two types of variables are involved, state variables and control variables, both of which vary with time. Control variables are managed by the players. State variables are subject to the dynamic influence of the control variables, and evolve according to differential equations. Each player has an objective function that consists of a stream of instantaneous payoffs integrated over a horizon, plus, perhaps, a salvage value if the horizon is finite. The decision problem for each player is to determine a continuous path of control variable values that maximizes the player's objective function, while taking into account what the player knows or anticipates about the decisions of the other players in the game.

Complete information is assumed in a differential game, so that player outcomes given different combinations of player strategies are known to all players, and each player is able to infer correctly the best strategies for the other players. Also, an assumption is typically made that the players are unable to agree to cooperate, and so are engaged in a noncooperative differential game. Further, if the players choose their strategies simultaneously, the appropriate way to determine what strategies the players are likely to adopt is to identify a Nash equilibrium. A Nash equilibrium is a set of player strategies such that each player is unable to improve their outcome, given the strategies of the remaining players. In a Nash equilibrium, no individual player has an incentive to deviate to another strategy.

There are two types of Nash equilibrium that can be derived: open-loop and feedback. Alternative terms for feedback are closed-loop and Markovian (Dockner et al. 2000, p. 59). The two equilibrium types differ in terms of what information is used to develop the players' strategies. In an open-loop Nash equilibrium, the players' strategies are a function of time only, while feedback Nash equilibrium strategies depend on levels of the state variables as well as time. Further, for a differential game with an infinite horizon, and in which time is an explicit factor in the objective functions only through discount factors, it is appropriate to focus on stationary feedback strategies, which depend on levels of the state variables only (Jørgensen and Zaccour 2004, pp. 7–8).

Different methods are typically used to derive the different Nash equilibrium concepts. The maximum principle of optimal control, with Hamiltonians and costate variables, is used to determine open-loop Nash equilibria (Kamien and Schwartz 1991, p. 274). To derive an open-loop equilibrium, a Hamiltonian is created for each player, and necessary conditions produce a system of differential equations that can be solved numerically as a two-point boundary value problem.

In theory, a feedback Nash equilibrium can also be determined using optimal control methods, but the maximum principle is difficult to apply for feedback strategies, since the solution requires that the strategies of the players be known even as they need to be derived. An alternative way to develop feedback Nash equilibrium strategies is through a dynamic programming approach with value functions and Hamilton-Jacobi-Bellman equations (Kamien and Schwartz 1991, p. 276). The Hamilton-Jacobi-Bellman equations form a system of partial differential equations, which for many problems are inherently impossible to solve. For certain problems, though, it is possible to discern an appropriate functional form for the value functions that allows a solution. In particular, for infinite horizon games, it is often possible to derive stationary feedback equilibrium strategies analytically as closed-form functions of the state variables.

An alternative to simultaneous play of strategies is that of Stackelberg games (Dockner et al. 2000, ch.5; Jørgensen and Zaccour 2004, pp. 17–22). Stackelberg games have an alternative information structure, one in which one player takes on a leadership role and makes their strategy choice known before other players choose their strategies. Such a structure can be appropriate for certain problems, such as supply chain management, where coordination may be achieved to benefit of the supply chain overall through one of the members of the supply chain taking a leadership role.

As for Nash equilibria in games with simultaneous play, there are open-loop and feedback Stackelberg equilibria that can be derived. In an open-loop Stackelberg equilibrium with two players (Dockner et al. 2000, pp. 113–134; Jørgensen and Zaccour 2004, pp. 17–20), the Stackelberg leader announces a control path, and, if the Stackelberg follower believes that the leader will stay with the announced control path, the follower will determine their best response control path by solving an optimal control problem with the leader's control path as given. The leader then solves an optimal control problem that incorporates the follower's best response.

For a feedback Stackelberg equilibrium, Basar and Olsder (1995, pp. 416–420) present a feedback Stackelberg solution, which involves instantaneous stagewise Stackelberg leadership, where a stage is an arbitrary combination of time and state variable values. In the development of the feedback Stackelberg solution, stagewise Hamilton-Jacobi-Bellman equations are formed for the leader and the follower, the equation for the follower defining an optimal response and that for the leader incorporating the optimal response of the follower.

The open-loop and feedback equilibrium concepts for both Nash and Stackelberg games can be further examined on the basis of important credibility-related criteria. Dockner et al. (2000, pp. 98–105) and Jørgensen and Zaccour (2004, pp. 15-16) discuss two such criteria, time consistency and subgame perfectness.

A Nash equilibrium is time consistent if at some intermediate point in a differential game, the players choose not to depart from their equilibrium strategies. Dockner et al. (2000, p. 99) and Jørgensen and Zaccour (2004, p. 15) define a subgame that begins at an intermediate time point in the game, and has particular values for the state variables at the time. An equilibrium for the original game "...is time consistent if it is also an equilibrium for any subgame that starts out on the equilibrium state trajectory..." (Jørgensen and Zaccour 2004, p. 15). Both open-loop and feedback Nash equilibria are time consistent. The open-loop Stackelberg equilibrium is not always time consistent, however. As Dockner et al. (2000, pp. 113–134) discuss, an open-loop Stackelberg equilibrium fails to be time consistent in games in which the leader finds it to their benefit to reset its control path at a some point in time after the game has begun.

Subgame perfectness is a stronger condition than time consistency, requiring that an equilibrium also be an equilibrium for any possible subgame, "...not only along the equilibrium state trajectory, but also in any (feasible) position...off this trajectory." (Jørgensen and Zaccour 2004, p. 16). A feedback Nash equilibrium that satisfies the Hamilton-Jacobi-Bellman equations, is by construction subgame perfect. Also, the feedback Stackelberg solution is, according to Basar

and Olsder (1995, p. 417), "...strongly time consistent (by definition)", and strong time consistency coincides, at least essentially, with subgame perfectness (Dockner et al. 2000, pp. 106–107).

Differential Game Applications

The differential game framework is designed to model the decisions of multiple decision makers in a continuous-time dynamic context. This framework can be applied to a variety of problem areas of interest and relevance to OR and MS. Furthermore, modeling the passage of time as continuous, rather than discrete, allows the possibility of mathematical, and therefore generalizable, conclusions. This section discusses applications in advertising, pricing, production, and supply chain management.

Advertising

Competitive advertising in the context of dynamics has been especially a popular area of study. Erickson (2003) provides a review. Two particular models of demand evolution have acted as foundations for differential-game applications to advertising. Kimball (1957, pp. 201–202) presents four versions of Lanchester's formulation of the problem of combat, one of which, Model 4,

$$dn_1/dt = k_1 n_2 - k_2 n_1, dn_2/dt = k_2 n_1 - k_1 n_2$$

has become the foundation for what is known as the Lanchester model. Kimball (1957, p. 203) offers the following interpretation of Model 4: "The n_1 and n_2 are then to be interpreted as the numbers of customers for two similar products, while k_1 and k_2 are in essence the amounts of advertising." The Lanchester model in application is generally interpreted in terms of market shares rather than numbers of customers (Erickson 2003, p. 10), so that advertising for a competitor works to attract market share from the competitor's rival.

Vidale and Wolfe (1957) introduce a model of sales evolution for a monopolistic company

$$dS/dt = \beta A(t)(M - S)/M - \lambda S$$

in which $A(t)$ is the advertising rate, S the sales rate, M the maximum sales potential, β an advertising effectiveness coefficient, and λ a sales decay parameter. In the Vidale-Wolfe model, advertising attracts demand from the untapped sales potential, and the sales attracted are subject to decay. Although the Vidale-Wolfe model is defined for a monopolist, it has been adapted for the study of advertising competition.

Many differential-game applications using the Lanchester and Vidale-Wolfe models study open-loop Nash equilibria, since the two models do not readily allow the derivation of subgame-perfect feedback Nash equilibria. Sorger (1989) offers a modification of the Lanchester model that does allow a feedback equilibrium to be derived for duopolistic competitors. Sorger (1989, p. 58) develops a differential game with market-share dynamics

$$\dot{x}(t) = u_1(t)\sqrt{1 - x(t)} - u_2(t)\sqrt{x(t)}, x(0) = x_0.$$

where $\dot{x}(t) = dx/dt$, $x(t)$ is competitor 1's market share, and $u_1(t)$ and $u_2(t)$ are advertising rates for firm's 1 and 2, respectively. The square-root form in the market share equation in the model allows value functions that are linear in the market share state variable, which allows a solution to the Hamilton-Jacobi-Bellman equations for the differential game. Sorger derives both open-loop and feedback equilibria, and finds that the feedback and open-loop equilibria do not coincide.

The Sorger (1989) modification of the Lanchester model allows subgame-perfect feedback Nash equilibria for a duopoly. Feedback equilibria, however, are not achievable in an extension of the Lanchester model to a general oligopoly, in which the number of competitors may exceed two. For an oligopoly, Erickson (2009a, b) provides a modification of the Vidale-Wolfe model that allows the derivation of feedback equilibria. Erickson's (2009a) model has sales dynamics for each oligopolistic competitor i of $n > 2$ total competitors,

$$\dot{s}_i = \beta_i a_i \sqrt{N - \sum_{j=1}^{n} s_j} - \rho_i s_i.$$

In the model, a_i is the advertising rate, s_i the sales rate, N the maximum sales potential, β_i an advertising effectiveness parameter, and ρ_i a sales decay parameter. The expression under the square-root sign

represents untapped potential, that is, the maximum sales potential minus the total sales for all n competitors, including competitor i. An instantaneous change in the sales rate for a competitor comes from two sources: (1) the competitor's advertising attracting sales from the untapped potential in square-root form, (2) a decay from the competitor's current sales rate. Erickson (2009b) extends the model to allow multiple brands for each competitor. As for the Sorger (1989) model, the square-root form in the model allows value functions linear in the state variables, so that the Hamilton-Jacobi-Bellman equations can be solved. Both the Sorger (1989) and Erickson (2009a, b) models are related to a monopolistic modification of the Vidale-Wolfe model suggested by Sethi (1983). Erickson (2009a) uses the derived expressions for feedback Nash equilibrium advertising strategies in an empirical study of the U.S. beer market, and Erickson (2009b) empirically applies the multiple-brand model extension to the carbonated soft drink market.

Pricing

Pricing is a primary and challenging task for management. Prices are the source of revenue for the firm, but also affect demand for the firm's products, especially in a competitive setting. The challenge is compounded when dynamics are involved, and prices are expected not to stay at the same levels. This is the case for new products, in particular new durable products, for which demand tends to develop through a diffusion process that is influenced by the price strategies of competing firms.

Bass (1969) provides a diffusion model of first-time adoption of a new durable product that combines innovation and imitation on the part of customers

$$S(T) = (p + qY(T)/m)(m - Y(T)),$$

where $S(T)$ represents current sales at time T and $Y(T)$ cumulative sales, so that $S(T) = dY(T) / dT$. Further, p is an innovation coefficient, q is an imitation coefficient, and m is the total number of customers who will eventually adopt the new product. The Bass (1969) model has been accepted by much of the OR and MS literature as the primary model of new durable product diffusion.

The Bass (1969) model is for a single firm, and does not consider price explicitly. Dockner and Jørgensen (1988) develop a more general framework for new product diffusion, one that includes competition and prices, which they use to study new-product pricing strategies through differential-game analysis. Dockner and Jørgensen (1988, p. 320) offer the general diffusion model specification

$$\dot{x}_i = f^i(x_1, ..., x_M, p_1, ..., p_M), x_i(0) = x_{i0} \geq 0.$$

In the model, x_i is the cumulative sales volume of competitor $i = 1, 2,\ldots, M$, and the prices $p_1,\ldots, p_M$ of the competitors are assumed to vary with time. To determine their dynamic price strategies, each competitor is assumed to seek to maximize its objective function

$$J^i = \int_0^T e^{-r_i t}(p_i - c_i) f^i dt$$

where unit cost c_i is a nonincreasing function of cumulative sales x_i, as is often the case with new durable products, that unit cost declines with experience. For mathematical tractability reasons, Dockner and Jørgensen (1988) study open-loop Nash equilibria.

Dockner and Jørgensen (1988) derive the necessary conditions for an open-loop Nash equilibrium for their differential game involving the general diffusion model; for further insights, they analyze more specific functional forms. They consider three special cases, competition with price effects only, multiplicative separable price and adoption effects, and adoption effects only with a multiplicative own-price effect.

Production

The management of production quantities and timing is a critical operations function. Dynamics are involved, since production plans may imply that production does not equal customer demand at particular times. This can result in inventories, which need to be carried at a cost, or backlogs, which involve delay in delivery to customers, presumably at a cost to the firm.

Production management can be studied in a competitive context. Eliashberg and Steinberg (1991) consider the dynamic price and production strategies of two competing firms with asymmetric

cost structures. As Eliashberg and Steinberg (1991, p. 1453) explain: "The objective of this paper is to gain insight into the dynamic nature of the competitive aspects of the various policies of two firms, one operating at or near capacity, facing a convex production cost, and the other operating significantly below capacity, facing a linear cost structure. The firms are assumed to face a demand surge condition. We will refer to the firm operating at or near capacity as the 'Production-smoother' and the firm operating below capacity as the 'Order-taker.' "

Eliashberg and Steinberg (1991) define a differential game in which production levels and prices are control variables for the two competing firms, and pursue an open-loop Nash equilibrium. They derive several propositions regarding the equilibrium policies of the two competitors. A particular finding is that the Production-smoother follows a strategy of first building up inventory, then drawing the inventory down, and finishing a seasonal period by engaging in a policy of carrying zero inventory for a positive interval.

Supply Chain Management

A supply chain involves various independent players—e.g., supplier, manufacturer, wholesaler, retailer—as raw materials become products that are distributed to retail locations where final customers are able to buy them. All players have an economic stake in their position in the supply chain that is affected by the decisions of the other players. The interest of supply chain management is in coordination of the decisions of the players, given the players' strategic interdependence.

When dynamics are involved, the interdependence of the players in a supply chain can be interpreted as a differential game. A cooperative differential game would produce full coordination. However, since binding agreements among the supply chain players are difficult to establish and maintain, an alternative focus is to consider noncooperative games with coordinating mechanisms.

One mechanism for achieving coordination is through one of the players in the chain becoming the leader. If there are two players in a supply chain, the differential game becomes a leader-follower game in which a Stackelberg equilibrium provides the coordinating solution. A study that considers this approach is Jørgensen et al. (2001), who analyze the advertising and pricing strategies of two players in a marketing channel, a manufacturer and a retailer.

With the differential game that they develop, Jørgensen et al. (2001) derive four different equilibrium solutions: Markovian (feedback) Nash, feedback Stackelberg with the retailer as the Stackelberg leader, feedback Stackelberg with the manufacturer as the leader, and a coordinated channel solution. They give a detailed comparison of the outcomes for the four solutions.

Concluding Remarks

This article outlines the basic concepts of differential games, along with brief descriptions of relevant applications. More in-depth coverage is given in Dockner et al. (2000) and Jørgensen and Zaccour (2004). Differential games provide a powerful modeling framework for the study of the interaction of multiple decision makers in dynamic settings. As the applications illustrate, the understanding of dynamic and game-theoretic OR and MS problems has been advanced through the analysis of differential-game models.

See

- ▶ Advertising
- ▶ Decision Analysis
- ▶ Dynamic Programming
- ▶ Game Theory
- ▶ Marketing
- ▶ Production Management
- ▶ Supply Chain Management

References

Basar, T., & Olsder, G. J. (1995). *Dynamic noncooperative game theory* (2nd ed.). London: Academic Press.

Bass, F. M. (1969). A new product growth model for consumer durables. *Management Science, 15*, 215–227.

Dockner, E., & Jørgensen, S. (1988). Optimal pricing strategies for new products in dynamic oligopolies. *Marketing Science, 7*, 315–334.

Dockner, E., Jørgensen, S., Long, N. V., & Sorger, G. (2000). *Differential games in economics and management science*. Cambridge, UK: Cambridge University Press.

Eliashberg, J., & Steinberg, R. (1991). Competitive strategies for two firms with asymmetric production cost structures. *Management Science, 37*, 1452–1473.

Erickson, G. M. (2003). *Dynamic models of advertising competition* (2nd ed.). Boston/Dordrecht/London: Kluwer Academic Publisher.
Erickson, G. M. (2009a). An oligopoly model of dynamic advertising competition. *European Journal of Operational Research, 197*, 374–388.
Erickson, G. M. (2009b). Advertising competition in a dynamic oligopoly with multiple brands. *Operations Research, 57*, 1106–1113.
Jørgensen, S., Sigué, S.-P., & Zaccour, G. (2001). Stackelberg leadership in a marketing channel. *International Game Theory Review, 3*, 13–26.
Jørgensen, S., & Zaccour, G. (2004). *Differential games in marketing*. Boston/Dordrecht/London: Kluwer Academic Publishers.
Kamien, M. I., & Schwartz, N. L. (1991). *Dynamic optimization: The calculus of variations and optimal control in economics and management*. Amsterdam/New York/London/Tokyo: North-Holland.
Kimball, G. E. (1957). Some industrial applications of military operations research methods. *Operations Research, 5*, 201–204.
Nerlove, M., & Arrow, K. J. (1962). Optimal advertising policy under dynamic conditions. *Economica, 39*, 129–142.
Sethi, S. P. (1983). Deterministic and stochastic optimization of a dynamic advertising model. *Optimal Control Applications and Methods, 4*, 179–184.
Sorger, G. (1989). Competitive dynamic advertising: A modification of the case game. *Journal of Economic Dynamics and Control, 13*, 55–80.
Vidale, M. L., & Wolfe, H. B. (1957). An operations research study of sales response to advertising. *Operations Research, 5*, 370–381.

Diffusion Approximation

A heavy-traffic approximation for queueing systems in which the infinitesimal mean and variance of the underlying process are used to develop a Fokker-Planck diffusion type differential equation which is then typically solved using Laplace transforms.

See

► Queueing Theory

Diffusion Process

A continuous-time Markov process on $\mathbb{R}$ or $\mathbb{R}''$ which is analyzed similar to a continuous-time physical diffusion.

Digital Music

Elaine Chew
Queen Mary University of London, London, UK

Introduction

The advent of digital music has enabled scientific approaches to the systematic study, computational modeling, and explanation of human abilities in music perception and cognition, and in music making, which includes the activities of music performance, improvisation, and composition. The move from analog to digital music, and from music stored on a compact disc to music streamed live over the Internet, has brought new engineering challenges, innovation opportunities, and creative outlets. The pervasiveness of computing power and the Internet has changed the ways in which people interact with, and make, music. The research communities at the cusp of music science and engineering came about after the turn of the last millennium, and have been increasing exponentially since. A short list of the communities involved in scholarly pursuits in music science and engineering is provided in Chew (2008).

Impact of Digital Music Research

Science and technology has changed the face of arts and humanities scholarship. Advances in digital music technology have enabled new discoveries by harnessing the computational power of modern computers for music scholarship. For example, the Joyce Hatto scandal, documented in *The Economist* and elsewhere in 2007, in which over 100 CDs released in recent years under her name were in fact the work of other pianists, was unveiled in part because of the machinery available to automatically evaluate and compare recordings of musical works. The technology exists to begin mapping the myriad decisions involved in composing and performing music, and to start charting human creativity. The fact that mathematical models, and by extension operations research (OR) methods, are widely applied in digital music research and practice should come as

no surprise, given the historical connections between music, mathematics, and computing.

The music technology industry has emerged as a major economic force. The phenomenal explosion in digital music information has led to the need for new technologies to organize, retrieve, and navigate digital music databases. Examples of major advances in the organizing and retrieval of digital music include Pandora, a personalized Internet radio service that helps people discover new music according to their tastes, and Shazam, a service that helps people identify and locate the music they are hearing. Pandora generates a playlist based on an artist or song entered by the user, and refines future recommendations based on user preference ratings of the songs in that list. Shazam identifies the song and artist, and the precise recording, from a musical excerpt supplied by the user over a device such as an iPhone. In both Pandora and Shazam, the user is offered the opportunity to purchase the song that is playing, or that has been identified, from various vendors. As of 2010, Pandora had 50 million registered users, and more than 1 billion stations, covering 52% of the Internet radio market share. In December 2010, Shazam announced that it has surpassed 100 million users in 200 countries.

Any young or young-at-heart person may be familiar with the music video game, Guitar Hero®, which allows everyone to live the dream of being a rock star in their own living room by pushing colored buttons on the guitar interface in sync with approaching knobs in the video screen. In a few short years, Guitar Hero took over a significant share of the video game market, grossing over two billion dollars by 2009 and leading to it being featured in a South Park television episode. Bands featured in the game — owned and marketed by Activision — experience significant increases in song sales, so much so that major labels vie for their music to be included in new versions of it and in its successor, Rock Bandpt® vie for their music to be included in new versions.

Music Structure

The understanding of music structure is fundamental to computer analysis of music, and a precursor to digital music processing and manipulation. Music consists of organized sounds with perceptible structures in both time and frequency domains. Often, music can be considered to comprise of a sequence of tones, or several concurrent sequences of tones. Each tone has properties such as pitch (the perceived fundamental frequency of the tone), duration, timbre, and loudness. Much of the music that is heard consists of more than a single stream of tones. When hearing multi-tone textures, the ear can segregate the collection of sounds into streams. The most prominent of these streams is often considered to be the melody of the music piece. Structures relating to individual streams as they progress over time are sometimes referred to as horizontal structures. Like language, music streams can be segmented into phrases. Salient tone patterns in music phrases form motifs, short patterns that recur and vary throughout the piece. The varying of these patterns forms the surface structure of the music piece.

Overlapping pitches in the overlay of multiple tone sequences form chords; conversely, one could say that chords consist of the synchronous sounding of two or more pitches. Chords constitute mid-level structure in music. Structures, such as chords, that relate to synchronous sounds or chunks of music over overlapping streams are sometimes referred to as vertical structures. In Western tonal music, the pitches and durations and their ordering generates the perception of pitch stability relative to one another. This pattern of perceived stability is set up as soon as the ear hears as few as only three to four tones in the sequence. The most stable pitch is the name of the key of the tone sequence. The key, in turn, implies adherence to the pitch set of the scale. The pitches in a scale have varying levels of perceived stability, the result of the physics of sound, the physiology of the ear, or exposure to music. The varying of the most stable pitch over time forms the deep structure of the piece.

The structure of a musical piece can also be conceptualized as a sequence of section labels such as AB (binary form), ABA (ternary form), ABACAC′ADA (a sample rondo form), and intro-(verse-chorus)^n-outro (a common popular music form). While some composers, when writing in a particular genre, choose to adopt a particular form for a composition, structure can also emerge from choices made in composition or improvisation to manage a listener's attention.

Sequences of durations, or sequences of inter-onset-intervals, form rhythms. Periodic onsets

generate perceived beats, and accent and stress patterns in beat and in rhythm sequences. The periodic accent patterns in beat sequences, in turn, result in meter. For example, there are cyclic patterns of four beats in the march with a strongest-weak-strong-weak accent pattern, whereas each of the four beats in a tango is subdivided into two with a resulting strong-weak-weak-strong-weak-weak-strong-weak accent pattern. Conversely, the meter of a composition often implies a persistent periodic accent pattern. The beat rate charts the tempo of the music: a high beat rate results in fast music, and a low beat rate results in slow music. Like many things in art, it is deviations from the norm that form the core of artistic expression. Thus, a large part of expressive musical performance is the art of systematically varying the tempo, and deviating from an underlying time grid. For example, not playing the beats as notated is essential to playing a convincing swing rhythm. Other important parameters of variation in expressive performance include loudness and timbre.

Structure guides expressive decisions in performance, and expressive performance, in turn, influences structure. For example, a performer may choose to emphasize unusual key changes by slowing down the tempo and dramatically reducing the loudness of the sound produced at the juncture of change. Alternatively, by punctuating the playback of a tone stream with judicially placed accents and pauses, the performer can impute phrase and motivic structure on a music stream.

Music problems can be broadly categorized into the areas of analysis, performance, and composition and improvisation. When the problems are concerned with human abilities in music making and listening, they also touch upon the area of music perception and cognition. It is beyond the scope of this article to give a comprehensive survey of problem formulations and solutions in computational modeling of music. Rather, this article focuses on representative problems in each category and solutions, covering some essential background on music representation and computation.

Computational Music Analysis

The goal of computational music analysis is to automatically abstract structures, such as those described above, from digital music.

Key and Harmony

The determination of key is a problem in the detection of vertical pitch structure. Key finding (a.k.a. tonal induction) can be described as the problem of finding the note on which a music piece is expected to end. The most stable pitch in a key is also the one that is expected to end a piece of music in that key. Key finding is an important step preceding a number of music applications such as automatic music transcription, accompaniment, improvisation, and similarity assessment. While the focus here is key finding, it is worthwhile to mention chord tracking, a related problem for which the solution bears similarities to key finding. A survey of automatic chord analysis algorithms can be found in Mauch (2010).

Key Finding Using Correlation: Key is most often inferred from pitch information. Each pitch can be represented as an integer, according to pitch height. For example, in MIDI (musical instrument digital interface) notation, the pitches A, B♭, B, C in the middle range of the piano keyboard are represented as 57, 58, 59, 60. Pitches repeat on the keyboard, and the twelfth tone above C is C again, one octave higher. Sometimes only the pitch class is of interest, and pitch numbers can be collapsed into pitch classes using modulo arithmetic. If p is a pitch number, then the corresponding pitch class is $p \mod 12$.

Key-finding algorithms tend to match music data with templates representing the prototypical profile for the 24 major and minor keys. A key-finding algorithm by Krumhansl and Schmuckler (described in Krumhansl 1990) compares a vector, $\mathbf{d} = [d_i]$, summarizing total note duration for each of the twelve pitch classes, to experimentally obtained probe tone profiles for each of the major and minor keys, $\mathbf{v}_i = [v_{ij}]$ for $i = 1 \ldots 24$, by calculating their correlation coefficients, $\rho_{\mathbf{dv}_i}$. Each probe tone profile is generated by playing a short sequence of chords to establish the key context, then having listeners rate (on a scale of 1 to 7) how well a probe tone that is then played fit in the context. The best match key probe tone profile is the one having the highest correlation coefficient with the query vector, i.e.

$$\arg\max_i \rho_{\mathbf{dv}_i} = \arg\max_i \frac{\sigma_{\mathbf{dv}_i}}{\sigma_{\mathbf{d}}\sigma_{\mathbf{v}_i}}.$$

Creating Spatial Models: Having a spatial model that mirrors the mental representation of tonal space is

something that is of interest not only to cognitive scientists, but also to computational scientists who use these spaces to design algorithms for tonal induction. Kassakian and Wessel (2005) proposed a convex optimization solution for incrementally creating spatial representations of musical entities, such as key and melody, in Euclidean space in such a way as to satisfy a set of dissimilarity measures. Assuming the existing elements to be $\mathbf{r}_i \in \mathbb{R}^n$ and the vector of dissimilarity distances between the new element and existing ones to be $\mathbf{s} = [s_i] \geq 0$, where $i = 1, 2, \ldots, m$. The problem then becomes one of finding

$$\arg\min_{\mathbf{x},\gamma} \sum_{i=1}^{m} (||\mathbf{x} - \mathbf{r}_i|| - \gamma s_i)^2.$$

Using the geometric insight that each $(||\mathbf{x} - \mathbf{r}_i|| - \gamma s_i)$ is the optimal value of $\min_{\mathbf{b}_i} ||\mathbf{x} - \mathbf{b}_i||^2$ for some $\mathbf{b}_i \in \mathbb{R}^n$ inscribed on the ball of radius γs_i around the point $\mathbf{r}_i$, the problem can be re-written as:

$$\begin{aligned} \min_{\mathbf{x},\gamma,\mathbf{b}} \quad & ||\mathbf{J}\mathbf{x} - \mathbf{b}||^2 \\ \text{s.t.} \quad & ||\mathbf{r}_i - \mathbf{b}_i||^2 = \gamma^2 s_i^2, i = 1, 2, \ldots, m \\ \text{where} \quad & \mathbf{b} \equiv [b_1^T, b_2^T, \ldots, b_m^T]^T \in \mathbb{R}^{mn} \\ \text{and} \quad & \mathbf{J} \equiv [I, I, \ldots, I]^T \in \mathbb{R}^{mn \times n} \end{aligned}$$

While the primal problem is not convex, the dual obtained by Lagrangian relaxation is convex, as is the dual of the dual. The authors used a semi-definite programming solver to obtain a solution to the dual of the dual. Because the dual's dual is a relaxation of the primal, they computed a primal feasible solution from the relaxation using a randomized method reported by Goemans and Williamson, and generalized by Nesterov. The problem can also be solved using more conventional gradient descent methods.The resulting key space map generated in this fashion corresponds well to Krumhansl's map created using multi-dimensional scaling (Krumhansl 1990).

Key Finding Using Geometric Spaces: Starting from a model of tonal space that concurs with human perception can be an advantage in the design of computational algorithms for key finding. Observing that the pitch classes in a major key and in a minor key each occupy distinctly shaped compact spaces on the harmonic network or tonnetz, Longuet-Higgins, and Steedman (1971) proposed a shape matching algorithm to determine key from pitch class information.

The tonnetz is a network model for pitch classes where horizontal neighbors are pitch classes whose elements have a fundamental frequency ratio of approximately 2:3 (four major/minor scale steps apart), neighbors on the northeast diagonal have a ratio of approximately 4:5 (two major scale steps apart), and neighbors on the northwest diagonal have a ratio of approximately 5:6 (two minor scale steps apart). The dual graph of the harmonic network connects all triads (three-note chords) sharing two pitches, the transition between which has the property of smooth voice leading. Lewin (1987) lays the foundation for the theory underlying transformations on this space in his treatise on Generalized Intervals and Transformations. Callendar, Quinn, and Tymoczko (Tymoczko 2006; Callender et al. 2008) further generalized these chord transition principles to non-Euclidean space.

The tonnetz is inherently a toroid structure. By rolling up the planar network so that repeating pitch classes line up one on top of another, one gets the pitch class spiral configuration of the harmonic network. Inspired by interior point approaches, Chew (2000) proposed the spiral array model, which uses successive aggregation to generate higher level representations, inside this three-dimensional structure, from their lower level components. For example, if pitch classes were indexed by their positions on the line of fifths, then each pitch classes can be represented as:

$$\mathbf{P}_{k+1} \equiv \mathbf{R} \cdot \mathbf{P}_k + \mathbf{h},$$

$$\text{where } \mathbf{R} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{h} = \begin{bmatrix} 0 \\ 0 \\ h \end{bmatrix}, k \in \mathbb{Z}.$$

The positions of major and minor chords are computed as convex combinations of their component pitches:

$$\mathbf{C}_{M,k} \equiv \omega_1 \cdot \mathbf{P}_k + \omega_2 \cdot \mathbf{P}_{k+1} + \omega_3 \cdot \mathbf{P}_{k+4},$$

and

$$\mathbf{C}_{m,k} \equiv u_1 \cdot \mathbf{P}_k + u_2 \cdot \mathbf{P}_{k+1} + u_3 \cdot \mathbf{P}_{k-3},$$

respectively, where $\omega_1 \geq \omega_2 \geq \omega_3 > 0$, $u_1 \geq u_2 \geq u_3 > 0$, $\sum_{i=1}^{3} \omega_i = 1$, and $\sum_{i=1}^{3} u_i = 1$. Major and minor keys are generated from the weighted average of their defining chords:

$$\mathbf{T}_{M,k} \equiv \omega_1 \cdot \mathbf{C}_{M,k} + \omega_2 \cdot \mathbf{C}_{M,k+1} + \omega_3 \cdot \mathbf{C}_{M,k-1},$$
$$\mathbf{T}_{m,k} \equiv \upsilon_1 \cdot \mathbf{C}_{M,k} + \upsilon_2 \cdot [\alpha \cdot \mathbf{C}_{M,k+1} + (1-\alpha) \cdot \mathbf{C}_{m,k+1}] + \upsilon_3 \cdot [\beta \cdot \mathbf{C}_{m,k-1} + (1-\beta) \cdot \mathbf{C}_{M,k-1}],$$

where $\omega_1 \geq \omega_2 \geq \omega_3 > 0$, $\upsilon_1 \geq \upsilon_2 \geq \upsilon_3 > 0$, $\sum_{i=1}^{3} \omega_i = 1$, $\sum_{i=1}^{3} \upsilon_i = 1$, and $0 \geq \alpha \geq 1$, $0 \geq \beta \geq 1$. The calibration of the spiral array, finding solutions to the variables that satisfy perceived properties of pitch relations, is a nonlinear constraint satisfaction problem for which the author found near-feasible solutions using a gradient-inspired heuristic.

Given a music sequence of pitches that map to the pitch representations $\{\mathbf{P}_i\}$, with corresponding durations, $\mathbf{d} = [d_i]$, where $i = 1, \ldots, m$, the center of effect of the sequence, $\text{CE} \equiv \sum_{i=1}^{m} d_i \cdot \mathbf{P}_i$. The most plausible key for the sequence is given by the key representation nearest to the center of effect of the sequence:

$$\arg \min_{\mu \in \{M,m\},k} ||\text{CE} - \text{T}_{\mu,k}||.$$

Extensions: The descriptions of key-finding algorithms have focussed on discrete information. It is possible to apply probabilistic approaches using the same representations. For example, Temperley (2007) explores a Bayesian approach to the Krumhansl key-finding framework.

Both Krumhansl's probe tone profile method and Chew's spiral array center of effect generator algorithm have been extended from symbolic to audio key finding. The underlying methodology remains the same. However, when starting from audio, some pre-processing of the signal needs to be done to convert it to pitch class information. Similarly, the key templates may have to be adapted for audio input. Common techniques for extracting frequency information from the signal include the Fast Fourier Transform and the Constant-Q Transform. This step is followed by the mapping of spectral information to pitch class bins, then the key finding algorithm is applied accordingly. While signal-based information tends to be more noisy than discrete symbolic information, much of the noise results from the harmonics of the fundamental frequency of each tone, which tend to be frequencies in the key, and help reinforce and stabilize key identity.

Meter and Rhythm

While historically the modeling of meter and rhythm has not received as much attention as that of key and harmony, the feeling of pulse, and the grouping of events embedded in that pulse, are some of the most visceral responses humans have to music. An overview of symbolic and literal (signal) representations of rhythm can be found in Sethares (2007) and Smith and Honing (2008). In symbolic music, tone onsets are encoded explicitly in the representation. When analyzing audio, a pre-processing step of extracting onset information must first be performed. An overview of onset detection methods is given in Bello et al. (2003).

Meter Induction: The determining of meter can be described as the finding of the periodic accent patterns in the underlying pulse of music. Meter induction, like key finding, is an important step for numerous music applications such as automatic music transcription, generation, and accompaniment. Most algorithms for finding meter apply autocorrelation to find periodicity in the signal, see for example, Gouyon and Dixon (2006). A different computational model for extracting meter from onset information is described in Mazzola's extensive volume on mathematical music theory (Mazzola 2002), and expanded by Volk (2008) to investigate local versus global meters.

The solution method is restated here in a slightly different format. Suppose $\mathbb{N}$ indexes the smallest grid possible to capture all event onsets in a score. And suppose we are interested in pulse layers at onset times of all possible periodicities, $g \in \mathbb{N}$, and offsets, $f = 0, \ldots, i-1$, then a pulse layer might be indexed by $y = \frac{1}{2} \cdot g(g-1) + 1 + f$ and be represented as a vector $\mathbf{p}_y = [p_{yi}]$, where

$$p_{yi} = \begin{cases} 1 \text{ if } i \in \{gk - f : k \in \mathbb{N}\}, \\ 0 \text{ otherwise.} \end{cases}$$

Suppose the onsets in the music are represented as a vector, $\mathbf{o} = [o_i]$, where

$$o_i = \begin{cases} 1 \text{ if an onset occurs on that grid marking, and} \\ 0 \text{ otherwise,} \end{cases}$$

$$p^o_{yi} = \begin{cases} 1 \text{ if } (p_{yi} = 1) \cap (o_i = 1), \text{ and} \\ \text{otherwise.} \end{cases}$$

Effectively, $\mathbf{p}^o_y$ serves as an indicator function for when an onset in the music coincides with a pulse at layer y. Introducing one more variable, let ℓ_{yi} be the span of the longest chain of ones surrounding p^o_{yi}. ℓ_{yi} can be defined recursively as follows:

$$\ell_{yi} = \ell^R_{yi} + \ell^L_{yi},$$

$$\text{where } \ell^R_{yi} = \begin{cases} 0 & \text{if} \quad p^o_{yi} = 0, \\ 1 + \ell_{yi+1} & \text{if} \quad p^o_{yi+1} = 1, \end{cases}$$

$$\ell^L_{yi} = \begin{cases} 0 & \text{if} \quad p^o_{yi} = 0, \\ 1 + \ell_{yi-1} & \text{if} \quad p^o_{yi-1} = 1. \end{cases}$$

The metric weight of an onset at time i is then given by

$$w_i = \sum_y \ell^a_{yi}.$$

The resulting vector, $\mathbf{w}$ gives a profile of the accents and reveals the periodicity in the rhythm. Recall that $y = \frac{1}{2} \cdot g(g-1) + 1 + f$. A variation on this technique (Nestke and Noll 2001) assigns the weight ℓ_{yi} to all points on pulse layer y, i.e. $\forall i = gk - f, k \in \mathbb{N}$.

Genre Classification using Metric Patterns: Periodicity patterns are one of the defining characteristics of dance music, and this feature has been used to classify music into different genres such as tango, rumba, and cha cha (Dixon et al. 2003; Chew et al. 2005). Dixon et al. (2003) uses a set of rules, which can be implemented using decision trees, to categorize the music using tempo and periodicity features. Similar to the key-finding methods, (Chew et al. 2005) uses correlation to compare the metric weight profiles derived from the data to templates for each dance category.

Segmentation in Time

Few pieces of music stay entirely in one key or one rhythmic pattern. Composers generate interest by varying the tonal and rhythmic content of the music over time. Thus, it would be unrealistic to compute only one key or one meter based on available information. A common adaptation of key-finding or meter induction algorithms to allow for changing key or metric identity is to use a sliding window (Shmulevich and Yli-Harja 2000), or an exponential decay function (Chew and François 2005).

The determining of section boundaries is important in music structure analysis, the applications for which include music summarization. Using the key and meter representation frameworks introduced above, it is possible to create a dynamic programming formulation, with an appropriate penalty function for change between two adjacent windows, for assigning boundaries in a piece of music, for example for key as discussed in Temperley (2007). Another method for determining key change is described in Chew (2002), which borrows ideas from statistical quality control. In large structure analysis, it is often useful to be able to label sections (for example, as chorus or verse in popular songs). Toward this end, Levy and Sandler (2008) have applied a number of clustering techniques to audio features extracted from music signal.

Melody

Melody represents the horizontal structure of music. Apart from the straightforward event string representation of melody, melody can also be decomposed into building blocks and represented as grammar trees, as prescribed by Lerdahl and Jackendoff (1983).

Similarity Assessment: Quantifying the similarity between two melodies is important for music information retrieval. Typke et al. (2003) describe the use of the Earth Mover's Distance (EMD) to quantify melodic similarity. Represent each melody as weighted points in pitch-time space, for example, melody $\mathrm{A} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m\}$ and melody $\mathrm{B} = \{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n\}$ with respective weights, $\omega_i, u_j \in \mathbb{R}^+ \cup \{0\}$, where $i = 1, \ldots m$ and $j = 1, \ldots n$. The similarity measure between the two melodies is the EMD, the minimum cost flow to transform one melody into another by moving weight from one point in A to one point in B, where the cost is the weight moved times the distance traveled.

Suppose $W = \sum_{i=1}^{m} w_i$ and $U = \sum_{i=1}^{n}$, and f_{ij} is the flow of weight from a_i to b_j over the distance d_{ij}. The problem can thus be stated as:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}d_{ij} \\ \text{s.t.} \quad & \sum_{j=1}^{n} f_{ij} \le w_i, i = 1,\ldots,m, \\ & \sum_{i=1}^{m} f_{ij} \le u_j, j = 1,\ldots,n, \\ & \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = \min(W,U), \\ & f_{ij} \ge 0,\ i = 1,\ldots,m,\ j = 1,\ldots,n, \end{aligned}$$

which can be solved using linear programming, and

$$\mathrm{EMD}(\mathrm{A},\mathrm{B}) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}^{*} d_{ij}}{\min(W,U)}.$$

Stream Segregation: A number of approaches have been proposed to tackle the problem of automatically separating a polyphonic (multi-line) music texture into its component voices. An example might be to separate a fugue by Johann Sebastian Bach into its four parts. A randomized local search method to optimize a parametric cost function that penalizes undesirable traits in a voice-separated solution was proposed by Kilian and Hoos (2002). Chew and Wu (2004) proposed a contig-mapping approach to first break a piece of music into contigs with overlapping fragments of music. Then, exploiting perceptual principles such as voices tend not to cross in maximal voice contigs, the algorithm re-connects the fragments in neighboring contigs using distance minimization.

Composition and Improvisation

The use of mathematical models in music composition has become an active area for musical innovation since Xenakis (2001), who used stochastic processes, probabilistic models, and game theory to guide his compositions. With widespread access to computing to help solve music composition mathematical problems, computer-assisted composition has emerged as a useful tool to help composers create new music, as well as an important area of digital music research.

Constraints

A number of music composition problems can be naturally described as constraint satisfaction problems (CSPs). Solution methods for CSPs include combinatorial optimization and local search techniques such as Tabu search, simulated annealing, and genetic algorithms.

Truchet and Codognet (2004) list fourteen examples of musical CSPs and propose to apply a heuristic adaptive search technique to solve the CSPs. An example of a compositional CSP is as follows: Given a sequence of chords, suppose the composer is interested in finding an ordering of the sequence such that two adjacent chords have the maximal number of common tones. If the chords were represented as nodes, and the distance between any two nodes is the number of common tones, then the problem of interest takes the form of the Traveling Salesman Problem. Every chord must be visited once, and the desired solution must minimize $(-1) \times$ distance.

Related to this is the classic problem of providing harmonization for a given melody. The most widely used solution method for generating a score from a melody is via constraints, and a variety of approaches and results are reviewed in Pachet and Roy (2004).

Markov Chains and Other Network Models

The use of Markov chains (MCs) forms another solution method that is commonly used in the generating of music. In the case of MCs, the probabilities are estimated from existing data, and used to generate music in the style of the training data set. Farbood and Schoner (2001) use MCs to generate music in the style of Palestrina. Using the tonnetz as scaffolding to reduce the search space, Chuan and Chew (2007a) use MCs to generate style-specific accompaniment for melodies given only a few examples. MCs are excellent models for imitating local structure, but lack high level structure knowledge to guide the shaping of a composition. To remedy this deficiency, researchers have considered computer systems that create the local surface structure while relegating higher level structural control to humans.

In Pachet's Continuator, the system builds prefix trees from music data, weights each possible continuation with a probability estimated from the data, and uses the resulting MC to generate music in dialog with a human musician. Extensions of the basic MC model consider hierarchical representations and ways of imputing rhythmic structure to the resulting music. Assayag and Dubnov (2004) describe an alternate approach using factor oracles. The suffix links in the resulting network model is assigned transition probabilities that causes the original music material to be recombined smoothly. Using the same factor oracle approach, François et al. (2010) created Mimi4x, an installation that allows users to make high-level structural improvisation decisions while the computer manages surface details on four improvising systems.

Expressive Music Performance

Music is rarely performed as notated. The score is an incomplete description of the experience of a music piece, and leaves much to interpretation by a performer. In expressive music performance, a performer manipulates parameters such as tempo, loudness, and articulation for expressive or interpretive ends, and to guide the listener's perception of groupings and meter. The expressive devices in the performance of music is sometimes called musical prosody. See Palmer and Hutchins (2006) for a definition and review of research on musical prosody. The extraction of performance parameters can be viewed as the continuous monitoring of expressive features such as tempo and loudness over time.

Representation

Tempo and loudness are two important features of music performance. Suppose the list of onsets in the performed music are $O = \{o_0, o_1, \ldots, o_n\}$. Then the inter-onset-interval at time i is $IOI_i = o_i - o_{i-1}$. If a listener sat and tapped along to the beat of the music, then the list of beat onsets might be $B = \{b_0, b_1, \ldots, b_n\}$. The interbeat-interval would be $IBI_i = b_i - b_{i-1}$, and the instantaneous tempo would be $T_i = \frac{1}{IBI_i}$. Often, some smoothing is desired, and one would report a moving average for the smoothed tempo. Sometimes, the acceleration is desired, where $a_i = \Delta T_i = T_i - T_{i-1}$. A number of models for deriving loudness from the signal exist, many of which have been implemented in Matlab. Timmers (2005) surveys some ways of measuring tempo and loudness in musical performance and of comparing them across performances.

Using the tempo-loudness representation proposed by Langner and Goebl, Dixon et al. (2002) created a computer system for for real-time visualization of performance parameters in the Performance Worm. The exploration of Langner's tempo-loudness space for performance analysis led to its use for performance synthesis in the Air Worm (Dixon et al. 2005).

In the spirit of annotations of speech prosody, Raphael proposed a series of markup symbols for expressing musical flow (Raphael 2009). The symbols consist of

$$\{l^-, l^\times, l^+, l^\rightarrow, l^\leftarrow, l^*\}.$$

$\{l^-, l^\times, l^+\}$ denote a sense of arrival, where l^- is a direct and assertive stress, $l^\times$ is a soft landing that relaxes upon arrival, and l^+ is an arrival whose momentum continues to carry forward into the future. $\{l^\rightarrow, l^*\}$ mark notes that continue to move forward toward a future goal, $l^\rightarrow$ is a passing tone and l^* is a passing stress, and $\{l^\leftarrow\}$ denotes a pulling back movement. Because it is hard to determine the exact sets of tempo and loudness parameters, and more locally, the exact amounts of delay or anticipating of an onset, that lead to these flow sensations, Raphael uses a hidden Markov model (HMM) to estimate the most likely hidden variables to have given rise to the observed prosodic annotation.

Phrases

In expressive performance, performers indicate phrase groupings by varying tempo (accelerate and decelerate at beginnings and ends of phrases) and/or loudness (crescendo and decrescendo at beginnings and ends of phrases). As a result, this aspect of a performer's interpretations can be directly inferred from tempo and loudness data. For example, Chuan and Chew (2007b) propose a dynamic programming (DP) method for automatic extraction of phrases. The authors test a model that fits a series of quadric curves (first modeled by polynomials of degree two, then by a series of quadratic splines) to the tempo time series. The best fit curve is found using quadratic programming, and the phrase boundaries are determined using DP.

Alignment

A common use of DP in music processing is in the alignment of music sequences that may be in the same or different format. Arifi et al. (2004) reviews the state of the art, and describes an algorithm for aligning music sequences in two of three possible formats – score, Musical Instrument Digital Interface (MIDI), and pulse-code modulation (PCM) audio format.

Assuming the two sequences are the score, $\mathbf{s} = [s_i]$, and a PCM representation of the audio performance, $\mathbf{p} = [p_j]$. The first task is to generate a cost matrix for aligning any point, s_i, in the score with any point, p_j, in the PCM audio. In Arifi et al. (2004), the distance minimization step is embedded in the cost matrix. Suppose the cost matrix is represented by $\mathbf{C} = [c_{ij}]$, each element of which expresses the cost minimizing SP-match for $[s_1, s_2, \ldots, s_i]$ and $[p_1, p_2, \ldots, p_j]$, i.e.

$$c_i j = \min\left\{c_{i,j-1}, c_{i-1,j}, c_{i-1,j-1}, d_{ij}^{SP}\right\}.$$

Then, the algorithm for synchronizing the two streams is as follows:

```
SCORE-PCM-SYNCHRONIZATION(C, s, p)
1    i = length(s), j = p, SP-Match = 0
2    while (i > 0) and (j > 0)
3        do if c[i, j] = c[i, j − 1]
4                then j = j − 1
5             else if c[i, j] = c[i − 1, j]
6                     then i = i − 1
7             else SP-Match = SP-Match ∪
                          {(i, j)}, i = i−1,  j = j−1
8    return SP-Match
```

Dixon and Widmer (2005) introduced MATCH, a tool chest for efficient alignment of two time series using variations on the classic dynamic time warping (DTW) algorithm. Niedermayer and Widmer (2010) proposed a multi-pass algorithm that uses anchor notes (notes for which the alignment confidence is high) to correct inexact matches.

Concluding Remarks

Digital music research has rapidly evolved with computing advances and the increasing possibilities for connections between music and computing. The latest advances in the field are reported in the annual *Proceedings of the International Conference on Music Information Retrieval*, *Proceedings of the Sound and Music Computing Conference*, and the *Proceedings of the International Symposium on Computer Music Modeling and Retrieval*, the biennial *Proceedings of the International Conference on Mathematics and Computation in Music*, and the occasional *Proceedings of the International Conference on Music and Artificial Intelligence*. They can also be found in the traditional conferences of the multimedia, databases, human computer interaction, and audio signal processing communities. The archival journals include the *Computer Music Journal*, the *Journal of New Music Research*, and the *Journal of Mathematics and Music*.

There exist close ties between digital music research and the fields of music perception and cognition and computer music (which places greater emphasis on the creating of music), and the community of researchers interested in interfaces for musical expression. Work that overlaps with these other areas can be found in the biennial *Proceedings of the International Conference on Music Perception and Cognition*, and the annual *Proceedings of the International Computer Music Conference* and *Proceedings of the International Conference on New Interfaces for Musical Expression*.

See

- ▶ Constraint Programming
- ▶ Dynamic Programming
- ▶ Linear Programming
- ▶ Markov Chains
- ▶ Mathematical Programming

References

Arifi, V., Clausen, M., Kurth, F., & Müller, M. (2004). Score-to-PCM music synchronization based on extracted score parameters. In *Proceedings of the international symposium on computer music modeling and retrieval*, Esbjerg, Denmark, pp. 193–210.

Assayag, G., & Dubnov, S. (2004). Using factor oracles for machine improvisation. *Soft Computing, 8*(9), 604–610.

Bello, J. P., Daudet, L., Abdallah, S., Duxbury, C., Davis, M., & Sandler, M. B. (2003). A tutorial on onset detection in music signals. *IEEE Transactions on Speech and Audio Processing, 13*(5), 1035–1047.

Callender, C., Quinn, I., & Tymoczko, D. (2008). Generalized voice-leading spaces. *Science, 320*(5874), 346–348.

Chew, E. (2000). *Towards a mathematical model of tonality* (Ph.D. dissertation, MIT Press, Cambridge, MA).

Chew, E. (2002). The spiral array: An algorithm for determining key boundaries. In *Music and artificial intelligence – Second international conference*. Springer LNCS/LNAI, Vol. 2445, pp. 18–31.

Chew, E. (2008). Math & music – The perfect match. *OR/MS Today, 35*(3), 26–31.

Chew, E., & François, A. R. J. (2005). Interactive multi-scale visualizations of tonal evolution in MuSA.RT Opus 2. *ACM Computers in Entertainment, 3*(4), 1–16.

Chew, E., & Wu, X. (2004). Separating voices in polyphonic music: A contig mapping approach. In *Proceedings of the international symposium on computer music modeling and retrieval*, Vol. 2, Esbjerg, Denmark, pp. 1–20.

Chew, E., Volk, A., & Lee, C.-Y. (2005). Dance music classification using inner metric analysis: A computational approach and case study using 101 Latin American Dances and National Anthems. In B. L. Golden, S. Raghavan, & E. A. Wasil (Eds.), *The next wave in computing, optimization, and decision technologies: Operations research computer science interfaces series*, New York: Springer, (Vol. 29, pp. 355–370).

Chuan, C. -H., & Chew, E. (2007a). A hybrid system for automatic generation of style-specific accompaniment. In *Proceedings of the international joint workshop on computer creativity*, London, UK, p. 4.

Chuan, C. -H., & Chew, E. (2007b). A dynamic programming approach to the extraction of phrase boundaries from tempo variations in expressive performances. In *Proceedings of the international conference on music information retrieval*, Vienna, Austria, p. 8.

Dixon, S., & Widmer, G. (2005). MATCH: A music alignment tool chest. In *Proceedings of the international conference on music information retrieval*, London, UK, pp. 492–497.

Dixon, S., Goebl, W., & Widmer, G. (2002). The performance worm: Real time visualisation of expression based on Langner's tempo-loudness animation. In *Proceedings of the international computer music conference*, Göteborg, Sweden, pp. 361–364.

Dixon, S., Goebl, W., & Widmer, G. (2005). The 'air worm': An interface for real-time manipulation of expressive music performance. In *Proceedings of the international computer music conference*, Barcelona, Spain, pp. 614–617.

Dixon, S., Pampalk, E., & Widmer, G. (2003). Classification of dance music by periodicity patterns. In *Proceedings of the international conference on music information retrieval*, Baltimore, Maryland.

Farbood, M., & Schoner, B. (2001). Analysis and synthesis of Palestrina-style counterpoint using Markov chains. In *Proceedings of the international computer music conference* Havana, Cuba, p. 27.

François, A. R. J., Schankler, I., & Chew, E. (2010). Mimi4x: An interactive audio-visual installation for high-level structural improvisation. In *Proceedings of the international conference on multimedia and expo*, Singapore, pp. 1618–1623.

Gouyon, F., Dixon, S. Computational Rhythm Description. 2006. Tutorial on computational rhythm description. *In International Conference of Music Information Retrieval*.

Kassakian, P., & Wessel, D. (2005). Optimal positioning in low-dimensional control spaces using convex optimization. In *Proceedings of the international computer music conference*, Barcelona, Spain, Vol. 31, pp. 379–382.

Kilian, J., & Hoos, H. (2002). Voice separation: A local optimisation approach. In *Proceedings of the international conference on music information retrieval*, Paris, France, Vol. 3, pp. 39–46.

Krumhansl, C. L. (1990). *Cognitive foundations of musical pitch*. New York: Oxford University Press.

Lerdahl, F., & Jackendoff, R. A. (1983). *A generative theory of tonal music*. Cambridge, MA: MIT Press.

Levy, M., & Sandler, M. (2008). Structural segmentation of musical audio by constrained clustering. *IEEE Transactions on Audio, Speech, and Language Processing, 16*(2), 318–326.

Lewin, D. (1987). *Generalized musical intervals and transformations*. New Haven, CT: Yale University Press.

Longuet-Higgins, H. C., & Steedman, M. J. (1971). On interpreting Bach. *Machine Intelligence, 6*, 221–241.

Mauch, M. (2010). *Automatic chord transcription from audio using computational models of musical context* (Ph.D. dissertation, Queen Mary University of London, UK).

Mazzola, G. (2002). *The topos of music, geometric logic of concepts, theory, and performance*. Basel: Birkhäuser.

Nestke, A., & Noll, T. (2001). Inner metric analysis. In J. Haluska (Ed.), *Music and mathematics* (pp. 91–111). Bratislava: Tatra Mountains Mathematical Publications.

Niedermayer, B., & Widmer, G. (2010). A multi-pass algorithm for accurate audio-to-score alignment. In *Proceedings of the international conference on music information retrieval,* Utrecht, Netherlands.

Pachet, F. (2003). The continuator: Musical interaction with style. *Journal of New Music Research, 32*(3), 333–341.

Pachet, F., & Roy, P. (2004). Musical harmonization with constraints: A survey. *Constraints, 6*(1), 7–19.

Palmer, C., & Hutchins, S. (2006). What is musical prosody? In B. H. Ross (Ed.), *Psychology of learning and motivation* (Vol. 46, pp. 245–278). Amsterdam: Elsevier Press.

Raphael, C. (2009). Representation and synthesis of melodic expression. In *Proceedings of the international joint conference on AI*, Pasadena, California, Vol. 21, pp. 1475–1480.

Sethares, W. A. (2007). *Rhythm and transforms*. London: Springer-Verlag.

Shmulevich, I., & Yli-Harja, O. (2000). Localized key finding: Algorithms and applications. *Music Perception, 17*(4), 531–544.

Smith, L. M., & Honing, H. (2008). Time-frequency representation of musical rhythm by continuous wavelets. *Journal of Mathematics and Music, 2*(2), 81–97.

Temperley, D. (2001). *The cognition of basic musical structures*. Cambridge, MA: MIT Press.

Temperley, D. (2007). *Music and probability*. Cambridge, MA: MIT Press.

Timmers, R. (2005). Predicting the similarity between expressive performances of music from measurements of tempo and dynamics. *Journal of the Acoustical Society of America, 117*(1), 391–399.

Truchet, C., & Codognet, P. (2004). Musical constraint satisfaction problems solved with adaptive search. *Soft Computing, 8*, 633–640.

Tymoczko, D. (2006). The geometry of musical chord. *Science, 313*(5783), 72–74.
Typke, R., Giannopoulos, P., Veltkamp, R. C., Wiering, F., & van Oostrum, R. (2003). Using transportation distances for measuring melodic similarity. In *Proceedings of the international symposium on computer music modeling and retrieval*, Montpellier, France, pp. 107–114.
Volk, A. (2008). Persistence and change: Local and global components of Metre induction using inner metric analysis. *Journal of Mathematics and Music, 2*(2), 99–115.
Xenakis, I. (2001). *Formalized music: Thought and mathematics in music*. Hillsdale, NY: Pendragon Press.

Digraph

A graph all of whose edges have a designated one-way direction.

See

- Graph Theory

Dijkstra's Algorithm

A method for finding shortest paths (routes) in a network. The algorithm is a node labeling, greedy algorithm. It assumes that the distance c_{ij} between any pair of nodes i and j is nonnegative. The labels have two components $\{d(i), p\}$, where $d(i)$ is an upper bound on the shortest path length from the source (home) node s to node i, and p is the node preceding node i in the shortest path to node i. The algorithmic steps for finding the shortest paths from s to all other nodes in the network are as follows:

Step 1. Assign a number $d(i)$ to each node i to denote the tentative (upper bound) length of the shortest path from s to i that uses only labeled nodes as intermediate nodes. Initially, set $d(s) = 0$ and $d(i) = \infty$ for all $i \neq s$. Let y denote the last node labeled. Give node s the label $\{0, -)$ and let $y = s$.

Step 2. For each unlabeled node i, redefine $d(i)$ as follows:

$d(i) = \min\{d(i), d(y) + c_{yi})\}$. If $d(i) = \infty$ for all unlabeled vertices i, then stop, as no path exists from s to any unlabeled node i with the smallest value of $d(i)$. Also, in the label, let p denote the node from which the arc that determined the minimum $d(i)$ came from. Let $y = i$.

Step 3. If all nodes have been labeled, stop, as the unique path of labels $\{d(i), p\}$ from s to i is a shortest path from s to i for all vertices i. Otherwise, return to *Step 2*.

See

- Greedy Algorithm
- Minimum-Cost Network-Flow Problem
- Network Optimization
- Vehicle Routing

Directed Graph

- Digraph

Direction of a Set

A vector $\boldsymbol{d}$ is a direction of a convex set if for every point $\boldsymbol{x}$ of the set, the ray $(\boldsymbol{x} + \lambda \boldsymbol{d})$, $\lambda \geq 0$, belongs to the set. If the set is bounded, it has no directions.

See

- Convex Set

Directional Derivative

A rate of change at a given point in a given direction of the value function of a optimization problem as a function of problem parameters.

See

- Nonlinear Programming

Disaster Management: Planning and Logistics

Gina M. Galindo Pacheco[1,2] and Rajan Batta[1]
[1]University at Buffalo, The State University of New York, Buffalo, NY, USA
[2]Universidad del Norte, Barranquilla, Colombia

Introduction

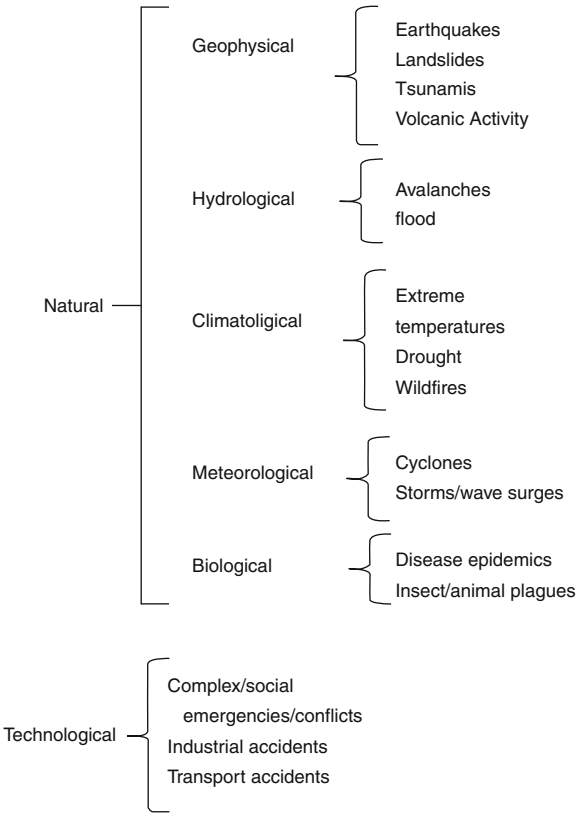
Due to significant losses of life, as well as extremely high economic costs, the prevention and improvement of disaster response has been a continuing area of research. OR analysts have been in the forefront of such work and have made significant contributions that have helped to mitigate the impact of disasters. This article reviews some of the basic concepts related to disaster management (DM) and summarizes many of the topics that have been addressed.

The presentation is as follows: section one reviews disaster definitions and types; section two focuses on the role of DM, the concepts associated, and the stages that are traditionally identified within DM; section three discusses the role of the planning process; section four reviews the related logistics issues; section five discusses DM topics based on a sample of work from the period 2005-2010; and the last section presents a summary and concluding remarks.

Definition of Disaster

According to the International Federation of Red Cross and Red Crescent Societies (IFRC), a disaster is a sudden event that causes disruption of the normal functioning of a community; causes human, economic, and environmental losses; and generates requirements that exceeds the capacity of response using available resources.

Losses due to disasters may be of the order of thousands of lives and billions of dollars. Kunkel, Pielke, and Changnon (1999) give some statistics about human and economic losses due to weather and climate extremes in the U.S. They estimate that between 1986 and 1995 there was an annual mean loss of 96 lives due to floods and 20 due to hurricanes. In the same period, the annual mean of economic losses was $6.2 billion for hurricanes. In 2005, the National Hurricane Center estimated that hurricane Katrina left a total of 1,200 reported casualties, with a total damage cost of $81 billion. Man-made disasters can also have drastic consequences if they are purposely planned. For example, according to the National Commission on Terrorist Attacks upon United States, more than 2,981 people died in the attacks of 9/11. Even though environmental disasters typically do not involve many human casualties, they do cause great ecological damages, e.g., the Gulf of Mexico oil spill that affected thousands of turtles, birds, and mammals, as reported by the International Disaster Database Web site (in addition to the considerable monetary loss for British Petroleum). The types of natural and man-made disasters are listed in Fig. 1.



Disaster Management: Planning and Logistics, Fig. 1 Types of disasters

This classification derives partly from IFRC, Alexander (2002), and Van Wassenhove (2006).

Criteria	Classification
Cause	Natural
	Technological
Onset	Sudden
	Slow
Detection	Predictable
	Unpredictable

Disaster Management: Planning and Logistics, Fig. 2 Disaster classification

Natural disasters may be grouped into predictable ones, such as hurricanes, and unpredictable events, such as earthquakes. Data about predictable disasters are not deterministic, but some information about the time and place of such disasters is available. Such disasters can also be classified with respect to their time of onset. Tornadoes happen suddenly and last for a short period of time, while events such as pandemics may go from a few days to several months. These classifications become important at the time of planning and responding: for predictable disasters actions like evacuation or prepositioning of supplies are possible, while for unpredictable ones, such actions are not possible alternatives; for very short-term disasters it is easier to estimate the amount of resources needed to overcome the situation, where for long-term disasters this is a more difficult task. Figure 2 summarizes these classifications.

Role of Disaster Management

According to the IFRC, the management of resources and responsibilities to respond to humanitarian needs after an emergency is known as Disaster Management (DM).

DM can be viewed as including the strategic, tactical, and operational activities, as well as the personnel and technologies involved at various stages of a disaster situation for the purpose of mitigating its possible consequences (Lettieri et al. 2009).

The different stages involved in DM are classified as mitigation, preparedness, response, and recovery (McLoughlin 1985). Miller, Engemann, and Yager (2006) provide a detailed explanation of the four DM stages. Each of these stages is briefly discussed below with respect to a flood disaster.

Mitigation consists of those activities that help to reduce the long-term risk of the occurrence of a disaster or its consequences. For a flood scenario, mitigation would involve not building on low lands, and creating barriers along rivers or ponds. Preparedness refers to planning operational activities to respond to a disaster—creating shelters, prepositioning supplies, and evacuating people from most dangerous locations is a way in which preparedness may be applied for a flood setting. The response stage includes actions that correspond to those performed upon the occurrence of the disaster to help affected people to overcome their needs of essential resources or getting them out from danger e.g., delivering supplies and rescuing people. The recovery phase involves short and long-term activities to restore normal functioning of the community, as well as repairing roads and buildings.

The recovery phase should be designed in such a way that it contributes to mitigation efforts. For the flood example, rebuilt houses should not be located in lands known to be highly exposed to floods. This is how DM could be viewed as a cycle created by the link of mitigation and recovery activities. In general, the different stages of DM require a previous planning process to coordinate all the ulterior actions that would be performed. In addition, a logistic process is involved mainly, but not exclusively, for the preparedness and response phases.

Disaster Management and Planning Process

The Oxford English Dictionary defines the verb "to plan" as meaning "to devise, contrive, or formulate (something to be done, or some action or proceeding to be carried out.)" For DM, Alexander (2002) distinguishes emergency planning in terms of long and short-term. The former gives the context for the latter. It involves forecasting, warning, educating, and training people for the event of a disaster. It includes the study of patterns to predict the possible time and place at which a disaster could occur. Seasonal natural disasters, such as tropical storms in the Caribbean, are examples. The concept of long-term planning is related

to the definition of emergency planning given by Perry and Lindell (2003) for whom emergency planning focuses on the two objectives of hazard assessment and risk reduction. The purpose of short-term planning is to guarantee the prompt deployment of resources where and when needed.

Alexander (2002) describes an outline of the methodological components of an emergency plan and includes a generic emergency planning model. The planning process may be summarized as gathering information, managing and analyzing it, extracting some conclusions and actions to be developed, and communicating the resulting plan to the staff involved.

Disaster Management and Logistics

Several definitions are used for the term logistics. Van Wassenhove (2006) gives a brief and illustrative review of some of these definitions as applied to business, military, and humanitarian DM logistics. In summary, logistics, when applied to DM, is referred to as the storage and deployment of resources and information, as well as the mobilization of people in an effective way to reduce the impact of the disaster. Kovács and Spens (2007) and Van Wassenhove (2006) reflect upon the comparison between business and humanitarian DM logistics. However, despite the differences, business and humanitarian logistics are intrinsically related and they both refer to a process that includes planning, distribution and transportation, storage, location and supply chain management (SCM).

In what follows, some common problems related to planning and logistics in DM and OR are discussed.

OR and DM

A survey of OR research related to DM since 2005 was conducted. A total of 222 items in journals, books, book chapters, and conference papers were reviewed. A finding was that topics of planning and logistics in DM attracted most of the attention. For planning, the most common topics were evacuation and risk analysis. General humanitarian logistics was a topic addressed in terms of (i) transportation, (ii) inventory, (iii) location analysis, and (iv) humanitarian logistics (in general). Material from (i) to (iii) are referred to as specific activities inside the concept of logistics, while that from (iv) considers logistics as a whole or that combines different aspects of humanitarian logistics. Other topics of logistics are reviewed separately because they constitute a widely studied topic as is the case for transportation that includes research on routing, traffic and network management.

Even though there were many other topics of OR interest in the reviewed research such as demand forecast, business continuity, and hospital capacity, the topics mentioned earlier represent the main streams that were studied. In the following sections, the topics will be discussed separately focusing on the relationship to DM phases, methodologies, objectives, and real-life applications.

Evacuation: The major way for reducing the potential population affected by a disaster is evacuation. An evacuation typically involves mobilizing people from endangered zones to safer ones, which includes routing strategies and preparation of shelters, among other activities. This process is mostly associated with the preparedness phase of DM, and, therefore, to the planning processes. However, some related work for real-time decisions may be linked to the response phase (Chiu and Zheng 2007). For predictable disasters, it is possible to develop evacuation plans to be performed before the disaster strikes; no pre-disaster-evacuation planning is possible for unpredictable disasters.

The most common objective in evacuation research was minimizing the evacuation time of the total affected population (Chen and Zhan 2008). Other objectives included maximizing the total number of evacuees during a given evacuation time (Miller-Hooks and Sorrel 2008), maximizing the minimum probability of reaching an exit for any evacuee (Opasanon and Miller-Hooks 2009), and minimizing total system travel time (Chiu et al. 2007). Some studies considered multiple objectives. In Saadatseresht, Mansourian, and Taleai (2009) the objectives were to minimize travel distance, evacuation time, and overload capacity of safe areas. Stepanov and Smith (2009) provide a critique of performance measures for evacuation that include clearance time, total traveled distance, and blocking probabilities.

Simulation was the most used method to solve evacuation problems. Bonabeau, (2002) and

Chen and Zhan, (2008) used agent-based simulation—the process in which entities termed autonomous agents assess their situations and make decisions according to a set of rules(say something about validation). Other studies developed multi-level models (Liu et al. 2006), queue analysis (Stepanov and Smith 2009), mixed integer linear programming (Sayyady and Eksioglu 2010); others used Cell Transmission Models (Chiu et al. 2007), and genetic algorithms (Miller-Hooks and Sorrel 2008).

Most of the studies employed real data to validate their results. For example, Chen, Meaker and Zhan (2006) developed a simulation model for evacuating the Florida Keys under a hurricane setting. They considered two questions: one related to the time for evacuating the total population, while the other considered how many residents would need to be accommodated if evacuation routes were impassable. The authors used a previous study as a reference for comparing the results of their model. However, no validation based on real evacuation times is reported.

Risk Analysis: DM risk analysis is mainly concerned with quantifying the risk of the occurrence of an undesirable event, as well as developing measures to diminish the impact of a disaster. Risk analysis is mainly a planning tool related to the mitigation. The objectives of the DM risk analysis studies were forecasting, infrastructure planning and design, vulnerability, and analysis of uncertainty, as discussed next.

In relation to forecasting, Hu (2010) uses a Bayesian approach to analyze flood frequencies. Infrastructure planning and design based on risk analysis refers in some cases to making the infrastructure (buildings, networks, supply chains, etc.) more resistant to disaster damages and disruptions, and to building physical barriers or diversions to diminish the impact of a disaster on an endangered community. Snyder et al. (2006) reviewed several models for designing supply chains resilient to disruptions. These models considered costs from the business point of view, with objectives, in most of the cases, being the minimization of the expected or the worst case cost. Li, Huang and Nie (2007) used a model for flood diversion planning under uncertainty where, among the objectives considered, was the minimization of risk of system disruption. Vulnerability relates to the way in which current systems are affected by damages. Matisziw and Murray (2009) maximized system flow for a disrupted network. Barker and Haimes (2009) focused on a sensitivity analysis of extreme consequences due to uncertainties on the parameters, and Xu, Booij and Tong (2010) analyzed the sources of uncertainty in statistical modeling.

Probability and statistics were the main methods used to analyze risk analysis. In the case of Li, Huang, and Nie (2007) the authors used a methodology that combines fuzzy sets and stochastic programming. Another example in which fuzzy sets have been incorporated into risk analysis is given by Huang and Ruan (2008). In this DM area, even though some researchers used real data to develop numerical examples, complete case studies were rare.

Transportation: Transportation problems typically deal with routing, vehicle schedule, traffic, and network management. The problems may be to transport goods to provide relief supplies, evacuate people from endangered areas, or movement of resources such as medical staff to areas where their services are required.

For transportation analyses, as applied to DM, there are a wide variety of objectives related to the efficiency of delivery times. Campbell, Vandenbussche, and Hermann (2008) considered two objectives for minimizing the arrival times of relief to demand points. Similarly, Yuan and Wang (2008) minimized the total travel time through a path selection methodology, while Jin and Ekşioğlu (2008) minimized vehicle delay.

Methods used included mathematical programming and its derivates, such as stochastic and integer programming, Campbell, Vandenbussche, and Hermann (2008) and Yuan and Wang (2009). Jotshi, Gong and Batta (2009) used the HAZUS program to develop a post-earthquake scenario in Los Angeles. [HAZUS is a computer-based system created and distributed via the Web by the Federal Emergency Management Agency (FEMA) for estimating potential losses caused by earthquakes, floods and hurricanes].

Inventory: Traditionally, in the commercial area, inventory analyses address a number of areas: materials, components, work-in-process, and finished goods (Nahmias 2009). But, businesses may use inventory theory to pre-analyze forecasted disasters, e.g., Taskin and Lodree (2011) developed an inventory

model for a manufacturing facility whose demand could be impacted by a potential storm. This might also be appropriate for DM in the case of items such as canned food, lamps, and coolers. In general, humanitarian logistics inventory concerns are mostly related to the prepositioning or early acquisition of relief goods. Decisions related to inventory problems fit better in the preparedness phase of DM, but they may affect directly the effectiveness of the response phase if a shortage of inventory occurs.

Most of the inventory-oriented papers shared one common objective: minimize expected cost. This cost may be expressed as a loss function (Taskin and Lodree 2011) or may be a composition of traditional inventory costs including the cost per order, holding inventory cost, and back-order cost (Beamon and Kotebla 2006). Salmerón and Apte (2010) developed a two-stage model for a humanitarian logistics for optimally allocating a budget for acquiring and positioning relief assets. Two objectives were pursued: minimization of the expected number of casualties, and minimization of the expected amount of unmet transfer population. Here, casualties were the result of seriously injured people who were not served promptly by medical staff, and people needing relief supplies who do not get them on time. On the other hand, transfer population represent people who are not in a critical condition, but still need to be evacuated to relief centers. Unmet transfer population applies when these people are not promptly evacuated.

DM inventory problems were analyzed using stochastic optimization combined with statistical tools such as Bayesian methods. Taskin and Lodree, (2011) present some numerical examples with simulated data, while other research used hypothetical data from previous studies.

Location: In general, location analysis deals with problems of siting facilities in a given area (ReVelle and Eiselt 2005). Such problems are commonly classified by businesses as strategic, i.e., a type of decision whose effects are expected to last for a long period due to the fixed cost of opening a facility, and/or changing the location of a facility may be a very expensive. In humanitarian logistics, however, location analysis may be best defined as a tactical decision, as most often it considers locating temporary shelters and warehouses where relief assets may be kept safe. These facilities generally consist of existing sites suitable, such as schools, stadiums, or churches.

Depending on the objectives pursued, results from location analysis may set the framework for ulterior decision problems such as: where to store prepositioned supplies; given the location of such relief supplies, how they would be distributed; where the evacuees will be directed to; and where to locate emergency vehicles or provisional health centers. Location analysis may be more accurately relate to the preparedness phase of DM. But, it could also be associated to the mitigation phase for locating facilities in low-risk areas, or, based on the disaster, in the response phase to improvise additional shelters or medical centers other than those that were planned.

Facility location applied in the preparedness phase is discussed by Balcik and Beamon (2008) who sought to locate distribution centers and determine the amount of supply to preposition at such centers to maximize the total expected demand covered. Lee et al. (2009) studied multiple dispensing points to service a large population searching for prophylaxis, with the objective to minimize the maximum expected traveled distance.

For the mitigation phase, Berman et al. (2009) analyzed where to locate p facilities to maximize coverage on a network whose links could be destroyed. Beraldi and Bruni (2009) studied the location of emergency vehicles under congested settings with the objective of minimizing cost.

Most of the DM location analysis research used mixed integer programming (MIP) and, in some cases, applied heuristic methods to help determine the solution of large problems (Berman et al. 2009). Other studies used stochastic programming models (Beraldi and Bruni 2009), or simulation to generate potential scenarios so as to compare the model results to actual data form a case study (Afshartous et al. 2009).

Logistics Models Overview

DM logistics involves several activities that include planning, warehousing, location, and distribution, among other elements. Some studies combined one or more of these activities, with others focused on an integrated and general concept of logistics.

Kovács and Spents (2007) and Van Wassenhoven (2006) describe humanitarian logistics as a whole. They sought a better understanding of planning and carrying out of logistics in disaster relief through a literature review. Van Wassenhoven presents a parallel between private and humanitarian logistics, and also proposes some guidelines for developing a better preparedness strategy for the latter.

Yi and Özdamar (2007) define an integrated capacitated location-routing model. Their model was designed to coordinate the distribution of relief material and the transportation of evacuees to emergency units selected through location analysis. The objective was to minimize the relationship between the weighted sum of unsatisfied demand and the weighted sum of wounded people at temporary and permanent emergency units using a two stage MIP model.

Chang, Tseng and Chen (2007) analyze a combination of location and transportation: the coordination activities related to rescue logistics efforts under a flood setting in an urban area. They consider the location of rescue resource inventory, allocation and distribution of rescue resources, and the structure of rescue organizations. Using two models, they first classified the rescue areas according to levels of emergency with the objective of minimizing the shipping cost of rescue equipments; the second model was a two stage stochastic-programming model that minimized set-up cost of storehouses and rescue equipment costs.

Yan and Shih (2009) developed a model for roadway repair scheduling and subsequent distribution of relief supplies. The objective was the minimizing the total expected time for repair and distribution using a MIP model. A related study in which a distribution system is modeled as a supply chain where the echelons are the relief suppliers, relief distribution centers, and relief demanding areas is described in Sheu (2007). Here, the objective was to minimize the expected cost of relief distribution during the three days following the onset of the disaster using a hybrid fuzzy-clustering method.

Balcik, Beamon and Smilowitz (2008) studied what is termed the last mile relief distribution, i.e., the distribution of relief assets from distribution centers to final demand. Their model dealt with the allocation of relief supplies to local distribution centers, and the delivery of schedules and routes for distributing vehicles. Their MIP model minimized the expected cost of distribution that included routing costs and a penalty for unmet demand.

Concluding Remarks

This article presented an overview of DM focused on planning and logistics. It is clear that planning and logistics are inseparable, intrinsically related, and both present in different phases of DM. These phases should be performed in a cyclic fashion so that the recovery efforts should also pursue mitigation objectives. Related research showed that many OR/MS-based studies have been directed at improving the effectiveness and efficiency of DM. The impetus for this is probably due to the catastrophic events of the Twin Towers attack in 2001, the 2004 tsunami in the Indian Ocean, and hurricane Katrina in 2005. These events have contributed to generating an increasing concern of reducing both the risk of such disasters happening and diminishing their consequences. A comparison between humanitarian and business logistics highlighted both their differences as well as their commonalities.

The main topics found from the review of OR/MS research, as related to DM, appear to be evacuation, risk analysis, and logistics. The following remarks with respect to these main topics are based on a review of a fraction of the available literature in this area; it is felt, however, that they do represent an accurate view of the state of the art in this growing field, circa 2011.

In general, the evacuation problems showed that the main concern was the minimization of evacuation time. Some researchers stated that one of the important limitations of such studies was predicting the behavior of evacuees—many variables would have to be considered, as well as social context of the evacuated population. Peacock, Morrow, and Gladwin (1997) analyzed how some people may not respond to evacuation measures before a disaster strikes as a function of their ethnic origin or their socio-economic level. The authors' main conclusion dealt with the perception the evacuee population may have about authorities who may stop them from following pre-disaster evacuation orders.

Risk analysis has proved to be a useful concept when planning for disasters, especially during the

mitigation phase. A problem is the difficulty of enumerating the possible risk scenarios. Moreover, many studies are based on statistical analyses to historical data, but in some occasions, the events being studied are so infrequent that no reliable analysis can be achieved.

For humanitarian logistics research, a distinction was made between transportation, location analysis, inventory, and humanitarian logistics, in general. A limitation that may arise in a transportation study is the inability to incorporate the presence of congestion, even though some studies do, see for example Beraldi and Bruni (2009). Inventory theory has been used by both business and humanitarian logistics to better prepare for disasters, including, as well, location analysis problems from business being applied in humanitarian location settings.

The research papers reviewed referred mainly to the preparedness phase of DM, followed by response and mitigation phases; no work was found related to the recovery phase. Altay and Green, (2006) noted the lack of OR studies related to recovery efforts in comparison to the other phases. Another aspect in which the findings obtained here agree with the ones presented by Altay and Green (2006) is that most of the studies reviewed consists of the development of models, rather than theoretical studies or application tools such as software. For the disasters most commonly studied, there was not a clear reference to man-made disasters such as terrorist attacks; the case studies always dealt with natural disasters.

For DM, an important challenge for the OR/MS community "is to develop a science of disaster logistics that builds upon, among others, private sector logistics and to transfer to private business the specific core capabilities of humanitarian logistics," (Van Wassenhove 2006).

See

- ▶ Inventory Modeling
- ▶ Linear Programming
- ▶ Logistics and Supply Chain Management
- ▶ Risk Assessment
- ▶ Scheduling and Sequencing
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Vehicle Routing

References

Afshartous, D., Guan Y., & Mehrotra, A. (2009). US Coast Guard air station location with respect to distress calls: A spatial statistics and optimization based methodology, *European Journal of Operational Research, 196*(3), 1086–1096.

Alexander, D. E. (2002). *Principles of disaster planning and management*. Oxford University Press.

Altay, N., & Green, W. G. (2006). OR/MS research in disaster operations management. *European Journal of Operational Research, 175*, 475–493.

Balcik, B., & Beamon, B. M. (2008). Facility location in humanitarian relief, *International Journal of Logistics Research and Applications: A Leading Journal of Supply Chain Management, 11*(2), 101–121.

Balcik, B., Beamon, B. M., & Smilowitz, K. (2008). Last mile distribution in humanitarian relief. *Journal of Intelligent Transportation Systems, 12*(2), 51–63.

Barker, K., & Haimes, Y. Y. (2009). Assessing uncertainty in extreme events: Applications to risk-based decision making in interdependent infrastructure sectors. *Reliability Engineering and System Safety, 94*, 819–829.

Beamon, B. M., & Kotebla, S. A. (2006). Inventory model for complex emergencies in humanitarian relief operations. *International Journal of Logistics: Research and Applications, 9*(1), 1–18.

Beraldi, P., & Bruni, M. E. (2009) A probabilistic model applied to emergency service vehicle location. *European Journal of Operational Research, 196*(1), 323–331.

Berman, O., Drezner, T., Drezner, Z., & Wesolowsky, G. O. (2009). A defensive maximal covering problem on a network. *International Transactions in Operational Research, 16*(1), 69–86.

Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences, 99*(3), 7280–7287.

Campbell, A. M., Vandenbussche, D., & Hermann, W. (2008). Routing for relief efforts. *Transportation Science, 42*(2), 127–145.

Chang, M.-S., Tseng, Y.-L., & Chen, J.-W. (2007). A scenario planning approach for the flood emergency logistics preparation problem under uncertainty. *Transportation Research Part E: Logistics and Transportation Review, 43*(6), 737–754.

Chen, X., Meaker, J. W., & Zhan, F. B. (2006). Agent-based modeling and analysis of hurricane evacuation procedures for the Florida keys. *Natural Hazards, 38*(3), 321–338.

Chen, X., & Zhan, F. B. (2008). Agent-based modelling and simulation of urban evacuation: Relative effectiveness of simultaneous and staged evacuation strategies. *Journal of the Operational Research Society, 59*, 25–33.

Chiu, Y.-C., & Zheng, H. (2007). Real-time mobilization decisions for multi-priority emergency response resources and evacuation groups: Model formulation and solution. *Transportation Research Part E: Logistics and Transportation Review, 43*(6), 710–736.

Chiu, Y.-C., Zheng, H., Villalobos, J., & Gautam, B. (2007). Modeling no-notice mass evacuation using a dynamic traffic flow optimization model. *IIE Transactions, 39*(1), 83–94.

Hu, Z.-H. (2010). Relief demand forecasting in emergency logistics based on tolerance model. *2010 Third International Joint Conference on Computer Science and Optimization (CSO 2010)*, Vol. 1, pp. 451–455.

Huang, C., & Ruan, D. (2008). Fuzzy risks and an updating algorithm with new observations. *Risk Analysis, 28*(3), 681–694.

Jin, M., & Ekşioğlu, B. (2008). Optimal routing of vehicles with communication capabilities in disasters. *Computational Management Science, 7*(2), 121–137.

Jotshi, A., Gong, Q., & Batta, R. (2009). Dispatching and routing of emergency vehicles in disaster mitigation using data fusion. *Socio-Economic Planning Sciences, 43*(1), 1–24.

Kovács, G., & Spents, K. M. (2007). Humanitarian logistics in disaster relief operations. *International Journal of Physical Distribution and Logistics Management, 37*(2), 99–114.

Kunkel, K. E., Pielke, Jr., R. A., & Changnon, S. A. (1999). Temporal fluctuations in weather and climate extremes that cause economic and human health impacts: A review. *Bulletin of the American Meteorological Society, 80*, 1077–1098.

Lee, E. K., Smalley, H. K., Zhang, Y., Pietz, F., & Benecke, B. (2009). Facility location and multi-modality mass dispensing strategies and emergency response for biodefence and infectious disease outbreaks. *International Journal on Risk Assessment and Management, 12*(2–4), 311–351.

Lettieri, E., Masella, C., & Radaelli, G. (2009). Disaster management: Findings from a systematic review. *Disaster Prevention and Management, 18*(2), 117–136.

Li, Y. P., Huang, G. H., & Nie, S. L. (2007). Mixed interval-fuzzy two-stage integer programming and its application to flood-diversion planning. *Engineering Optimization, 39*(2), 163–183.

Liu., Y., Lai., X., & Chang., G. (2006). Two-level integrated optimization system for planning of emergency evacuation. *Journal of Transportation Engineering*, 800–807.

Matisziw, T. C., & Murray, A. T. (2009). Modeling s–t path availability to support disaster vulnerability assessment of network infrastructure. *Computers and Operations Research, 36*(1), 16–26.

McLoughlin, D. (1985). A framework for integrated emergency management. *Public Administration Review, 45*, 165–172.

Miller, H. E., Engemann, K. J., & Yager, R. R. (2006). Disaster planning and management. *Communications of the IIMA, 6*(2), 25–36.

Miller-Hooks, E., & Sorrel, G. (2008). Maximal dynamic expected flows problem for emergency evacuation planning. *Transportation Research Record: Journal of the Transportation Research Board, 2089*, 26–34.

Nahmias, S. (2009). *Production and operation analysis* (6th ed., pp. 201–202). McGraw-Hill.

Opasanon, S., & Miller-Hooks, E. (2009). The safest escape problem. *Journal of the Operational Research Society, 60*, 1749–1758.

Peacock, W. G., Morrow, B. H., & Gladwin, H. (1997). *Hurricane Andrew: Ethnicity, gender and the sociology of disasters* (p. 278). New York: Routledge.

Perry, R. W., & Lindell, M. K. (2003). Preparedness for emergency response: Guidelines for the emergency planning process. *Disasters, 27*(4), 336–350.

ReVelle, C. S., & Eiselt, H. A. (2005). Location analysis: A synthesis and survey. *European Journal of Operational Research, 165*(1), 1–19.

Saadatseresht, M., Mansourian, A., & Taleai, M. (2009). Evacuation planning using multiobjective evolutionary optimization approach. *European Journal of Operational Research, 198*(1), 305–314.

Salmerón, J., & Apte, A. (2010). Stochastic optimization for natural disaster asset preposition. *Production and Operations Management, 19*(5), 561–574.

Sayyady, F., & Eksioglu, S. D. (2010). Optimizing the use of public transit system during no-notice evacuation of urban areas. *Computers and Industrial Engineering, 59*(4), 488–495.

Sheu, J.-B. (2007). An emergency logistics distribution approach for quick response to urgent relief demand in disasters. *Transportation Research Part E: Logistics and Transportation Review, 43*(6), 687–709.

Snyder, L. V., Scaparra, M. P., Daskin, M. S., & Church, R. L. (2006). *Planning for disruptions in supply chain networks*. Tutorials in Operations Research INFORMS.

Stepanov, A., & Smith, J. M. (2009). Multi-objective evacuation routing in transportation networks. *European Journal of Operational Research, 198*, 435–446.

Taskin, S., & Lodree, E. J. (2011). A Bayesian decision model with hurricane forecast updates for emergency supplies inventory management. *Journal of the Operational Research Society, 62*, 1098–1108. Published online 19 May 2010.

Van Wassenhove, L. N. (2006). Blackett memorial lecture – humanitarian aid logistics: Supply chain management in high gear. *Journal of the Operational Research Society, 57*(5), 475–489.

Xu, Y.-P., Booij, M. J., & Tong, Y.-B. (2010). Uncertainty analysis in statistical modeling of extreme hydrological events. *Stochastic Environmental Research and Risk Assessment, 24*(5), 567–578.

Yan, S., & Shih, Y.-L. (2009). Optimal scheduling of emergency roadway repair and subsequent relief distribution. *Computers and Operations Research, 36*(6), 2049–2065.

Yi, W., & Özdamar, L. (2007). A dynamic logistics coordination model for evacuation and support in disaster response activities. *European Journal of Operational Research, 179*(3), 1177–1193.

Yuan, Y., & Wang, D. (2009). Path selection model and algorithm for emergency logistics management. *Computers and Industrial Engineering, 56*(3), 1081–1094.

Discrete-Programming Problem

► Integer and Combinatorial Optimization

Discrete-Time Markov Chain (DTMC)

A discrete-time, countable-state Markov process. It is often just called a Markov chain.

See

- ▶ Markov Chains
- ▶ Markov Processes

Disease Prevention, Detection, and Treatment

Jingyu Zhang[1], Jennifer E. Mason[2], Brian T. Denton[3] and William P. Pierskalla[4]
[1]Philips Research North America, Briarcliff Manor, NY, USA
[2]University of Virginia, Charlottesville, VA, USA
[3]University of Michigan, Ann Arbor, MI, USA
[4]University of California, Los Angeles, CA, USA

Introduction

Advances in medical treatment have resulted in a patient population that is more complex, often with multiple diseases, competing risks of complications, and medication conflicts, rendering medical decisions harder because what helps one patient or condition may harm another. The use of Operations Research (OR) methods for the study of healthcare has a long history. Furthermore, there is a growing literature on emerging applications in this area. This article provides examples of contributions of OR methods, including mathematical programming, dynamic programming, and simulation, to the prevention, detection, and treatment of diseases. More extensive surveys of OR studies of health care delivery, including medical decision making, can be found in Pierskalla and Brailer (1994), Brandeau et al. (2004), and Rais and Viana (2010).

Advances in medical treatment have extended the average lifespan of individuals, and transformed many diseases from life threatening in the near term to chronic conditions in need of longterm management. Many new applications of OR are emerging as treatment options and population health evolve over time. For example, new treatments have become available for various forms of cancer, HIV, and heart disease. In some cases, patients are living decades with diseases that previously had low short-term survival rates. As a result, more patients are living with co-morbid conditions, and competing risks, creating challenging decisions that must balance the downside of treatment (e.g., medication side effects and long-term complications) with the benefits of treatment (e.g., longer life expectancy and better quality of life).

Diabetes is a good example of a chronic disease for which medical treatment is complex. With nearly 8% of the U.S. population estimated to have diabetes, it is recognized as a leading cause of mortality and morbidity. It is associated with long-term complications that affect almost every part of the body, including coronary heart disease (CHD), stroke, blindness, kidney failure, and neurological disorders. For many patients, diabetes might be prevented through improved diet and exercise. However, due to the slow development of symptoms in many patients, diabetes can go undetected for years. For patients that are diagnosed with diabetes, risk models exist to predict the probability of complications, but alone these models do not provide optimal treatment decisions. Rather, they provide raw data that can be used in OR models to make optimal treatment decisions. This general situation is true of many chronic diseases. As a result, there are many emerging opportunities for applications of OR to disease prevention, detection, and management.

This article is organized as follows. The section on Disease Prevention and Screening describes important contributions of OR to disease prevention, including vaccination and screening methods for detecting disease in a population of potentially infected people. The section on Treatment Choices focuses on applications to long-term management of chronic diseases, including selection among multiple treatment choices, and decisions about timing and dosage of treatment. The section on Emerging Applications reviews some emerging applications to real-time decision making at the point of care and patient decision aids. Finally, research opportunities are discussed in the Conclusions section.

Disease Prevention and Screening

Prevention and screening are important factors in determining overall population health. OR has been applied to help inform decisions related to prevention and screening for decades. Two major topics in this area, that are prominent in the OR literature, are vaccination and disease screening. Vaccination emphasizes the prevention of infectious diseases, while disease screening is common for both non-infectious and infectious diseases. Each of these topics will be discussed in detail in this section.

Vaccination

The biological and genetic sciences have greatly increased the knowledge of how viruses and bacteria operate within the body to create disease. This has led to the discovery of many new vaccines. However, the myriad interactions as well as controversy about their effects on individuals, and an overall population, have drawn considerable public attention. These interactions and effects present several challenges in the utilization of the vaccines for disease control. First, there are a large number of diseases for which effective vaccines are available. Some have specific requirements, such as multiple doses that must be administered within a minimum or maximum time window. Also, some have conflicts with other vaccines. Second, many new vaccines are coming on the market, including combination (multi-valent) vaccines that can cover multiple diseases. Third, for some diseases there is uncertainty about the future evolution of epidemic strains, leading to questions about optimal design of vaccines. Finally, there are challenges in the vaccine manufacturing process including uncertain yields, quality control, supply chain logistics, and the optimal storage location of vaccine supplies. OR models have been applied to address many of these challenges.

Pediatric Vaccination

Pediatric or childhood vaccination is the most common means of mass vaccination. OR researchers have developed models to aid in the selection of a vaccine formulary, pricing of vaccines, and design of vaccination schedules. Jacobson et al. (1999) proposed integer-programming models to determine the price of combination vaccines for childhood immunization. Their models considered all available vaccine products at their market prices and constraints based on the U.S. national recommended childhood immunization schedule. Their objective was to find the vaccine formularies with the lowest overall cost from the patient, provider, and societal perspectives. Their integer-programming models considered the first five years of the recommended childhood immunization schedule against six diseases. They used binary decision variables to denote whether a vaccine is scheduled for a particular month's visit.

In a later study, Jacobson et al. (2006) investigated a pediatric vaccine supply shortage problem to assess the impact of pediatric vaccine stockpile levels on vaccination coverage rates of the guidelines during supply interruption. Their model was similar to inventory models that consider stock-outs, as well as lot sizing problems with machine breakdowns. Objectives of their model included optimizing service level and minimizing a standard loss function. Using their model, they concluded that the guidelines are only sufficient to mitigate a vaccine production interruption of eight months.

Hall et al. (2008) considered a childhood vaccination formulary problem that allows for combination vaccines. They proposed an integer-programming model to minimize the cost of fully immunizing a child under the constraints of a recommended schedule. They proved their proposed model is NP-hard. They proposed exact algorithms using dynamic programming and heuristics for approximating near optimal solutions to their model. Engineer et al. (2009) further investigated an extension that involves catch-up scheduling for childhood vaccination. They provided details of a successful implementation of their model as a decision support system.

Flu Vaccination

Some diseases evolve rapidly over time, necessitating frequent vaccination on a regular basis. For example, the composition of seasonal flu vaccine changes every year. Wu et al. (2005) proposed a model for flu vaccine design. They used a continuous-state discrete-time dynamic-programming model to find the optimal vaccine-strain selection policy. In their dynamic program, the state was represented by the antigenic history, including previous vaccine and epidemic strains. The decision variable (action) was the vaccine strain to be selected, and the reward is the

cross-reactivity representing the efficacy of the vaccine. The objective was to maximize the expected discounted reward. Approximate solutions were obtained by state-space aggregation and compared to an easy to-implement myopic policy based on approximating the multi-stage problem by a series of single period problems. They compare policies suggested by their model to theWorld Health Organization (WHO) recommended policy. Based on their results, the authors suggested that the WHO policy is reasonably effective and should be continued.

Vaccination for Bio-defense

OR researchers have contributed to problems related to vaccination strategy for bio-defense. For instance, Kaplan et al. (2003) analyzed bio-terror response logistics using smallpox as an example. The authors proposed a trace vaccination model using a system of ordinary differential equations (ODEs) incorporating scarce vaccination resources and queueing of people for vaccination. An approximate analysis of the ODEs yields closed-form estimates of numbers of deaths and maximum queue length. They also obtained approximate closed-form expressions for the total number of deaths under mass vaccination. Using these results, approximate thresholds for controlling an epidemic were derived.

Kress (2006) also considered the problem of optimizing vaccination strategy in response to potential bio-terror events. The author developed a flexible, large-scale analytic model with discrete-time decisions. The author used a set of difference equations to describe the transition of the number of people at each epidemic stage and proposed a vaccination policy, which is a mixture of mass and trace vaccination policies.

Other Vaccination Related Problems

Several other vaccine-related problems have been investigated by OR researchers. For example, vaccine allocation problems must consider criteria and constraints related to vaccine manufacturing and supply chain logistics. Becker and Starczak (1997) formulated the optimal allocation of vaccine as a linear-programming problem. Their objective was to prevent epidemics with the minimum required vaccine coverage. Their linear-programming model considered heterogeneity among individuals and minimized the initial reproduction number for a given vaccination coverage. The optimal vaccine allocation strategy suggested more individuals need to be vaccinated in larger households.

Disease Screening

Disease screening is important in extending life expectancy and improving people's quality of life. Effective screening can also reduce costs to the healthcare system by avoiding the high costs associated with treatment of late-stage disease. However, when and how to screen for a specific disease is a complex decision. For instance, model formulation is often difficult due to unclear pathology and risk factors, uncertainty in disease staging and the relationship to symptoms and test results, and the trade-off between the benefit of early detection and the side effects and costs of screening and treatment. The types of OR methods employed depend on whether the disease is non-infectious or infectious. Following are several examples from each category of diseases.

Non-infectious Disease Screening

Modeling disease progression among different stages throughout a patient's lifetime, as well as the trade-off between pros (e.g., longer life expectancy and better quality of life) and cons (e.g., side effects and costs of over-diagnosis and over-treatment) of disease screening are central to non-infectious diseases. Shwartz (1978) proposed one of the first models for breast cancer screening to evaluate and compare alternative screening strategies. Their stochastic model consisted of a discrete set of breast cancer disease states and criteria including life expectancy and the probability of diagnosis. A significant amount of research on breast cancer screening has developed; see Mandelblatt et al. (2009) for a review of breast cancer screening models.

Eddy (1983) presented a general model of monitoring patients with repeated and imperfect medical tests. The model considered clinical and economic outcomes such as the probability of detecting a disease, the method and timing of detection, the stage at which the disease is detected, costs, and the benefit of screening based on the

willingness to pay. The model incorporated disease incidence, the natural history of disease progression, the effectiveness of tests and subsequent treatments, and the order and frequency of tests. The model was illustrated using a hypothetical example. The model had subsequently been applied in clinical practice to several cancer screening problems.

To capture uncertainty in identifying disease states, OR techniques such as partially observable Markov decision process (POMDP) have been applied. For example, Zhang et al. (2012) developed a POMDP model for prostate cancer screening. Due to the slow growing nature of prostate cancer, the imperfect nature of diagnostic tests, and the quality of life impact of treatment, whether and when to refer a patient for biopsy is controversial. The objective of their model was to maximize the quality adjusted life expectancy and minimize the costs of screening and treatments. They assumed that cancer states are not directly observable, but the probability a patient has cancer can be estimated from their PSA test history. A control-limit type policy of biopsy referral and the existence of stopping time of prostate cancer screening were proven. The authors compared policies suggested by their model, to commonly recommended screening policies, and concluded there may be substantial benefits from using prostate cancer risk to make screening decisions.

Screening for disease is greatly influenced by the diagnostic accuracy of the tests. An example of work done in this area is given by Rubin et al. (2004) in which the authors used a Bayesian network to assist mammography interpretation. Interpreting mammographic images and making correct diagnoses are challenging even to experienced radiologists. False-negative interpretations can cause delay in cancer treatment and lead to higher morbidity and mortality. False positives, on the other hand, result in unnecessary biopsy causing anxiety and increased medical costs. The American College of Radiology developed BI-RADS which is a lexicon of mammogram findings and the distinctions that describe them. The authors showed that their Bayesian network model may help to reduce variability and improve overall interpretive performance in mammography.

Many other diagnostic areas have been addressed including gastrointestinal diseases, neurological diseases, and others.

Infectious Disease Screening

In infectious diseases screening, one of the goals is to prevent an epidemic outbreak. Therefore, disease progression and communication throughout a population is an important consideration. Lee and Pierskalla (1988) proposed a mathematical-programming model for contagious diseases with little or no latent periods. The objective of their model was to minimize the average number of infected people in the population. Their model was converted to a knapsack problem. They considered both perfect and imperfect reliability of tests and showed the optimal screening policy has equally spaced screening intervals when the tests have perfect reliability.

Disease screening problems often involve multiple criteria, stemming from the patient, provider, and societal perspectives. For example, Brandeau et al. (1993) provided a cost-benefit analysis of HIV screening for women of childbearing age based on a dynamic compartmental model incorporating disease transmission and progression over time. The model was formulated as a set of simultaneous nonlinear differential equations. The authors found the primary benefit of screening is to prevent the infection of their adult contacts, and that screening of the medium to high risk groups may be cost-beneficial, but it is not likely to be cost-beneficial for low risk women.

Blood screening tests have been used to improve the quality of the blood supply. An early example to improve the performance of testing strategies in the 1980s was provided by Schwartz et al. (1990) for screening blood for the HIV antibody, and making decisions affecting blood donor acceptance. At the time the work was done, limited knowledge was available about the biology, epidemiology, and early blood manifestations of HIV. Furthermore, the initial and conditional sensitivities and specificities of enzyme immunoassays and Western blot tests had wide ranges of errors. A decision tree, with the decisions probabilistically based on which screening test to use, and in what sequence, was used to minimize the number of HIV infected units of blood and blood products entering the nation's blood supply subject to a budget constraint. The model was used at a meeting of an expert panel of the U.S. National Heart Lung and Blood Institute to inform the panelists who were deciding which blood screening protocol to

recommend. The model provided outputs including: expected number of infected units entering the blood supply per unit time, expected number of uninfected units discarded per unit time, expected number of uninfected donors falsely notified, and the incremental cost among screening regimens.

Efficiency of screening can be a defining factor in the success or failure of proposed screening methods. Wein and Zenios (1996) proposed models for pooled testing of blood products for HIV screening. Optimization of pooled testing involves decisions such as transfusion, discarding of samples in the pool, and division of the pool into sub-pools. Several models were proposed to minimize the expected costs. The outcome of an HIV test was measured by an optical density (OD) reading, a continuous measurement which is determined by the concentration of the antibodies. The states of the system were the previous history of the OD readings. A dynamic-programming model with a discretized state space and a heuristic solution algorithm were introduced to obtain near optimal solutions. The policy obtained by the heuristic algorithm was proposed as a cost-effective, accurate, and relatively simple alternative to the implemented HIV screening policies.

Treatment Choices

The following section focuses on treatment decisions for patients with chronic diseases such as diabetes, HIV, cancer, and end-stage renal disease. Treatment of patients with chronic diseases is often complex due to the long-term nature of the illness and the future uncertainty in patient health. Complicating matters, these patients may have other comorbidities that need to be taken into account when treatment decisions are made. In the following section, two areas related to choice of treatment are presented where OR is used to address challenges related to drug treatment decisions and organ transplantation for patients with chronic conditions.

Drug Treatment Decisions

Many diseases involve complex drug treatment decisions, particularly for chronic conditions. Decisions about which medications to initiate, when to initiate treatment, and the appropriate dosage are of primary importance. Additional challenges arise from the fact that there is uncertainty about the future health of the patient, adherence to treatment, and the efficacy of drugs for a particular patient. Treatment decisions must also take into account the often irreversible nature of treatment decisions. Many treatment optimization models employ the use of a natural history model of the disease and all-cause mortality, incorporating the influence of competing risks into the treatment decision.

Choice of Treatment

When there are multiple candidate treatments available, the choice of treatment may be unclear. OR techniques have been used to select treatments. For example, Pignone et al. (2006) presented a Markov model to select among aspirin, statins, and combination treatment, for the prevention of coronary heart disease (CHD). The model simulated the progression of middle-aged males with no history of CHD. The model was used to estimate cost per quality-adjusted life year (QALY) gained. The authors found that aspirin dominates no treatment when a patient's ten-year risk of CHD is at least 7.5%. If a patient's risk is greater than 10%, combination treatment is recommended.

Hazen (2004) used dynamic influence diagrams to analyze a chain of decisions as to whether a patient should proceed to total hip replacement surgery or not. The objective in making this decision was to calculate the optimal expected costs and QALYs under each choice. The use of QALYs for the objective was important because an older person undergoing hip replacement may not have more expected years of life relative to not doing surgery, but the quality of life improvement can be considerable and, quite possibly, worth the cost.

Timing of Treatment

With chronic conditions that can span many years, the optimal time to initiate particular treatments may be unknown. There have been several studies that researched the optimal timing of treatment. Two models relate to the optimal timing of HIV treatment. This question is of particular interest since patients that begin HIV treatment will only be able to use the drug for a limited amount of time, as the virus builds up resistance to the drug. Shechter et al. (2008) used a Markov decision process (MDP) model to find the optimal time to initiate HIV therapy, while

maximizing the patient's quality of life. At monthly decision epochs, the decision was made to initiate therapy or wait until the next month to decide. The health states were based on the number of CD4 white blood cells, the primary target of HIV, and the reward was the expected remaining lifetime in months. They assumed a stationary infinite horizon model and found that if it is optimal to initiate treatment at a given CD4 count, it is also optimal to initiate treatment for patients with higher CD4 counts. The model supported earlier treatment, despite trends toward later treatment. Braithwaite et al. (2008) analyzed the timing of initiation based on CD4 counts for varying viral loads. They used a simulation to compare different CD4 count treatment thresholds for initiation of therapy. The model compared life expectancy and QALYs for the different strategies of initiation. In agreement with Shechter et al.'s finding, the simulation suggested that the use of earlier initiation of treatment (higher CD4 count thresholds) results in greater life years and QALYs.

Agur et al. (2006) developed a method to create treatment schedules for chemotherapy patients using local search heuristics. The model simulated cell growth over time and finds two categories of drug protocols: one-time intensive treatment and a series of nonintensive treatments. Chemotherapy schedules were evaluated based on a patient's state at the end of a given time period, number of cancer and host cells, and the time to cure. Simulated annealing, threshold acceptance, and old bachelor acceptance—a variant of threshold acceptance in which the trial length is set by users—were used to obtain better treatment schedules. The authors reported good results with all three techniques, but they showed simulated annealing resulted in the greatest computational effort.

Denton et al. (2009) investigated the optimal timing of statin therapy for patients with type 2 diabetes. This problem was formulated as a discrete time, finite horizon, discounted MDP in which patients transition through health states corresponding to varying risks of future complications, their history of complications, and death from other causes unrelated to diabetes. The objective was to maximize reward for QALYs minus costs of treatment. The optimal timing of treatment for patients was determined using three published risk models for predicting cardiovascular risk. The earliest time to start statins was age 40 for men, regardless of which risk model was used. However, for female patients, the earliest optimal start time varied by 10 years, depending on the risk model. Mason et al. (2012) extended this work to account for poor medication adherence. The authors used a Markov model to represent uncertain future adherence after medication was initiated. They observed that the optimal timing of statins should be up to 11 years later for patients with uncertain future adherence. However, they also found that improving adherence has a much larger effect on QALYs than delaying the timing of initiation.

Paltiel et al. (2004) constructed a simulation model to treat asthma. The model forecasted asthma-related symptoms, acute exacerbations, quality adjusted life expectancy, health-care costs, and cost-effectiveness. Their intent was to reduce asthma manifestations, improve life quality, and reduce costs of care. The authors pointed out that similar models could be constructed for the control of other subpopulation-wide diseases such as obesity, smoking, and diabetes.

A great deal of work has also been done on modeling CHD interventions. Cooper et al. (2006) provided an excellent review of many models used for this disease. Most of the models reviewed by the authors are decision trees, Markov processes, or simulation models. Decisions included when and what types of interventions, and what types of drugs to employ, at various stages of disease.

Dosage of Treatment

Given a particular treatment has been selected, the appropriate dosage must be determined. He et al. (2010) provided a discrete-state MDP model for determining gonadotropin dosages for patients undergoing in vitro fertilization-embryo transfer therapy. This work focused on patients with the chronic condition of polycystic ovaries syndrome that tend to be more sensitive to the gonadotropin treatment. The resulting policies from the MDP model were evaluated through simulation to determine the impact of misclassifying patients. In general, the use of OR techniques can be used to provide a better starting dosage with less fine tuning needed after initiation of treatment.

Dosage decisions are also important in radiation treatment planning. Several studies have focused on radiotherapy for cancer using mathematical optimization techniques. Although the vast majority of these treatment plans are designed by clinicians through intelligent trial and error, it is becoming

essential to use optimization for extremely complicated and complex plans. Holder (2004) used linear programming for intensity modulated radiotherapy treatment (IMRT). Ferris et al. (2004) discussed various optimization tools for radiation treatment planning. In both of these papers, the objective was to deliver a specified dose to the target area (above a minimum and below a maximum level of dosage) and spare or minimize damage to surrounding healthy tissue and nearby critical body structures and organs.

Organ Transplants

End-stage liver disease (ESLD) and end-stage renal disease (ESRD) have received a great deal of study in the OR literature. They are chronic conditions that can result in patients eventually needing liver or kidney transplants, respectively. Chronic liver disease or liver failure can result from many causes, including liver cancer and chronic hepatitis. Often, initial treatment of liver failure attempts to manage the underlying cause, followed by intensive care and management of complications such as bleeding problems. If patients continue to deteriorate to ESLD, liver transplantation may be the only option. Patients with chronic kidney disease have a continuing loss of renal function, leading to ESRD. Once a patient has ESRD, renal replacement therapy in the form of dialysis or kidney transplantation is necessary.

While organ transplants are the best long-term solution for patients with chronic liver or kidney disease, there is a shortage of organs for transplant and a growing waiting list of patients. OR techniques have been applied to optimize the allocation of organs and timing of transplants for increasing quality and length of life of the recipients. The allocation of kidneys and livers for transplantation is challenging because both living and cadaveric donors are possible. With living donors, there is more flexibility in the timing of the transplant, allowing for the transplant timing decision to be optimized. For both kidney and liver transplantation, there are challenging decisions about whether to use a living or cadaveric donor (if both are available), and when the transplant should occur. OR techniques have also aided in finding the greatest number of donor-recipient matches, considering the challenges of blood and tissue type compatibilities.

Alagoz et al. (2004) studied the question of the optimal timing of liver transplantation. They developed an MDP model to find the optimal timing for a patient to have a transplant from a living donor. The patients transitioned through health states defined by a scoring system for ESLD. With the donor assumed to be available at any time, the MDP maximized the patient's quality adjusted lifetime—striking a balance between having the transplant before the patient becomes too sick and waiting long enough due to the limited amount of time a patient can live after a transplant.

Su and Zenios (2004) presented an M/M/1 queueing model to determine if incorporating patient choice into allocation will improve efficiency and reduce waste of organs offered to patients but not accepted. Their model incorporated uncertain arrival of patients and organs, with the service process being the kidney transplant. Since organs cannot be stored, the service time was given by the interarrival time of organs. In addition to the traditional M/M/1 assumptions, each organ had a reward corresponding to its quality, and patients may reject an organ they believe has poor quality. The authors found that a first-come-first-serve policy can lead patients to refuse organs of lesser quality, leading to waste of up to 15% of organs. They also found that last-come-first-serve (LCFS) allocation lowers the wasteful effect of patient preference. While LCFS was not a feasible rule to implement, their results highlighted the need for adjustment of incentives associated with patient choice to prevent wasting organs.

A common way for patients to find organ donors is to ask willing family members or friends to be tested for compatibility. Another area, where OR has contributed, considers patients with willing donors that are not matches. Segev et al. (2005) considered the problem of paired kidney donation, matching two incompatible pairs with each other resulting in two successful transplants. The study considered a graph theory representation of a large pool of incompatible patient-donor pairs where each pair was represented with a node and two compatible pairs were linked with an edge. An algorithm based on the Edmonds matching algorithm (Edmonds 1965) was used to find all feasible matching solutions, and the best solution was chosen based on some predefined criteria, including the number of total matches and the number of transplant patients alive five years after the operation.

This matching strategy was compared to the first-accept scheme, which only finds one feasible solution, that is used in practice. The authors found that their algorithm could increase the total number of matches and take into account patient priorities.

Emerging Applications

Rapid advances in medicine are driving new OR research opportunities. As evidence of this, over the period from 2000–2010 the total number of health care related presentations at the Institute for Operations Research and Management Science (INFORMS) annual meeting has grown from 35 in 2000 to 281 in 2009 (Denton and Verter 2010). This section provides some specific examples of emerging areas of research.

Personalized Medicine

With the sequencing of the human genome and many advances in biomarkers for certain diseases, the idea of personalized medicine has received a great deal of attention. There are some examples of successful applications of personalized medicine, such as breast cancer treatment. However, for most diseases even basic risk factors are not yet considered as part of the standard guidelines. For example, gender is a well known risk factor for heart disease and stroke. While this has been known for decades, in many countries, including the U.S., the published treatment guidelines for control of risk factors such as cholesterol and blood pressure are the same for men and women. These examples point to opportunities to improve the design of screening and treatment guidelines through consideration of individual patient risk factors.

Decision Aids

The use of OR techniques in the development of decision aids is not as wide as in other areas of treatment choices. This is an area of research that must expand if OR models are to be translated into practice. Researchers have attempted to use artificial intelligence and computer science/information systems to provide decision support to the physician and/or patient. However, many clinicians still hesitate to use models for diagnosis or treatment. There are many possible reasons for the slow diffusion into practice. An important goal is the study of the clinician-model interface. In spite of adoption difficulties, there are examples of where OR has contributed significantly to treatment decisions. Several examples follow.

White et al. (1982) developed a quantitative model for diagnosing medical complaints in an ambulatory setting with the goal of reducing costs and improving quality of diagnoses. The model structure was influenced by three methods: decision analysis, partially observed semi-Markov decision process models, and multi-objective optimization therapy (MOOT). The authors used Bayesian-based modeling of disease progression and heuristics (a single-stage decision tree that reduces the amount of computation time and storage space per patient) to consider individual patient and physician preferences. For the MOOT heuristic, suggested by White et al. (1982), the list of possible diagnosis tests were provided, highlighting nondominated tests. The authors described a detailed example of the decision aid to treat a patient in an ambulatory setting.

Policies related to health information exchanges assume patients want to explicitly decide who can have access to their medical records. Marquard and Brennan (2009) tested this assumption by questioning 31 patients from a neurology clinic about their willingness to share information about their medication with a primary care physician, a neurologist, and an emergency room physician. Almost all patients decided to share their current medication usage with all three doctors citing the potential clinical care benefits. However, not all patients understood the possible effects of sharing this information. The use of realistic decision scenarios and structured conversations used in this study are likely to reveal more true patient preferences than abstract opinion surveys that are commonly used in practice. In addition to correctly identifying patient preferences, it is important to assess patient understanding of the consequences of their choices. Understanding the true willingness of patients to share health information is an important step in the development of decision aids and the inclusion of patient choices in medical decisions.

Using multi-attribute utility theory, Simon (2009) considered the choice of treatments for prostate cancer including surgery, external beam radiation, brachytherapy, and no treatment. The model used data collected from the medical literature to compute

probabilities regarding the likelihood of death and other side effects for each of the choices. The model also incorporated the patient's individual preferences regarding length of life and quality of life in view of the possible side effects (impotence, incontinence, and toxicity). The model evaluated each treatment alternative and compared the results for the particular patient.

Real Time Decision Making

Many medical treatment decisions must be made in real time. Depending on the particular application, the definition of real time could be anything from a few seconds to several minutes. Such applications can be highly demanding, often trading off the need for high quality decisions with available time.

One area in which OR has contributed to real time decision making is blood glucose control in patients with diabetes. Patients with type 1 diabetes are insulin dependent, and careful control of blood glucose within defined physiological limits is necessary to avoid a potentially life threatening occurrence of hypoglycemia (very low blood glucose that can lead to coma and/or death if not treated immediately). Blood glucose levels can change significantly over very short periods of time (seconds) depending on a variety of factors, such as caloric intake. The most common treatment for patients with type 1 diabetes is to inject insulin. However, the need for regular injection has a serious impact on a patient's quality of life. Research has been conducted on the design of closed loop control algorithms that could enable an implantable device to optimize insulin delivery (Parker et al. 2001).

Outpatient procedures can also pose a series of challenging decisions that must be made in real time (minutes). For instance, radiation treatment for cancer patients involves a series of complex decisions that can influence the effectiveness of treatment. One example is brachytherapy for prostate cancer treatment, that involves the implantation of radioactive seeds in close proximity to a tumor. The method of brachytherapy is to place seeds in and around a tumor such that dual goals of maximizing dose to the tumor and minimizing dose to healthy tissue are balanced. Due to changes that occur in tumor size and shape and the physical movement of healthy tissue and organs in proximity to the tumor over short time periods, such decisions must be made in real time at the point of placement. This real time analysis selects the actual placements of the seeds in the prostate from the thousands of possible locations, millimeters apart. Lee and Zaider (2008) presented a nonlinear mathematical-programming model to make location decisions using real time imaging information. They demonstrated a practical application in which the clinical goals of reduced complications (e.g., impotence and incontinence) and reduced costs ($5,600 per patient) were achieved.

Concluding Remarks

The use of OR for the study of disease treatment and screening decisions has a long history. Furthermore, advances in medicine are creating new challenges which are in turn resulting in new applications of OR and new methods. This article surveyed some of the significant contributions of OR methods, including mathematical programming, dynamic programming, and simulation. Contributions of OR to disease prevention and screening, long term management of chronic conditions, and several emerging application areas for OR were discussed.

Many examples of successful OR applications were described, as well as many challenges. For example, the availability of data for analyzing medical decisions is often more complex compared to other real-world decision situations. This is true for a variety of reasons including confidentiality concerns, the fragmented nature of health care delivery, and the lack of the requisite information systems. There are also challenges related to the fundamental difficulty of measuring criteria related to medical decision making, such as the cost to the patient as a result of a burdensome treatment plan. Finally, there are significant challenges in the translation of OR models from theory to practice.

References

Agur, Z., Hassin, R., & Levy, S. (2006). Optimizing chemotherapy scheduling using local search heuristics. *Operations Research, 54*(5), 826–846.

Alagoz, O., Maillart, L. M., Schaefer, A. J., & Roberts, M. S. (2004). The optimal timing of living-donor liver transplantation. *Management Science, 50*(10), 1420–1430.

Becker, N. G., & Starczak, D. N. (1997). Optimal vaccination strategies for a community of households. *Mathematical Biosciences, 139*(2), 117–132.

Braithwaite, R. S., Roberts, M. S., Chang, C., Goetz, M. B., Gibert, C. L., Rodriguez-Barradas, M. C., Shechter, S., Schaefer, A., Nuclfora, K., Koppenhaver, R., & Justice, A. C. (2008). Influence of alternative thresholds for initiating HIV treatment on quality-adjusted life expectancy: A decision model. *Annals of Internal Medicine, 148*, 178–185.

Brandeau, M. L., Owens, D. K., Sox, C. H., & Wachter, R. M. (1993). Screening women of childbearing age for human-immunodeficiency-virus — a model-based policy analysis. *Management Science, 39*(1), 72–92.

Brandeau, M. L., Sainfort, F., & Pierskalla, W. P. (2004). *Operations research and health care*. Boston, MA: Kluwer Academic Publishers.

Cooper, K., Brailsford, S. C., Davies, R., & Raftery, J. (2006). A review of health care models for coronary heart disease interventions. *Health Care Management Science, 9*(4), 311–324.

Denton, B. T., & Verter, V. (2010). Health care O.R. *ORMS Today, 37*(5).

Denton, B. T., Kurt, M., Shah, N. D., Bryant, S. C., & Smith, S. A. (2009). Optimizing the start time of statin therapy for patients with diabetes. *Medical Decision Making, 29*, 351–367.

Eddy, D. M. (1983). A mathematical model for timing repeated medical tests. *Medical Decision Making, 3*(1), 45–62.

Edmonds, J. (1965). Paths, trees, and flowers. *Canadian Journal of Mathematics, 17*, 449–467.

Engineer, F. G., Keskinocak, P., & Pickering, L. K. (2009). OR practice—catch-up scheduling for childhood vaccination. *Operations Research, 57*(6), 1307–1319.

Ferris, M. J., & Shepard, L. D. (2004). Optimization tools for radiation treatment mining in matlab. In M. L. Brandeau, F. Sainfort, & W. P. Pierskalla (Eds.), *Operations research and healthcare: A handbook of methods and applications* (pp. 775–806). Boston, MA: Kluwer Academic Publishers. chap. 30.

Hall, S. N., Jacobson, S. H., & Sewell, E. C. (2008). An analysis of pediatric vaccine formulary selection problems. *Operations Research, 56*(6), 1348–1365.

Hazen, G. B. (2004). Dynamic influence diagrams: Applications to medical decision modeling. In M. L. Brandeau, F. Sainfort, & W. P. Pierskalla (Eds.), *Operations research and healthcare: A handbook of methods and applications* (pp. 613–638). Boston, MA: Kluwer Academic Publishers. chap. 24.

He, M., Zhao, L., & Powell, W. B. (2010). Optimal control of dosage decisions in controlled ovarian hyperstimulation. *Annals of Operations Research, 178*(1), 223–245.

Holder, A. (2004). Radiotherapy treatment design and linear programming. In M. L. Brandeau, F. Sainfort, & W. P. Pierskalla (Eds.), *Operations research and healthcare: A handbook of methods and applications* (pp. 741–774). Boston, MA: Kluwer Academic Publishers. chap. 29.

Jacobson, S. H., Sewell, E. C., Deuson, R., & Weniger, B. G. (1999). An integer programming model for vaccine procurement and delivery for childhood immunization: A pilot study. *Health Care Management Science, 2*(1), 1–9.

Jacobson, S. H., Sewell, E. C., & Proano, R. A. (2006). An analysis of the pediatric vaccine supply shortage problem. *Health Care Management Science, 9*(4), 371–389.

Kaplan, E. H., Craft, D. L., & Wein, L. M. (2003). Analyzing bioterror response logistics: The case of smallpox. *Mathematical Biosciences, 185*(1), 33–72.

Kress, M. (2006). Policies for biodefense revisited: The prioritized vaccination process for smallpox. *Annals of Operations Research, 148*, 5–23.

Lee, H. L., & Pierskalla, W. P. (1988). Mass-screening models for contagious-diseases with no latent period. *Operations Research, 36*(6), 917–928.

Lee, E. K., & Zaider, M. (2008). Operations research advances cancer therapeutics. *Interfaces, 38*(1), 5–25.

Mandelblatt, J. S., Cronin, K., Bailey, S., Berry, D., de Koning, H., Draisma, G., Huang, H., Lee, S., Munsell, M., Plevritis, S., Ravdin, P., Schechter, C., Sigal, B., Stoto, M., Stout, N., van Ravesteyn, N., Venier, J., Zelen, M., & Feuer, E. J. (2009). Effects of mammography screening under different screening schedules: Model estimates of potential benefits and harms. *Annals of Internal Medicine, 151*, 738–747.

Marquard, J. L., & Brennan, P. F. (2009). Crying wolf: Consumers may be more willing to share medication information than policymakers think. *Journal of Health Information Management, 23*(2), 26–32.

Mason, J. E., England, D. A., Denton, B. T., Smith, S. A., Kurt, M., & Shah, N. D. (2012). Optimizing statin treatment decisions for diabetes patients in the presence of uncertain future adherence. *Medical Decision Making, 32*(1), 154–166.

Paltiel, A., Kuntz, K., Weiss, S., & Fuhlbrigge, A. (2004). Asthma policy model. In M. L. Brandeau, F. Sainfort, & W. P. Pierskalla (Eds.), *Operations research and healthcare: A handbook of methods and applications* (pp. 659–694). Boston, MA: Kluwer Academic Publishers. chap. 26.

Parker, R., Doyle, F., & Peppas, N. (2001). The intravenous route to blood glucose control. *IEEE Engineering in Medicine and Biology, 20*(1), 65–73.

Pierskalla, W. P, Brailer, D. J. (1994). *Applications of Operations Research in Health Care Delivery. 6*. North Holland.

Pignone, M., Earnshaw, S., Tice, J. A., & Pletcher, M. J. (2006). Aspirin, statins, or both drugs for the primary prevention of coronary heart disease events in men: A cost-utility analysis. *Annals of Internal Medicine, 144*, 326–336.

Rais, A., & Viana, A. (2010). Operations research in healthcare: A survey. *International Transactions in Operational Research, 18*, 1–31.

Rubin, D., Burnside, E., & Shachter, R. (2004). A Bayesian network to assist mammography interpretation. In M. L. Brandeau, F. Sainfort, & W. P. Pierskalla (Eds.), *Operations research and healthcare: A handbook of methods and applications* (pp. 695–720). Boston, MA: Kluwer Academic Publishers. chap. 27.

Schwartz, J. S., Kinosian, B. P., Pierskalla, W. P., & Lee, H. (1990). Strategies for screening blood for humanimmunodeficiency- virus antibody — use of a decision support system. *Journal of the American Medical Association, 264*(13), 1704–1710.

Segev, D. L., Gentry, S. E., Warren, D. S., Reeb, B., & Montgomery, R. A. (2005). Kidney paired donation and

optimizing the use of live donor organs. *JAMA — Journal of the American Medical Association, 293*(15), 1883–1890.
Shechter, S. M., Bailey, M. D., Schaefer, A. J., & Roberts, M. S. (2008). The optimal time to initiate HIV therapy under ordered health states. *Operations Research, 56*(1), 20–33.
Shwartz, M. (1978). Mathematical-model used to analyze breast-cancer screening strategies. *Operations Research, 26*(6), 937–955.
Simon, J. (2009). Decision making with prostate cancer: A multiple-objective model with uncertainty. *Interfaces, 39*(3), 218–227.
Su, X., & Zenios, S. (2004). Patient choice in kidney allocation: The role of the queueing discipline. *Manufacturing and Service Operations Management, 6*(4), 280–301.
Wein, L. M., & Zenios, S. A. (1996). Pooled testing for HIV screening: Capturing the dilution effect. *Operations Research, 44*(4), 543–569.
White, C. C., III, Wilson, E. C., & Weaver, A. C. (1982). Decision aid development for use in ambulatory health care settings. *Operations Research, 30*(3), 446–463.
Wu, J. T., Wein, L. M., & Perelson, A. S. (2005). Optimization of influenza vaccine selection. *Operations Research, 53*(3), 456–476.
Zhang, J., Denton, B. T., Balasubramanian, H., Shah, N. D., & Inman, B. A. (2012). Optimization of PSA screening policies: A comparison of the patient and societal perspectives. *Medical Decision Making, 32*(2), 337–349.

Distribution Selection for Stochastic Modeling

Donald Gross
George Mason University, Fairfax, VA, USA

Introduction

The choice of appropriate probability distributions is the most important step in any complete stochastic system analysis and hinges upon knowing as much as possible about the characteristics of the potential distribution and the physics of the situation to be modeled. Generally, the first thing that has to be decided is which probability distributions are appropriate to use for the relevant random phenomena describing the model. For example, the exponential distribution has the Markovian (memoryless) property. Is this a reasonable condition for the particular physical situation under study? Assume the problem is to describe the repair mechanism of a complex maintained system. If the service for all customers is fairly repetitive, then an assumption might be that the longer a failed item is in service for repair, the greater the probability that its service will be completed in the next interval of time (non-memoryless). In this case, the exponential distribution would not be a reasonable candidate for consideration. On the other hand, if the service is mostly diagnostic in nature (the trouble must be found and fixed), or there is a wide variation of service required from customer to customer so that the probability of service completion in the next instant of time is independent of how long the customer has been in service, the exponential with its memoryless property might indeed suffice.

The actual shape of the density function also gives quite a bit of information, as do its moments. One particularly useful measure is the ratio of the standard deviation to the mean, called the coefficient of variation (CV). The exponential distribution has a $CV = 1$, while the Erlang or convolution of exponentials has a $CV < 1$, and the hyperexponential or mixture of exponentials has a $CV > 1$. Hence, choosing the appropriate distribution is a combination of knowing as much as possible about distribution characteristics, the physics of the situation to be modeled, and statistical analyses when data are available.

Hazard Rate

An important concept that helps in characterizing probability distributions that is strongly associated with reliability modeling is the hazard-rate (also termed the failure-rate) function. This concept, however, can be useful in general when trying to decide upon the proper probability distribution to select. In the discussion that follows, the hazard rate will be related to the Markov property for the exponential distribution, and its use as a way to gain insight about probability distributions will be discussed.

Suppose it is desired to choose a probability distribution to describe a continuous lifetime random variable T with a cumulative distribution function (CDF) of $F(t)$. The density function, $f(t) = df(t)/dt$, can be interpreted as the approximate probability that the random time to failure will be in a neighborhood about a value t. The CDF is, of course, the probability that the time will be less than or equal to

the value t. Then the hazard rate $h(t)$ is defined as the conditional probability that the lifetime will be in a neighborhood about the value t, given that the time is already at least t. That is, if the situation deals with failure times, $h(t)dt$ is the approximate probability that the device fails in the interval $(t, t + dt)$, given it is working at time t.

From the laws of conditional probability, it can be shown that

$$h(t) = \frac{f(t)}{1 - F(t)}.$$

This hazard or failure-rate function can be increasing in t (called an increasing failure rate, or IFR), decreasing in t (called a decreasing failure rate, or DFR), constant (considered to be both IFR and DFR), or a combination. The constant case implies the memoryless or ageless property, and this holds for the exponential distribution, as will be shown. If, however, it is believed that the device ages and that the longer it has been operating the more likely it is that the device will fail in the next dt, then it is desired to have an $f(t)$ for which $h(t)$ is increasing in t; that is, an IFR distribution. This concept can be utilized for any stochastic modeling situation. For example, if instead of modeling lifetime of a device, the concern is with describing the service time of a customer at a bank, then, if service is fairly routine for each customer, then an IFR distribution would be desired. But if customers required a variety of needs (say a queue where both business and personal transactions were allowed), then a DFR or perhaps a CFR exponential might be the best choice.

Reversing the algebraic calculations, a unique $F(t)$ can be obtained from $h(t)$ by solving a simple linear, first-order differential equation, i.e.,

$$F(t) = -\exp\left(-\int_0^t h(u)\,du\right).$$

The hazard rate is another important information source (as is the shape of $f(t)$ itself) for obtaining knowledge concerning candidate probability distributions.

Consider the exponential distribution

$$f(t) = \theta \exp(-\theta t).$$

From the discussion above, it is easily shown that $h(t) = \theta$. Thus, the exponential distribution has a constant failure (hazard) rate and is memoryless. Suppose, for a particular situation, there is a need for an IFR distribution for describing some random times. It turns out that the Erlang has this property. The density function is

$$f(t) = \theta^k t^{k-1} \exp(-\theta t)/(k-1)!$$

(a special form for the gamma), with its CDF determined in terms of the incomplete gamma function or equivalently as a Poisson sum. From these, it is not too difficult to calculate the Erlang's hazard rate, that also has a Poisson sum term, but is somewhat complicated to ascertain the direction of $h(t)$ with t without doing some numerical work. It does turn out, however, that $h(t)$ increases with t and at a decelerating rate.

Suppose the opposite IFR condition is desired, that is, an accelerating rate of increase with t. The Weibull distribution can obtain this condition. In fact, depending on how the shape parameter of the Weibull is chosen, an IFR can be obtained with decreasing acceleration, constant acceleration (linear with t), or increasing acceleration, as well as even obtaining a DFR or the constant failure rate exponential. The CDF of the Weibull is given by

$$F(t) = 1 - \exp(-at^b)$$

and its hazard rate turns out to be the simple monomial $h(t) = abt^{b-1}$, with shape determined by the value of b (called the shape parameter).

As a further example in the process of choosing an appropriate candidate distribution for modeling, suppose, for an IFR that has a deceleration effect, such as the Erlang, there is a believe that the CV might be greater than one. This latter condition eliminates the Erlang from consideration. But, it is known that a mixture of (k) exponentials (often denoted by H_k) does have a CV > 1. It is also known that any mixture of exponentials is DFR. In fact, it can be shown that all IFR distributions have CV < 1, while all DFR distributions have CV > 1 (Barlow and Proschan 1975). Thus, if there is convincing evidence that the model requires an IFR, CV < 1 must be accepted. Intuitively, this can be explained as follows. Situations that have CV > 1

often are cases where the random variables are mixtures (say, of exponentials). Thus, for example, if a customer has been in service a long time, chances are that it is of a type requiring a lot of service, so the probability of completion in the next infinitesimal interval of width dt diminishes over time. Situations that have an IFR condition indicate a more consistent pattern among items, thus yielding a CV < 1.

Range of the Random Variable

Knowledge of the range of the random variable under study can also help narrow the possible choices in selecting an appropriate distribution. In many cases, there is a minimum value that the random variable can assume. For example, suppose the analysis concerns the interarrival times between subway trains, and it is given that there is a minimum time for safety of γ. The distributions discussed thus far (and, indeed, many distributions) have zero as their minimum value. Any such distribution, however, can be made to have a minimum other than zero by adding a location parameter, say γ. This is done by subtracting γ from the random variable in the density function expression. Suppose the exponential distribution is to be used, but we have a minimum value of γ. The density function would then become $f(t) = \theta \exp(-\theta[t - \gamma])$. It is not quite so easy to build in a maximum value if this should be the case. For this situation, a distribution with a finite range would have to be chosen, such as the uniform, the triangular or the more general beta distribution (Law and Kelton 1991).

Data

While much information can be gained from knowledge of the physical processes associated with the stochastic system under study, it is very advantageous to obtain data, if at all possible. For existing systems, data may already exist or can be obtained by observing the system. These data can then be used to gain further insight on the best distributions to choose for modeling the system. For example, the sample standard deviation and mean can be calculated, and it can be observed whether the sample CV is less than, greater than, or approximately equal to one. This would give an idea as to whether an IFR, DFR or the exponential distribution would be the more appropriate.

If enough data exist, just plotting a histogram can often provide a good idea of possible distributions from which to choose, since theoretical probability distributions have distinctive shapes (although some do closely resemble each other). The exponential shape of the exponential distribution is far different, for example, than the bell-shaped curve of the normal distribution.

There are rigorous statistical goodness of fit procedures to indicate if it is reasonable to assume that the data could come from a potential candidate distribution. These do, however, require a considerable amount of data and computation to yield satisfactory results. But, there are statistical packages, for example, Expert Fit (Law and Vincent 1995), which will analyze sets of data and recommend the theoretical distributions that are the most likely to yield the kind of data being studied.

Distribution selection (or input modeling, as it is sometimes called) is not a trivial procedure. But this is a most important aspect of stochastic analysis, since inaccuracies in the input can make the output meaningless. Fitting data to standard statistical distributions, which are mostly two-parameter distributions, limits focus on the first two moments only. There is evidence to suggest that this is not always sufficient (see Gross and Juttijudata 1997).

Finally, for emphasis, the point is made again that choosing an appropriate probability model is a combination of knowing as much as possible about the characteristics of the probability distribution being considered and as much as possible about the physical situation being modeled.

See

- ▶ Failure-Rate Function
- ▶ Hazard Rate
- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Reliability of Stochastic Systems
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Stochastic Input Model Selection

References

Barlow, R. E., & Proschan, F. (1975). *Statistical theory of reliability and life testing*. New York: Holt, Rinehart and Winston.

Forbes, C., Evans, M., Hastings, N., & Peacock, B. (2011). *Statistical distributions* (4th ed.). Hoboken, NJ: Wiley.

Gross, D., & Juttijudata, M. (1997). Sensitivity of output performance measures to input distributions in queueing simulation modeling. In S. Andradottir, K. J. Healy, D. H. Withers, & B. L. Nelson (Eds.), *Proceedings of the 1997 winter simulation conference*. Piscataway, NJ: IEEE.

Law, A. M., & Kelton, W. D. (1991). *Simulation modeling and analysis* (2nd ed.). New York: McGraw-Hill.

Law, A. M., & Vincent, S. (1995). *Expert fit user's guide*. Tucson, AZ: Averill M. Law and Associates.

DMU

Decision making unit.

See

► Data Envelopment Analysis

Documentation

Saul I. Gass
University of Maryland, College Park, MD, USA

Introduction

As many operations research studies involve a mathematical decision model that is quite complex in its form, it is incumbent upon those who developed the model and conducted the analysis to furnish documentation that describes the essentials of the model, its use, and its results. Of especial concern are those computer-based models that are represented by a computer program and its input data files. The most serious weakness in the majority of OR model applications, both those that are successful and those that fail, is the lack of documents that satisfy the minimal requirements of good documentation practices (Gass et al. 1981; Gass 1984). The reasons for requiring documentation are many-fold and include, among others, "to enable system analysts and programmers, other than the originators, to use the model and program," "to facilitate auditing and verification of the model and the program operations," and "to enable potential users to determine whether the model and programs will serve their needs" (Gass 1984).

The most acceptable view of model documentation is that which calls for documents that record and describe all aspects of the model development life-cycle. The life-cycle model documentation approach given in Gass (1979) calls for the production of 13 major documents. However, it is recognized that in terms of the basic needs of model users and analysts, these documents can be rewritten and combined into the following four manuals: *Analyst's Manual*, *User's Manual*, *Programmer's Manual, and Manager's Manual*. Brief descriptions of the contents of these manuals are given below; detailed tables of contents for each are given in Gass (1984).

Analyst's Manual

The analyst's manual combines information from the other project documents and is a source document for analysts who have been and will be involved in the development, revisions, and maintenance of the model. It should include those technical aspects that are essential for practical understanding and application of the model, such as a functional description, data requirements, verification and validation tests, and algorithmic descriptions.

User's Manual

The purpose of the user's manual is to provide (nonprogramming) users with an understanding of the model's purposes, capabilities, and limitations so they may use it accurately and effectively. This manual should enable a user to understand the overall structure and logic of the model, input requirements, output formats, and the interpretation and use of the results. This manual should also enable technicians to prepare the data and to set up and run the model.

Programmer's Manual

The purpose of the programmer's manual is to provide the current and future programming staff with the information necessary to maintain and modify the model's program. This manual should provide all the details necessary for a programmer to understand the operation of the software, to trace through it for debugging and error correction, for making modifications, and for determining if and how the programs can be transferred to other computer systems or other user installations.

Manager's Manual

The manager's manual is essential for computer-based models used in a decision environment. It is directed at executives of the organization who will have to interpret and use the results of the model, and support its continued use and maintenance. This manual should include a description of the problem setting and origins of the project; a general description of the model, including its purpose, objectives, capabilities, and limitations; the nature, interpretation, use, and restrictions of the results that are produced by the model; costs and benefits to be expected in using the model; the role of the computer-based model in the organization and decision structure; resources required; data needs; operational and transfer concerns; and basic explanatory material.

See

- ▶ Implementation
- ▶ Model Evaluation
- ▶ Model Management
- ▶ Practice of Operations Research and Management Science

References

Brewer, G. D. (1976). Documentation: An overview and design strategy. *Simulation & Games, 7*, 261–280.

Gass, S. I. (1979). *Computer model documentation: A review and an approach, National Bureau of Standards Special Publication 500–39, U.S. GPO Stock No. 033-003-02020-6*, Washington, DC.

Gass, S. I. (1984). Documenting a computer-based model. *Interfaces, 14*, 84–93.

Gass, S. I., Hoffman, K. L., Jackson, R. H. F., Joel, L. S., & Sanders, P. B. (1981). Documentation for a model: A hierarchical approach. *ACM Communications, 24*, 728–733.

NBS. (1976). *Guidelines for documentation of computer programs and automated data systems, FIPS PUB 38*. Washington, DC: U.S. Government Printing Office.

NBS. (1980). *Computer model documentation guide, NBS special publication 500-73*. Washington, DC: U.S. Government Printing Office.

Domain Knowledge

The knowledge that an expert has about a given subject area.

See

- ▶ Artificial Intelligence
- ▶ Forecasting

DP

- ▶ Dynamic Programming

DSS

- ▶ Decision Support Systems (DSS)

Dual Linear-Programming Problem

A companion problem defined by a linear-programming problem. Every linear-programming problem has an associated dual-programming program. When the linear-programming problem has the form

$$\begin{aligned} \text{Minimize} \quad & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

then its dual problem is also a linear-programming problem with the form

$$\begin{aligned} \text{Maximize} \quad & \boldsymbol{b}^T\boldsymbol{y} \\ \text{subject to} \quad & \boldsymbol{A}^T\boldsymbol{y} \leq \boldsymbol{c} \\ & \boldsymbol{y} \geq \boldsymbol{0} \end{aligned}$$

The original problem is called the primal problem. If the primal minimization problem is given as equations in nonnegative variables, then its dual is a maximization problem with less than or equal to constraints whose variables are unrestricted (free). The optimal solutions to primal and dual problems are strongly interrelated.

See

- ▶ Complementary Slackness Theorem
- ▶ Duality Theorem
- ▶ Symmetric Primal-Dual Problems
- ▶ Unsymmetric Primal-Dual Problems

References

Dantzig, G. B. (1963). *Linear programming and extensions*. Princeton, NJ: Princeton University Press.

Gass, S. I. (1984). *Linear programming* (5th ed.). New York: McGraw-Hill.

Duality Theorem

A theorem concerning the relationship between the solutions of primal and dual linear-programming problems. One form of the theorem is as follows: If either the primal or the dual has a finite optimal solution, then the other problem has a finite optimal solution, and the optimal values of their objective functions are equal. From this it can be shown that for any pair of primal and dual linear programs, the objective value of any feasible solution to the minimization problem is greater than or equal to the objective value of any feasible solution to the dual maximization problem. This implies that if one of the problems is feasible and unbounded, then the other problem is infeasible. Examples exist for which the primal and its dual are both infeasible. Another form of the theorem states: if both problems have feasible solutions, then both have finite optimal solutions, with the optimal values of their objective functions equal.

See

- ▶ Dual Linear-Programming Problem
- ▶ Strong Duality Theorem

Dualplex Method

A procedure for decomposing and solving a weakly-coupled linear-programming problem.

See

- ▶ Block-Angular System

Dual-Simplex Method

An algorithm that solves a linear-programming problem by solving its dual problem. The algorithm starts with a dual feasible but primal infeasible solution, and iteratively attempts to improve the dual objective function while maintaining dual feasibility.

See

- ▶ Dual Linear-Programming Problem
- ▶ Feasible Solution
- ▶ Primal-Dual Algorithm
- ▶ Simplex Method (Algorithm)

Dummy Arrow

A dashed arrow used in a project network diagram to show relationships among project items, a logical dummy, or to give a unique designation to an

activity, thus called a uniqueness dummy. A dummy or dummy arrow represents no time or resources.

See

▶ Network Planning

Dynamic Programming

Chelsea C. White III
Georgia Institute of Technology, Atlanta, GA, USA

Introduction

Dynamic programming (DP) is both an approach to problem solving and a decomposition technique that can be effectively applied to mathematically describable problems having a sequence of interrelated decisions. Such decision-making problems are pervasive. Determining a route from an origin (e.g., home) to a destination (e.g., school) on a network of roads requires a sequence of turns. Managing a retail store (e.g., that sells, say, television sets) requires a sequence of wholesale purchasing decisions.

Such problems share important characteristics. Each is associated with a criterion to be optimized: choosing the shortest or most scenic route from home to school, and the buying and selling of television sets by the retail store manager to maximize expected profit. Also, each problem has a structure such that a currently determined decision has impact on the future decision-making environment. In going from home to school, the turn currently selected will determine the geographical location of the next turn decision; in managing the retail store, the number of items ordered today will affect the level of inventory next week.

Roots and Key References

In his 1957 book, Richard Bellman described the concept of DP and its broad potential for application. See Bellman's earlier publications that describe his initial developments of DP (Bellman 1954a, b); also see Bertsekas (1987); Denardo (1982); Heyman and Sobel (1984); Hillier and Lieberman (2004, Chapter 10), and Ross (1983) for in depth descriptions and applications of DP.

Central to the philosophy and methodology of DP is the Principle of Optimality, as related to the following multistage decision problem (Bellman 1957). Let $\{q_1, q_2, \ldots q_n\}$ be a sequence of allowable decisions called a policy; specifically, an n-stage policy. A policy that yields the maximum value of the related criterion function is called an optimal policy. Decisions are based on the state of the process, that is, the information available to make a decision. The basic property of optimal policies is expressed by the following:

> *Principle of Optimality*: An optimal policy has the property that whatever the initial state and the initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision (Bellman 1957).

The Principle of Optimality can be expressed as an optimization problem over the set of possible decisions by a recursive relationship, the application of which yields the optimal policy. This is illustrated next by two examples.

1. An itinerary selection problem. The problem is to find the shortest path from home to school. A map of the area describes the network of streets that includes home and school locations, intermediate intersections, connecting streets, and the distance from one intersection to any other intersection that is directly connected by a street. The DP model of this problem is as follows. Let N be the set composed of home, school, and all intersections. An element of N is termed a node. For simplicity, assume all of the streets are one-way. A street is described as an ordered pair of nodes; that is, (n, n') is the street going from node n to node n' (n' is an immediate successor of node n). Let $m(n, n')$ be the distance from node n to node n'; that is, $m(n, n')$ represents the length of street (n, n').

   The problem is examined recursively as follows. Let $f(n)$ equal the shortest distance from the node n to the goal node school. The objective is to find f (home), the minimum distance from home to school, and a path from home to school that has a distance equal to f (home), a minimum distance path.

Note that $f(n) \leq m(n, n') + f(n')$ for any node n' that is an immediate successor of node n. Assume that an immediate successor n'' of n such that $f(n) = m(n, n'') + f(n'')$ has been found. Then, if at node n, it seems reasonable that the street that takes us to node n'' is traversed. Thus, the evaluation of all of the values $f(n)$ determine both f (home) and a minimum distance path from home to school. Formally, determination of these values can proceed recursively from the equation $f(n) = \min\{m(n, n') + f(n')\}$, where the minimum is taken over all nodes n' that are immediate successors of node n and where f (school) $= 0$ is the initial condition.

2. An inventory problem. Let $x(t)$ be the number of items in stock at the end of week t, $d(t + 1)$ the number of customers wishing to make a purchase during week $t + 1$, and $u(t)$ the number of items ordered at the end of week t and delivered at the beginning of week $t + 1$. Although it is unlikely that $d(t)$ is known precisely, assume the probability that $d(t) = n$ is known for all $n = 0, 1, \ldots$. Keeping backorders, then $x(t + 1) = x(t) - d(t + 1) + u(t)$. A reasonable objective is to minimize the expected cost accrued over the period from $t = 0$ to $t = T$ $(T > 0)$ by choice of $u(0), \ldots, u(T - 1)$, assuming that ordering decisions are made on the basis of the current inventory level, that is, the mechanism that determines $u(t)$ (e.g., the store manager) is aware of $x(t)$, for all $t = 0, \ldots, T - 1$. Costs might include a shortage cost (a penalty if there is an insufficient amount of inventory in stock), a storage cost (a penalty if there is too much inventory in stock), an ordering cost (reflecting the cost necessary to purchase items wholesale), and a selling price (reflecting the income received when an item is sold; a negative cost). Let $c(x, u)$ represent the expected total cost to be accrued from the end of week t till the end of week $t + 1$, given that $x(t) = x$ and $u(t) = u$. Then the criterion to be minimized is

$$E\{c[x(0), u(0)] + \ldots + c[x(T-1), u(T-1)]\},$$

where E is the expectation operator associated with the random variables $d(1), \ldots, d(T)$.

This problem can be examined recursively. Let $f(x, t)$ be the minimum expected cost to be accrued from time t to time T, assuming that $x(t) = x$. Clearly, $f(x, T) = 0$. Note also that

$$f[x(t), t] \leq c[x(t), u(t)] + E\{f[x(t) - d(t+1) + u(t), t+1]\}$$

for any available $u(t)$. As was true for Example 1, an order number u'' which is such that

$$f[x(t), t] = c[x(t), u''] + E\{f[x(t) - d(t+1) + u'', t+1]\}$$

is an order to place at time t when the current inventory is $x(t)$. Thus, the recursive equation determines both $f(x, 0)$ for all x and the order number as a function of current inventory level.

Common Characteristics

Two key aspects of DP are the notion of a state and recursive equations. The state of the DP problem is the information that is currently available to the decision maker on which to base the current decision. For example, in the itinerary selection problem, the state is the current node; in the inventory problem, the state is the current number of items in stock. In both examples, how the system arrived at its current state is inconsequential from the perspective of decision making. For the itinerary selection problem, all that is needed is the current node and not the path that lead to that node to determine the best next street to traverse. The determination of the number of items to order this week depends only on the current inventory level equations (other names include functional equations and optimality equations) that can be used to determine the minimum expected value of the criterion and an optimal sequence of decisions that depend on the current node or current inventory level. In both cases, the recursive equations essentially decompose the problem into a series of subproblems, one for each node or current state value.

See

- Approximate Dynamic Programming
- Bellman Optimality Equation
- Dijkstra's Algorithm
- Markov Decision Processes
- Network

References

Bellman, R. E. (1954a). Some problems in the theory of dynamic programming. *Econometrica, 22*(1), 37–48.

Bellman, R. E. (1954b). Some applications of the theory of dynamic programming. *Journal of the Operations Research Society of America, 2*(3), 275–288.

Bellman, R. E. (1957). *Dynamic programming*. Princeton, NJ: Princeton University Press.

Bertsekas, D. P. (1987). *Dynamic programming: Deterministic and stochastic models*. Englewood Cliffs, NJ: Prentice-Hall.

Denardo, E. V. (1982). *Dynamic programming: Models and applications*. Englewood Cliffs, NJ: Prentice-Hall.

Dreyfus, S., & Law, A. (1977). *The art and theory of dynamic programming*. New York: Academic Press.

Heyman, D. P., & Sobel, M. J. (1984). *Stochastic models in operations research* (Vol. II). New York: McGraw-Hill.

Hillier, F. S., & Lieberman, G. J. (2004). *Introduction to operations research* (8th ed.). New York: McGraw-Hill.

Lew, A., & Mauch, H. (2007). *Dynamic programming: A computational tool*. New York: Springer.

Ross, S. M. (1983). *Introduction to stochastic dynamic programming*. New York: Academic Press.

E

Earliest Finish Time

The earliest possible time an activity can be completed without reducing the duration of any of the preceding activities as described in a project network. It is simply the sum of the earliest start time for the activity and the duration of the activity.

See

- ▶ Critical Path Method (CPM)
- ▶ Network Planning
- ▶ Program Evaluation and Review Technique (PERT)

Earliest Start Time

The earliest possible time an activity can begin without reducing the duration of any of the preceding activities as described in a project network. It is calculated by summing the durations of all activities on the longest path leading to the event that identifies the beginning of the activity.

See

- ▶ Critical Path Method (CPM)
- ▶ Network Planning
- ▶ Program Evaluation and Review Technique (PERT)

Early British OR

Maurice W. Kirby and Graham K. Rand
Lancaster University, Lancaster, UK

The term operational research (OR) was first used in the later 1930s to describe the process of evaluation of radar as an essential aid to the air defense of Great Britain. Emanating from the work of Robert Watson-Watt of the National Physical Laboratory, the novel concept of controlled interception of enemy aircraft by electronic means could only be tested by practical experiment entailing the application of quantitative techniques of analysis. Under the auspices of the Committee for the Scientific Survey of Air Defence, the resulting research program was an outstanding success insofar as the home radar chain proved decisive in more than offsetting RAF Fighter Command's numerical inferiority during the Battle of Britain in 1940. With its credentials intact as a means of enhancing the effectiveness of an entire military command at a critical stage in the war, OR was thereafter diffused throughout the greater part of the British armed forces both at home and abroad.

At the end of the war, in conformity with the experience in Fighter Command, operational researchers, under the leadership of Blackett, regarded as the "father of Operational Research,"could congratulate themselves on their substantial and, on occasion, decisive contributions to the war effort in a number of theaters. In the RAF Coastal Command and the Admiralty, for example, operational

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

researchers were responsible for a sequence of tactical innovations which led to the defeat of the U-boats in the North Atlantic. For Operation Overlord, moreover, a detailed plan of targets in the French railway system was devised on the basis of quantitative assessment of their capacity for enemy logistical reinforcement, thereby enabling RAF Bomber Command to offer outstanding tactical support to the allied invading forces.

With achievement on this scale it is hardly surprising that the advocates of OR should have sought to secure its peacetime future via its diffusion beyond the military sector. In this respect, the period from 1945 to the mid-l970s may be viewed as the golden age of British OR, at least in methodological terms, when the new discipline was diffused into the nationalized industries, departments of civil government, and the corporate sector. Institutional developments were also notable, with the transformation of an informal OR Club, founded in 1948, created following the initiative of Sir Charles Goodeve, into the Operational Research Society in 1953, together with the establishment of that ultimate hallmark of professional status — a specialist journal: *The Operational Research Quarterly* first appeared in 1950, and continued in that form until 1978 when it was redesignated as the official *Journal of the Operational Research Society* with twelve annual issues. These developments may be viewed as a testament to the growing professionalization of the discipline, the subscription to common methodologies, and belief in its economic and social utility. Coincidentally, as OR was being recognized as a practically useful tool of analysis for executive decision makers, its public profile was further enhanced by formal academic recognition. By the early 1960s, several universities were making provision for OR taught courses for postgraduate students, and this served as a precursor to the expansion of the subject at undergraduate level after 1964.

A combination of acknowledged utilitarian value and formal academic recognition within thirty years of its foundation is consistent with an impressive trajectory of achievement for any human endeavor. Yet despite open acknowledgment of its wartime role, it is instructive to note that beyond the nationalized coal industry and the British Iron and Steel Research Association (BISRA), OR made little headway in civil government in the early postwar years. This is all the more surprising in view of the election in 1945 of Britain's first majority Labor Government ostensibly committed to centralized measures of economic and social planning. Indeed, the new government sought to alleviate resource constraints at the level of manufacturing industry by emphasizing the need for enhanced efficiency both in terms of managerial standards and the organization of work. In this respect, the official Committee on Industrial Productivity proclaimed the virtues of OR from the standpoint of "a scientific approach to running industry," and also as an aid to the machinery of government in planning the allocation of resources.

In any event, a combination of civil service conservatism and mounting political opposition to centralized measures of economic planning were sufficient to severely constrain the diffusion of OR in this particular context for a generation to come. Even within the coal and iron and steel industries it is evident that successful diffusion was heavily dependent on the efforts of specific individuals, in the former case, Sir Charles Ellis in his capacity as Scientific Member of the National Coal Board (NCB), and in the latter Sir Charles Goodeve, the first director of BISRA. Both men professed a firm appreciation of the wartime benefits of OR and were determined to apply quantitative methods of analysis as an aid to rational managerial decision making in their particular spheres. Goodeve, however, possessed a missionary zeal for the dissemination of OR beyond the iron and steel industry to embrace the manufacturing sector as a whole. From the late 1940s to the late 1970s, his advocacy was vigorous and persuasive and there is much documentary and anecdotal evidence to suggest that Goodeve, more than any other individual, was responsible for the adoption of OR in an increasing range of firms in the corporate sector.

Other key individuals in the dissemination process were Goodeve's colleagues at BISRA, Roger Eddison and Roger Collcutt, and in the private sector iron and steel firms, Steve Cook, Stafford Beer and Keith Tocher, and in the coal industry, Donald Hicks, Pat Rivett and Brian Houlden. It is significant that several of these key practitioners were instrumental in establishing OR as an academic discipline worthy of advanced study in the university sector. In this setting, the leading role was fulfilled by Pat Rivett who was appointed to the new University of Lancaster in 1964

as the first UK Professor of OR. This followed in the wake of his efforts to publicize the value of OR in higher education after his election as President of the OR Society in 1960.

The diffusion of OR was also facilitated powerfully by the development of analytical techniques and the innovation, during the course of the 1950s, of digital computers geared to the needs of commercial firms. In the former case, there were two outstanding developments. The first was linear programming emanating from the USA in the later 1940s. Although the foremost pioneer in the UK was Steven Vajda working at the Admiralty Research Laboratory in the early 1950s, the first significant use of linear programming in British industry took place in the NCB from the mid-1950s. Thereafter, it spread rapidly to the electricity, chemical and oil industries with British Petroleum (BP) developing a substantial OR effort in this area by the late 1960s. Complementing linear programming was the technique of simulation as an aid to rational decision making. The modeling of complex systems simulation, entailing the creation of a physical or mathematically-based analogue, was pioneered in the NCB and BISRA in the 1950s. In addition, an abstract approach could be adopted utilizing mathematical equations and logical relationships. However, such procedures were time consuming and repetitive and were thus ideally suited to digital computerization, a procedure which became increasingly commonplace in the coal and steel industries during the 1960s.

Innovative OR techniques, therefore, complemented by the development of the computer, had important roles to play in the diffusion of OR and in extending its practical scope. During the 1960s, the quantum leap in computer power through the use of the transistor, and the associated hype and mystique attached to computers, helped to ensure their commercial proliferation. Although there was a double-edged factor in this development — in the sense that the specialized needs of OR departments were rendered increasingly subordinate to overall company computing needs — the fact remains that by the mid-1960s computers were an essential tool for operational researchers, encouraging more powerful linear programming codes and new high-level simulation languages.

If the later 1950s and the 1960s witnessed methodological and technological developments conducive to the spread of OR, it remains to be said that the period also gave birth to corporate OR in a variety of business settings both in manufacturing and services. The precipitating factor was an upsurge in merger activity which entailed considerable changes in managerial styles and structures. Hitherto, the managerial organization of British business had reflected a continuing commitment to personal capitalism or family influence and control. This fact in itself had served to limit the extent of professionalism in British management, especially in an Anglo-American context.

The merger movement of the 1960s, however, in giving birth to a U.S.-style corporate economy, resulted in the recruitment of professional managerial hierarchies on the U.S. model, often on the advice of North American management consultants. In these circumstances, the Anglo-American corporate gap was closed, but in such a way as to encourage the adoption of OR as a management tool. As the structure of business became increasingly complex, the administration of larger scale enterprises was rendered more tractable by a combination of new instruments of control and decision making enshrined in the computer and OR. In a survey carried out on behalf of the Operational Research Society in 1967, 766 OR groups were identified in industrial sectors as wide ranging as printing and publishing, shoe manufacturing, textiles, glass, brewing, transport and banking. Although the figure embraced a large number of one-person teams, it remains true that by the end of the 1960s, corporate OR was an established fact with an impressive sectoral coverage.

Coincident with the revolution in the managerial structure of British business, the 1960s also witnessed a reforming movement at the level of central government which had major implications for the British OR community. This concerned the efforts of the post-1964 Labor government to reform the Civil Service in the wake of the recommendations of the Fulton Committee. The committee's terms of reference, embracing management as well as recruitment, provided the Operational Research Society with an excellent opportunity to advance the cause of OR as a solution to major policy problems. The formal evidence submitted by the Society fell on fertile ground in the Treasury, where the Permanent Secretary, Sir William Armstrong, proved entirely receptive to the Society's cause. Thus, following the publication of the Fulton Report, the Treasury was divided in 1968 with the Treasury retaining responsibility for economic policy and the

management of the Civil Service being taken over by a new Civil Service Department headed by Armstrong. It was the latter which provided a home for a central government OR facility and this served as a precursor to the establishment of OR groups in an increasing range of government departments after 1970.

In surveying the early history of British OR, a number of interrelated themes can be identified. In the first instance, there can be no doubting the vigor and enthusiasm of the discipline's postwar advocates, motivated by the conviction that the application of OR was conducive to the public good in a variety of settings, both economic and social. It is also true that the peacetime diffusion of OR was propelled, in part, by perceptions of its wartime role and status–the fact that operational researchers had outstanding achievements to their credit, both strategic and tactical, and had enjoyed privileged access to military commanders of the highest rank. But these war-induced relationships could not be replicated in peacetime. The organization of civil affairs was always unlikely to offer the same opportunities for high-level influence if only because the sheer urgency of wartime problem-solving was lacking. It is true that OR was adopted with some success by the postwar nationalized industries and penetrated into the corporate sector from the mid-1950s with a notable acceleration in diffusion during the 1960s. This was one consequence of the professionalization of decision-making, entirely consistent with the continuing demise of family influence and control in British business.

But although OR enjoyed substantial penetration into civil affairs, questions remained about its relative position in organizational structures. In the NCB, the high status of OR was underwritten from the outset by the presence on the board of a scientific member, and, in the case of BISRA, the long standing director was himself an ardent advocate of the discipline as a result of wartime experience. Elsewhere, however, the status of OR and the level at which advice was delivered was dependent upon the idiosyncratic enthusiasms of individual executives. This was exemplified by the rapid diffusion of OR within the civil service after 1967 as a result of the strong personal backing of Sir William Armstrong.

A second and related theme concerns the nature of peacetime OR. To the extent that it was outstandingly tactical it was but one element in the decision- making process and therefore vulnerable to slippage of status according to the perceived value of the work done. Even when OR had a powerful role to play in determining overall strategy, as in the case of the oil companies, it was dependent not only upon high-level backing, but also on a stable external environment. As the world became more turbulent after 1970, the limitations of linear programming became all too evident, especially for global corporations increasingly sensitive to prevailing economic and political conditions. Within Britain itself, the end of the long postwar boom — signaled by the economic recession of the early 1970s — underlined the still uncertain status of OR as a number of corporate OR groups were downsized, closed, or absorbed elsewhere. This was repeated in the early 1980s by which time the British OR community in the university sector was attempting to come to terms with Russell Ackoff's powerful and controversial attack on the extreme mathematization of the discipline in that particular context. It is clear, therefore, that British OR was entering a new phase of its evolution after 1970. The expansionary phase, as indicated by the rapid growth of membership of the Operational Research Society in the 1960s, was now to be followed by a period of introspection. This was most evident at the level of academic OR, but practitioners too began to modify their claims to the status of expert problem solvers by taking on a charge agent role conducive to the enhancement of decision-making processes in general.

See

- ▶ Air Force Operations Analysis
- ▶ Center for Naval Analyses
- ▶ Operations Research Office and Research Analysis Corporation
- ▶ Operational Research Society (ORS)
- ▶ RAND Corporation

References

Collcutt, R. H. (1965). *The first twenty years of operational research*. London: BISRA.

Davies, M., Eddison, R. T., & Page, T. (Eds.). (1957). *Proceedings of the first international conference on operational research (Oxford 1957)*. London: English Universities Press.

Keys, P. (1991). *Operational research and systems: The systemic nature of operational research*. New York: Plenum Press.

Kirby, M. (2010). Charles Frederick goodeve (1904), Chapter 5. In A. A. Assad & S. I. Gass (Eds.), *Profiles in operations research: pioneers and innovators*. New York: Springer Scientific.

Kirby, M. W. (2000). Operations research and the defeat of Nazi Germany. *Military Operations Research, 5*(4), 57–70.
Kirby, M. W. (2003). *Operational research in war and peace: The British experience from the 1930s to 1970*. London: Imperial College Press.
Kirby, M. W., & Capey, R. (1997a). The air defence of Great Britain, 1920–1940: An operational research perspective. *Journal of the Operational Research Society, 48*, 555–568.
Kirby, M. W., & Capey, R. (1997b). The area bombing of Germany in World War II: An operational research perspective. *Journal of the Operational Research Society, 48*, 661–667.
Kirby, M. W., & Capey, R. (1998). The origins and diffusion of operational research in the UK. *Journal of the Operational Research Society, 49*, 307–326.
Kirby, M. W., & Godwin, M. T. (2010). The 'invisible science': Operational research for the British Armed forces after 1945. *Journal of the Operational Research Society, 61*, 68–81.
Kirby, M., & Rosenhead, J. (2010a). Patrick Blackett (1897), Chapter 1. In A. A. Assad & S. I. Gass (Eds.), *Profiles in operations research: Pioneers and innovators*. New York: Springer Scientific.
Kirby, M., & Rosenhead, J. (2010b). Russell Lincoln Ackoff (1919), Chapter 21. In A. A. Assad & S. I. Gass (Eds.), *Profiles in operations research: Pioneers and innovators*. New York: Springer Scientific.
Rand, G. K. (2010). Berwyn Hugh Patrick "Pat"Rivett (1923), Chapter 26. In A. A. Assad & S. I. Gass (Eds.), *Profiles in operations research: Pioneers and innovators*. New York: Springer Scientific.
Ranyard, J. L. (1988). A history of OR and computing. *Journal of the Operational Research Society, 39*, 1973–1986.
Rosenhead, J. (1989). Operational research at the crossroads: Cecil Gordon and the development of post-war OR. *Journal of the Operational Research Society, 40*, 3–28.
Rosenhead, J. (2010). Anthony Stafford Beer (1926), Chapter 32. In A. A. Assad & S. I. Gass (Eds.), *Profiles in operations research: Pioneers and innovators*. New York: Springer Scientific.
Tomlinson, R. C. (1971). *OR comes of age*. London: Tavistock.
Trefethen, F. N. (1954). A history of operational research. In J. F. McCloskey & F. N. Trefethen (Eds.), *Operational research for management* (pp. 25–30). Baltimore: Johns Hopkins Press.

Econometrics

Harry H. Kelejian and Ingmar R. Prucha
University of Maryland, College Park, MD, USA

Introduction

Literally speaking, econometrics stands for measurement in economics. Broadly speaking, econometrics is concerned with the empirical analysis of economic relationships. While early empirical work goes back at least to Sir William Petty's political arithmetic in the seventeenth century, econometrics as a field was firmly established through the foundation of the Econometric Society in 1930. Publication of its journal, *Econometrica*, started in 1933. The scope of the society is defined as follows: "The Econometric Society is an international society for the advancement of economic theory in its relation to statistics and mathematics . . ." Samuelson et al. (1954, p. 142) in a report on *Econometrica* defined econometrics

> . . . as the quantitative analysis of actual economic phenomena based on the concurrent development of theory and observation, related by appropriate methods of inference.

Similar definitions can be found in most econometric texts. For example, Goldberger (1964, p. 1) defined econometrics

> . . . as the social science in which the tools of economic theory, mathematics, and statistical inference are applied to the analysis of economic phenomena. Its main objective is to give empirical content to economic theory. . .

Single Equation Regression Models

Much of the early work in econometrics is related to the classical linear regression model

$$y_t = x_t\beta + u_t, \quad t = 1, \ldots, n, \tag{1}$$

where y_t is the tth observation on the dependent variable, x_t is the $1 \times k$ vector of observations on the explanatory variables, β is a $k \times 1$ vector of unknown parameters and u_t is the tth disturbance term. The assumptions of the classical model are: (i) $E(u_t) = 0$, (ii) $E(u^2_t) = \sigma^2$ and $E(u_t\, u_s) = 0$ for $t \neq s$, (iii) x_t is nonstochastic and (iv) $X = (x'_1, \ldots, x'_n)'$ has full column rank. Under these assumptions, the Gauss-Markov theorem implies that the ordinary least squares estimator is best (in the sense of having the smallest variance covariance matrix) within the class of linear unbiased estimators. If the disturbances are normally distributed, exact small sample inference is available. If normality is not maintained, then approximate inference is possible under additional assumptions on x_t and u_t.

The nature of economic data and models are such that the above assumptions are restrictive in certain applications, and hence, various extensions of the classical model have been considered. In particular, disturbances have been permitted to be autocorrelated and/or to have different variances, that is, to be heteroskedastic. Other extensions permit for the regressors to be stochastic. Stochastic regressors arise, for example, if the regressors are measured with error. They also arise in dynamic models in which one or several of the regressors depend on lagged values of the dependent variable. Models in which the parameters are permitted to vary deterministically or stochastically from observation to observation have also been considered. Still other extensions relate to sample selection issues. Text presentations of the issues discussed above are, for example, given in Amemiya (1985), Davidson and MacKinnon (1993), Judge et al. (1985), and Schmidt (1976).

Simultaneous Equation Models

The economy is a complex system of relationships. For this reason, economic models often involve more than one equation and so more than one dependent variable. To see the issues involved consider the following system of m equations:

$$y_t = y_t B + z_t C + u_t, \qquad t = 1, \ldots, n, \tag{2}$$

where y_t is a $1 \times m$ vector of the jointly dependent variables, $z_t = (y_{t-1}, \ldots, y_{t-h}, x_t)$ where x_t is a $1 \times k$ vector of nonstochastic variables, u_t is a $1 \times m$ vector of disturbances, and B and C are correspondingly defined matrices of parameters.

Basic assumptions for the model are: (i) u_t is i.i.d. with finite fourth moments and $\mathrm{E}(u_t) = 0$, $\mathrm{E}(u'_t u_t) = \Omega$ with Ω nonsingular, (ii) $(I - B)$ is nonsingular and the diagonal elements of B are zero, (iii) $n^{-1} \sum x'_t x_{t-\tau} \to Q(\tau)$ where the matrices $Q(\tau)$ are finite, and nonsingular for $\tau = 0$, (iv) the system is dynamically stable. Since $(I - B)$ is invertible the system can be solved as

$$\begin{aligned} &y_t = z_t \Pi + v_t, \\ &\Pi = C(I - B)^{-1} \text{ and } v_t = u_t (I - B)^{-1}. \end{aligned} \tag{3}$$

In the literature, equations (2) and (3) are called the structural and reduced form of the model, respectively. The parameters in B and C are generally not identified and hence, not consistently estimable without additional parameter restrictions. These parameter restrictions often take the form of exclusion restrictions based on economic theory; that is, theory may suggest that every variable does not appear in every equation and so certain elements of B and C are specified to be zero.

As is obvious from (3), the elements of y_t depend in general on all of the elements of u_t. As a consequence the structural equations in (2) cannot in general be estimated consistently by ordinary least squares. Fundamental work on estimation and identification of the model in (2) was done by the Cowles Foundation, which focused on the maximum likelihood technique based on the normal distribution; see Koopmans (1950) and Hood and Koopmans (1953). Estimation procedures developed later were typically based on instrumental variable techniques which do not require specific distributional assumptions; see Basmann (1957) and Theil (1953) for early fundamental contributions, and, for example, Amemiya (1985), Davidson and MacKinnon (1993), Judge et al. (1985) and Schmidt (1976) for later text presentations.

In recent years, the model in (2) has been generalized in ways that are similar to those mentioned above in reference to model (1). In addition, starting with the fundamental contributions of Jenrich (1969) and Malinvaud (1970), estimation theory has been developed for nonlinear counterparts to models (1) and (2); for recent presentations of estimation theory for dynamic nonlinear systems see, for example, Gallant and White (1988) and Pötscher and Prucha (1991a, b). Finally, Bayesian extensions of these models have been considered, see, for example, Zellner (1971) for early fundamental work and Judge et al. (1985) for a more recent text presentation.

Other Modeling Techniques

(a) *Time series models* — An important class of models used to describe economic data are autoregressive moving average (ARMA) models. These models have been popularized in economics by Box and Jenkins (1976); for a recent discussion of time series techniques see, for example, Brockwell and Davis (1991), and Harvey (1993).

A stationary stochastic process y_t (time series) that satisfies for every t

$$y_t = a_1 y_{t-1} + \cdots + a_p y_{t-p} + \varepsilon_t + b_1 \varepsilon_{t-1} + \cdots + b_q \varepsilon_{t-q} \quad (4)$$

where $\mathrm{E}(\varepsilon_t) = 0, E(\varepsilon^2{}_t) = \sigma^2 \varepsilon$ and $E(\varepsilon_t \varepsilon_s) = 0$ for $t \neq s$ is called an ARMA(p, q) process. If y_t was obtained by differencing some process z_t, then z_t is called an autoregressive integrated moving average (ARIMA) process. If the specification in (4) also permits nonstochastic regressors, then the corresponding processes are called ARMAX and ARIMAX, respectively. Clearly, the reduced form in (3) can be viewed as an ARMAX model. Although ARMAX models do not describe the structure of the system, they have been found, for example, to be useful for prediction purposes.

An important recent development in the time series literature is the introduction of the concept of cointegration as an equilibrium relationship between integrated variables. This development has particular appeal to economists because many economic variables appear to have random walk representations but yet certain linear combinations of them appear to be stationary. The basic ideas were proposed by Granger (1981); recent extensions and developments are discussed in Davidson and MacKinnon (1993), and Engle and Granger (1991).

(b) *Qualitative and limited dependent variable models* — Economists often formulate models to explain events which are at least partially qualitative in nature. For example, one might be interested in the factors determining whether or not a bank fails, a firm undertakes an investment, etc. More generally, such models could relate to events that are described by more than one category. Models relating to occupational choice, firm structure, and travel mode fall in this class.

Another class of models are limited dependent variable models. In these models the range of the dependent variable is constrained in some way. As one example, consider a model describing the selling price of a house. A limited dependent variable problem would arise if, for example, the only transactions that are recorded are those for which the selling price exceeds a certain dollar amount. The techniques involved for limited dependent variable models are similar to those in qualitative models. In recent years econometric models relating to qualitative and limited dependent variables have been generalized in ways that are similar to those described in the sections above. Excellent early reviews are given in McFadden (1974, 1976) and in Amemiya (1981). Later text presentations are given in Amemiya (1985), Maddala (1983), and Judge et al. (1985).

Concluding Remarks

This review has just touched upon a few of the major topics in econometrics. Many other topics were not covered, including model specification tests, rational expectations models, and model simulation.

See

- ▶ Economics and Operations Research
- ▶ Regression Analysis
- ▶ Time Series Analysis

References

Amemiya, T. (1981). Qualitative response models, a survey. *Journal of Economic Literature, 19*, 1483–1536.

Amemiya, T. (1985). *Advanced econometrics*. Cambridge, MA: Harvard University Press.

Amemiya, T. (1994). *Introduction to statistics and econometrics*. Cambridge, MA: Harvard University Press.

Basmann, R. L. (1957). A generalized classical method of linear estimation of coefficients in a structural equation. *Econometrica, 25*, 77–83.

Box, G. E. P., & Jenkins, G. M. (1976). *Time series analysis, forecasting and control*. San Francisco: Holden Day.

Brockwell, P. J., & Davis, R. A. (1991). *Time series, theory and methods*. New York: Springer Verlag.

Davidson, R., & MacKinnon, J. G. (1993). *Estimation and inference in econometrics*. New York: Oxford University Press.

Engle, R. F., & Granger, C. W. J. (Eds.). (1991). *Long-run economic relationships, reading in cointegration*. Oxford: Oxford University Press.

Gallant, A. R., & White, H. (1988). *A unified theory of estimation and inference for nonlinear dynamic models*. New York: Basil Blackwell.

Goldberger, A. S. (1964). *Econometric theory*. New York: Wiley.

Granger, C. W. J. (1981). Some properties of time series data and their use in econometric model specification. *Journal of Econometrics, 16*, 121–130.

Greene, W. H. (2011). *Econometric analysis* (7th ed.). Englewood Cliffs, NJ: Prentice Hall.
Harvey, A. C. (1993). *Time series models*. Cambridge, MA: MIT Press.
Hood, W. C., & Koopmans, T. C. (Eds.). (1953). *Studies in econometric methods. Cowles commission monograph 14*. New York: Wiley.
Jenrich, R. I. (1969). Asymptotic properties of non-linear least squares estimators. *Annals of Mathematical Statistics, 40*, 633–643.
Judge, G. G., Griffiths, W. E., Hill, R. C., Lütkepohl, H., & Lee, T. C. (1985). *The theory and practice of econometrics* (2nd ed.). New York: Wiley.
Koopmans, T. C. (Ed.). (1950). *Statistical inference in dynamic economic models. Cowles commission monograph 10*. New York: Wiley.
Maddala, G. S. (1983). *Limited-dependent and qualitative variables in econometrics*. Cambridge: Cambridge University Press.
Malinvaud, E. (1970). The consistency of nonlinear regressions. *Annals of Mathematical Statistics, 41*, 956–969.
McFadden, D. (1974). Conditional logit analysis of qualitative choice behavior. In P. Zarembka (Ed.), *Frontiers in econometrics* (pp. 105–142). New York: Academic Press.
McFadden, D. (1976). Quantal choice analysis, a survey. *Annals of Economic and Social Measurement, 5*, 363–390.
Pötscher, B. M., & Prucha, I. R. (1991a). Basic structure of the asymptotic theory in dynamic nonlinear econometric models. I, consistency and approximation concepts. *Econometric Reviews, 10*, 125–216.
Pötscher, B. M., & Prucha, I. R. (1991b). Basic structure of the asymptotic theory in dynamic nonlinear econometric models. II, asymptotic normality. *Econometric Reviews, 10*, 253–325.
Samuelson, P. A., Koopmans, T. C., & Stone, J. R. (1954). Report of the evaluative committee for econometrica. *Econometrica, 22*, 141–146.
Schmidt, P. (1976). *Econometrics*. New York: Marcel Dekker.
Theil, H. (1953). *Estimation and simultaneous correlation in complete equation systems*. The Hague (mimeographed): Central Planning Bureau.
Zellner, A. (1971). *An introduction to bayesian inference in econometrics*. New York: Wiley.

Economic Order Quantity

The policy for a simple, deterministic inventory model that tells how much to order so that the sum of ordering and holding costs is minimized.

See

▶ Inventory Modeling
▶ Economic Order Quantity Model Extensions

Economic Order Quantity Model Extensions

Benjamin Lev
Drexel University, Philadelphia, PA, USA

The classical EOQ (Economic Order Quantity) model has a long list of assumptions. Begin by assuming that the horizon of the process is infinite and that all parameters stay the same over time. Then the solution to the classical EOQ problem is $\text{EOQ}(\infty) = Q = \sqrt{2AD/vr}$ where the parameters A, D, v, and r are the fixed cost of replenishment in \$/ order, (constant) annual demand in units/year, unit cost in \$/unit, and carrying charge in \$/unit/year, respectively. A typical picture of inventory level over time is displayed in Fig. 1.

A first possible extension to the classical model is to assume a finite horizon T; an example might be when a producer announces the discontinuation of an existing model at a future time T. The optimal solution is a series of n equal orders, each one of size DT/n, where n is the smallest integer such that

$$n(n+1) \geq \frac{Drv T^2}{2A}$$

(Schwartz 1972). The optimal solution is then either the integer

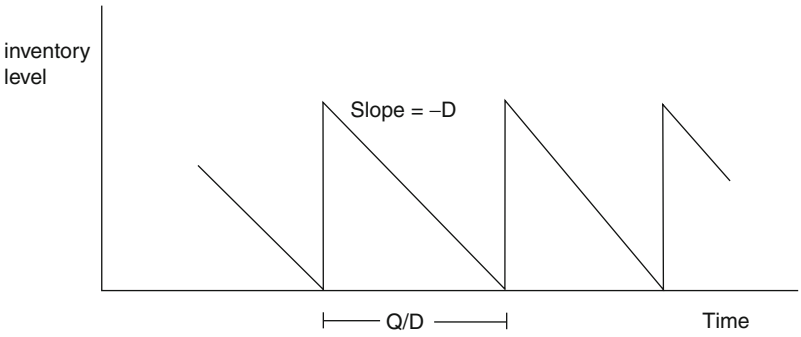
$$n = \left[T\sqrt{\frac{Drv}{2A}} \right]$$

or

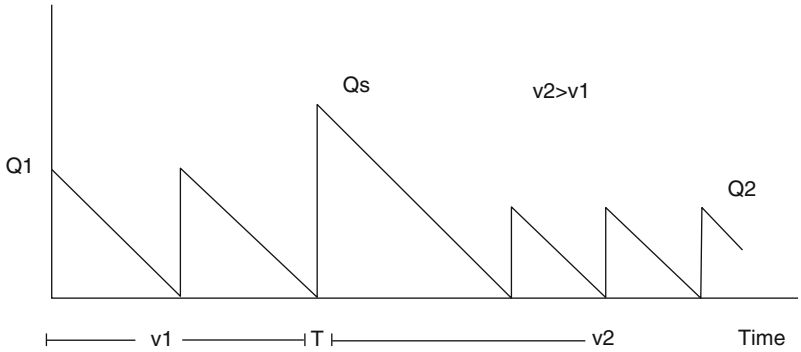
$$n = \left[T\sqrt{\frac{Drv}{2A}} \right] + 1.$$

Another extension is to relax the assumption that v is constant over time. An example is that the producer announces that in a future time T price will increase from v_1 to $v_2 (v_2 > v_1)$. The U.S. Postal Service increased the first class stamp rate on January 1999 from $v_1 = 32$ to $v_2 = 33$ cents (Fig. 2).

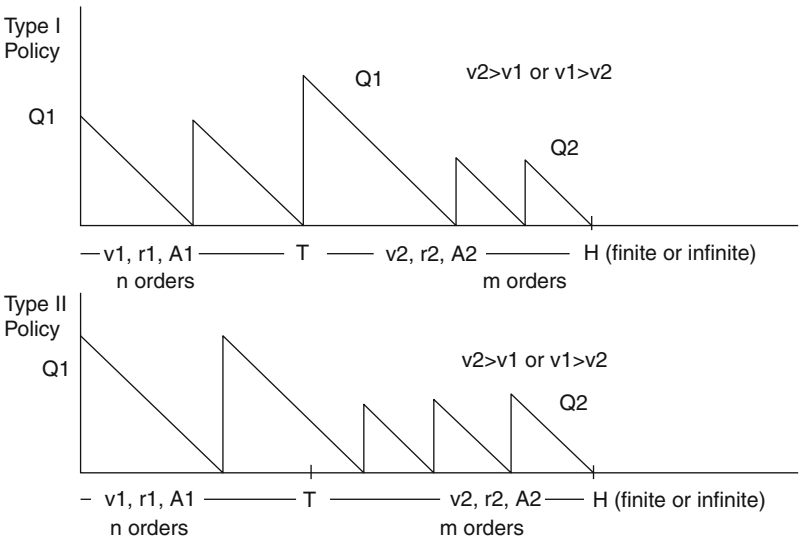
Naddor (1966) assumed that the inventory on hand is zero at the last opportunity to order at v_1 and suggested the solution of



Economic Order Quantity Model Extensions, Fig. 1 Classical EOQ model



Economic Order Quantity Model Extensions, Fig. 2 EOQ with price increases at time T



Economic Order Quantity Model Extensions, Fig. 3 EOQ with finite or infinite horizon H and parameters changing at time T

$$\text{EOQ}(\infty, v_1) = Q_1 = \sqrt{\frac{2AD}{v_1 r}}$$

(read this as the EOQ with infinite horizon and parameter v_1) followed by a single order size

$$Q_s = \frac{D(v_2 - v_1) + \sqrt{2AD/rv_2}}{rv_1}$$

followed by a series of order sizes

$$\mathrm{EOQ}(\infty, v_2) = Q_2 = \sqrt{\frac{2AD}{v_2 r}}.$$

This solution was later improved to include three possible solutions. The one suggested by Naddor where the inventory on hand is zero at time T, a solution when the inventory at time T is positive with either n orders during $(0, T)$, or $n + 1$ orders during $(0, T)$. A complete solution is presented in Lev and Soyster (1979).

The next extension is a combination of the previous two. Assume a finite horizon H, and at time T ($T < H$), some or all the parameters A, v, r might change (each one may increase or decrease, Fig. 3). There are two possible policies: Type I is used when the inventory on hand at time T is zero, and Type II is used when the inventory on hand at time T is positive. A procedure for finding n, Q_s, m for Type I and (n, m) for Type II is in Lev and Weiss (1990). See also Goyal (1992).

See

- ▶ Economic Order Quantity
- ▶ Inventory Modeling

References

Goyal, S. K. (1979). A note on the paper: An inventory model with finite horizon and price changes. *Operations Research, 30*, 839–840.

Goyal, S. K. (1992). A note on inventory models with cost increases. *Operations Research, 20*, 414–415.

Goyal, S. K., & Bhatt, S. K. (1988). A generalized lot size ordering policy for price increases. *Opsearch, 25*, 272–278.

Harris, F. (1913). How many parts to make at once? Factory. *The Magazine of Management, 10*(135–136), 152.

Lev, B., & Soyster, A. L. (1979). An inventory model with finite horizon and price changes. *Operational Research, 30*(1), 43–53.

Lev, B., Soyster, A. L., & Weiss, H. J. (1979). Comment on improved procedure for the finite horizon and price changes inventory models. *Operations Research, 30*, 840–842.

Lev, B., Weiss, H. J., & Soyster, A. L. (1981). Optimal ordering policies when anticipating parameter changes in EOQ systems. *Naval Research Logistics Quarterly, 28*, 267–279.

Lev, B., & Weiss, H. J. (1990). Inventory models with cost changes. *Operations Research, 38*, 53–63.

Naddor, E. (1966). *Inventory systems*. New York: John Wiley.

Schwartz, L. B. (1972). Economic order quantities for products with finite demand horizons. *American Institute of Industrial Engineers Transactions, 4*, 234–237.

Taylor, S. G., & Bradley, C. E. (1985). Optimal ordering strategies for announced price increases. *Operations Research, 33*, 312–325.

Economics and Operations Research

Frederic H. Murphy
Temple University, Philadelphia, PA, USA

Introduction

To understand the relationship between economics and operations research, one needs to understand some of the history of both fields. Operations research was developed prior to and during World War II with the pragmatic goal of improving military operations through the use of mathematics. The founders of the field of operations research came from diverse backgrounds, including physics, mathematics, engineering and economics. Operations research as a field has maintained its multidisciplinary character. Yet, the vast majority of the literature in the field has remained within the planning and operations areas of organizations and on algorithms to solve the models used for planning and operations. The textbooks on the subject contain a common set of subjects: stochastic modeling, simulation, optimization, inventory, and game theory.

Economics as a subject has been explored and developed for centuries. The field used to be called political economy, reflecting its public policy orientation, which carries through to today. The subject areas of economics can be defined broadly as follows: macroeconomics, the study of economic aggregates and the state of entire economies, and microeconomics, the study of economic agents, such as firms, and the market or organizational structures within which these agents operate to optimize their utility or profits. Examples of markets include monopolies, oligopolies, and perfect competition. Economists also develop tools such as the statistical techniques of econometrics, which are used for estimating the parameters of economic models. Despite strong connections between microeconomics and operations research, little overlap exists between macroeconomics and operations research.

While some early writers such as Adam Smith and Karl Marx were very influential as social philosophers, professional economists did not play an important role in day-to-day discussions of specific public policies and programs until well into the 20th century.

For example, the first tax cut based on macroeconomic theory and intended to stimulate the U.S. economy was implemented in the Eisenhower administration in the late 1950s, and regulatory reform based on microeconomic theory started to make headway in the 1970s, decades after the basics of the theory of workably competitive markets was understood. Economists drove many of the reforms in regulations around the world, most notably in the U.S., with Alfred Kahn leading the changes during the Carter administration. Economists played an even more central role in the restructuring of the British economy under Margaret Thatcher. The decisions of Competition Authority in the E.U. are based on modern economic theory and their work has led to new theory, such as understanding how electricity markets work. The European economies have also seen some restructuring reflecting the better understanding of markets. The remarkable rise of China is a result of the switch from a controlled economy to one that is more market based.

Economics developed contemporaneously with operations research and its broad impact in decision making is a post-World War II phenomenon as with operations research. The post-war development of economics and operations research was driven by the infusion of mathematics into areas beyond the hard sciences and engineering and the declining cost of computers and the greater reliance on analytics in making policy and operational decisions.

The Common History

Both fields apply mathematics to build and understand models that only approximate the reality being studied but improve decision making. Given this common starting point and the interest of operations researchers in finding the most economic solutions, the overlap in the fields has to be significant. The connections were most prominent in the early days of operations research and they involved such areas as optimization, inventory theory, and game theory.

Hitchcock (1941), a physicist, and Koopmans (1951), an economist, independently developed the first useful optimization model, the transportation model. Kantorovitch (1939), a mathematician in the Russian central planning agency, developed several linear programming (LP) models for production and distribution including the transshipment model. Stigler (1945), an economist, developed the diet/feed mix model. Dantzig (1951a and 1963), at the time, a mathematician in the US Air Force, invented the first generic linear programs and the simplex algorithm for solving them. The simplex algorithm has survived for 60 years as the primary method for solving linear programs. The collection of papers in Koopmans (1951) defined the beginning of the subjects of optimization, game theory, and the relationship between the two. It also devoted a substantial amount of space to generalizing the input–output model of an economy. Dantzig (1963) pointed to the work of Leontief in input–output models of the US economy as an important beginning for his ideas. The contributing authors to Koopmans' book were a mix of economists and mathematicians. Another important early book on linear programming, Dorfman et al. (1958), was written by economists. Indeed, Dorfman (1953) wrote the intuitive description of linear programming models that are found in all of the textbooks today. Current texts on microeconomics continue to include chapters on optimization and game theory.

Many of the first articles on optimization appeared in such journals as *Econometrica* (see the references in Dantzig 1963). Charnes and Cooper, the developers of many of the first linear programming models, also published in the economics journals (see, e.g., Charnes et al. 1952). Agricultural economists were quick to develop the feed-mix model for farmers. Mathematical programming has become a mainstay for agricultural economists (see Hazell 1986).

In the early days of inventory theory, the links between economists and operations researchers were equally strong. This area involved using such optimization techniques as dynamic programming and traditional, calculus-based methods to find optimal inventory policies (Arrow et al. 1958; Whitin 1957). However, the development of the field moved very quickly into the hands of operations researchers because the issues in inventory analysis evolved into the implementation of inventory systems and situation-specific models, away from the more broadly-based economic considerations.

Game theory was developed by von Neumann to study issues of conflict and cooperation at a theoretical level. He and Morgenstern applied game theory to economics (von Neumann and Morgenstern 1944).

The RAND Corporation became an early center for the development of game theory right after World War II, in good part to understand geopolitical and military strategy. The link between non-cooperative game theory and optimization was understood from very early on (see Dantzig 1951b; Gale et al. 1951). Game theory outside the military context plays a larger role in economics than it does in operations research, mainly because games are used for the qualitative analysis of interactions among players, which is appropriate for developing general policies, and few models are ever calibrated with numbers that reflect a specific situation.

The theory of auctions is an area of game theory with both qualitative and quantitative results and of interest to both economists and operations researchers. Auctions have been around for millennia, back to Babylonia. In one of the more notable auctions, Praetorian guards auctioned off the Roman Empire in A.D. 193, Shubik (1983). See Klemperer (1999) for a review of the literature. The economics literature focuses on single-item auctions whereas research on combinatorial auctions is in the operations research literature because of the emphasis on solution methods. Rothkopf et al. 1998 developed an approach to solve combinatorial auctions using integer programming. See de Vries and Vohra (2003) for a survey of combinatorial auctions.

The restructuring of electric utilities into a highly regulated market was driven by economics. However, the method for clearing the daily electricity markets is mainly linear programming, with integer programming coming in. Integer programming models for clearing the market take into account that there is a fixed cost to generating from the need to ramp power plants up and down. Some independent system operators such as PJM Interconnection are using integer programming. This has led to a literature on developing prices (see O'Neill et al. 2005, for one approach) using these models, as integer programs do not generate prices in the way linear programs do (see for example, Murphy et al. 2010).

Operations research techniques and operations researchers have contributed significantly to economics. Once Samuelson (1952) recognized the connection between mathematical programming and economic equilibrium models, mathematical programming became an important tool for economic analysis. In fact, the GAMS modeling language was developed by operations researchers at the World Bank for the purpose of solving computable general equilibrium models for evaluating national development plans (Brooke et al. 1993). An economist, Gustafson (1958), used dynamic programming, an operations research tool, to develop the first grain storage models to protect against famine. One of the most prominent microeconomic policy-analysis models of the 1970s, the Project Independence Evaluation System (PIES), was built by a team of operations researchers and economists led by William Hogan (1975), an operations researcher, who went on to organize the International Association of Energy Economists. Anyone working in electricity market restructuring has to be conversant in optimization, see Stoft (2002).

The Different Perspectives

Economics and operations research are distinct fields because the economists and operations researchers have different interests. Economists are primarily interested in qualitative analysis for policymaking and econometric modeling and estimation to understand the structure of an industry. Operations researchers are more interested in assisting decision making within the firm and have a strong computational orientation. For example, oil companies use the results of their mathematical programming models for operating their refineries and taking positions in forward markets. Even when economists are interested in numbers, they are looking to measure the impact of the sum of individual decisions rather than determining the decisions. This distinction between the fields is not absolute. Econometricians are interested in computational issues and the theoretical properties of their estimation methods. Scarf and Hansen (1973) has developed algorithms for computing economic equilibria. An emerging computationally intensive area of economics is agent-based modeling, which is a form of simulation where programmed agents act in their self interest and the simulations show how economies or ecologies evolve. The function of corporate planning and public policy studies produced by operations researchers is to provide insight rather than specific numbers. This is done through constructing multiple scenarios, examining alternative policies, and

analyzing the sensitivity of the results to the underlying parameters.

The different perspectives can be seen in the study of inventories. For the past few decades operations researchers and computer scientists have been implementing inventory systems, while the economists have been focusing on the effect of inventories on the business cycle rather than inventory policies per se and they gather data on inventories to measure aggregate inventory levels. The popularity of scientific inventory management in corporations and the desire to reduce inventories to free up capital and gain operational flexibility with just-in-time manufacturing has led to a significant decline in the inventory-to-sales ratio and rapid adjustments to inventories in response to changes in sales. That is, inventories turn over more quickly and companies are able to adapt to fluctuations in demand more rapidly with less draconian changes in production levels. The recession of 2008–2009 had a classic inventory reduction with the rebuilding of inventories contributing to the early recovery.

Inventory changes are known accelerators of business cycles. See Forrester (1961) for an illustration of this at the firm level. The smaller aggregate inventories are, relative to GNP, the less effect they have on business cycles. Economists measure this drop at the national level and factor this secular change into their macroeconomic models to explain the resultant dampening of business cycles. For example, the recession in the early 1990s was slow in coming and going but also shallow relative to past recessions because of the cumulative impact of individual improvements of inventory systems and production management. The 2000 recession did not have a strong inventory component to the decline or recovery, which made the recovery extremely sluggish, while recession of 2008–2009 is notable for the inventory movements that deepened the recession and provided an important component to growth in the first half of 2010. However, inventories were a secondary factor in that recession and not as important as they were in the recessions of the 1950's through the 1980's.

The different views of production functions taken by the two fields further illustrates the distinctions between the fields. When economists estimate production functions and are not building process models, they typically posit a differentiable functional form, gather data and estimate the parameters of the function using regression techniques. They do this to estimate output prices and understand the rates of substitution of inputs as a function of input prices. They are not looking inside the firm at the production process. Instead they are looking at market consequences. For example, from the rates of substitution, one can derive a demand curve for an input given the prices of the other inputs.

The operations research tool of data envelopment analysis (Charnes et al. 1978) estimates production functions using an alternative approach with different assumptions and goals. In data envelopment analysis the goal is to identify which decision-making units are efficient and which are not. That is, data envelopment analysis is a benchmarking tool for finding the best production practices with the ultimate goal of improving production processes. Unlike the econometric assumption that errors are in the data, data envelopment analysis assumes the data is error free and differences among decision-making units are due to different resource mixes and managerial effectiveness. The production function is the inputs and outputs of the decision-making units as activities in a linear program. A linear program is solved for each decision-making unit to see if it is on the efficient frontier. If it is, it represents best practices, given its mix of resources and products. If not, it is a candidate for improvement.

For every differentiator between operations research and economics, one can find an exception. Economists have focused on how agents interact in a framework and draw conclusions about the effect of changes in the framework, while operations researchers have been more interested in aiding the agents making the decisions. An exception to this is the study of traffic equilibria where the agents are travelers on a network of roads. The defining paper of this subject appeared in a civil engineering journal. Wardrop (1952) stated a set of equilibrium conditions based on trip times that are directly related to the equilibrium conditions for spatial economic equilibria based on cost. Although economists contributed to the early literature (e.g., Beckmann et al. 1956, established the relationship with economic equilibria), the bulk of the literature is in transportation journals with an operations research connection. See Nagurney (1993) for a description of different types of equilibrium models.

Agent-based modeling and OR-style simulation illustrate an area that has commonalities and differences between the economic and OR perspectives. In agent-based modeling one endows the agents with some knowledge of their environment and procedural rationality rather than a full understanding of how to optimize. The procedural rationality can be the steps of an optimization or search algorithm or a set of behavioral rules. See Epstein and Axtell (1996) and Tesfatsion and Judd (2006) for introductions to the subject. Usually, randomness in choices and/or outcomes is introduced. The agents interact in simulations and the equilibria or lack of equilibria are observed. This form of modeling is especially useful in exploring possible outcomes that can be later assessed using standard qualitative analysis methods or for observing markets that are too complicated for qualitative analysis. See Weidlich (2008) for a simulation of the German electricity market, a market that is too elaborate for standard qualitative analysis. The difference between agent-based modeling and the typical stochastic simulation built for operational purposes is that these simulations describe physical situations and look at material flows. When these simulations include people, the people are typically represented using probability distributions on some characteristic such as arrival patterns in a queueing simulation. Simulations can incorporate optimizations. However, the optimizations are for the system being studied and not the individual agents (see, for example, Andrews et al. 1996).

The Common Interests

The two fields overlap in several areas, including public policy analysis, finance, game theory, and decision analysis. There are others such as yield management that link both fields. The convergence of the fields in policy analysis comes about because politicians want quantitative analyses of programs. Economic models have a lot to say about how economic agents behave and operations researchers have the computational skills and modeling expertise to implement the economic theories and solve for the economic impacts of policy alternatives.

Examples here include the activity at the World Bank in building country and sector models. The close working relationships between economists and operations researchers have continued with the successor models to PIES, the Intermediate Future Forecasting System (Murphy et al. 1988) and the National Energy Modeling System (Energy Information Administration 2009). See Murphy and Shaw (1995) for a history of the energy models at the Energy Information Administration.

A key feature of these kinds of policy models is that in some sectors they model the decisions using optimization by representing the technology choices directly in the model. The main reason for using optimization is that the models need to have representations for policies and technologies that affect more than input and output prices and quantities and there is no history to assess the resulting decisions for some sectors. Other reasons include the need to link more than one sector and the existence of a convoluted data history that muddies the econometric analysis for estimating such things as a production function for electric utilities. The optimization models are usually simplified versions of the planning models used by the industry with coefficients based on industry aggregates and are process models as described by Manne and Markowitz (1961). They are treated as simulation models based on the result of Samuelson (1952) showing the connection between optimization and economic equilibrium models.

Markowitz (1954) proposed using process models almost 60 years ago as a way to model the whole economy, extending the input–output model to represent alternative production technologies. Henderson (1955) and Land (1956) successfully built models of coal markets using this approach. Only recently have databases, computers, and algorithms progressed to the point where these ideas can be realized for economy-wide models.

Policy models almost always include econometric components as well. For example, the above-mentioned energy models include econometrically estimated demand curves along with the process models for coal supply and transportation and electricity generation and transmission. In econometric models of production, one measures the inputs and outputs to statistically estimate the parameters of a production function. The model makes no statement about the actual decisions made by the agents. Instead, it models the outcomes of the decisions made by

the actors in the economic sector. Econometric approaches dominate optimization when there is too much heterogeneity among participants to specify the parameters of their decision environment, as in demand modeling or the behavior of producers when the industry has a large number of independent, small firms. In large-scale, capital-intensive industries, process models work well.

The finance literature is dominated by economic studies of financial markets and their efficiency. An example is the book *A Random Walk Down Wall Street* by Malkiel (1973). This book showed that movements in stock prices are a random walk, illustrating why stock pickers in general cannot beat the market. Also, Tobin's (1958) results on the relationship between risk and return were key to the development of decision models in finance. Equilibrium conditions lead to an efficient frontier in portfolio theory that simplifies the choices to a tradeoff between an index of risky assets and a risk-free asset, typically, government bonds. See Bodie et al. (2005) for an introduction to the subject. For a book that covers finance from an operations research perspective see Luenberger (1998).

Financial markets are not entirely efficient and the Black and Scholes (1973) model for pricing options created a whole new segment of the finance industry. Its basis is dynamic programming. The book by the economists Dixit and Pindyck (1994) emphasized the role of dynamic programming in properly valuing investments with uncertain returns.

Optimization models have come to play an important role in determining the mix of assets in a portfolio, the first one being the model by Markowitz (1952), which represents the beginning of computational finance. That model is a quadratic program that trades off expected return and portfolio variance. Because of the ability to solve far larger linear programs than in the past, stochastic programming models for building portfolios have made an important mark in the industry. For example, see Carino et al. (1994) for a description of the kind of operations research models used by the people known as "rocket scientists" in the financial press. Because of the difficulty of determining probability distributions and finding stable correlations among assets, an alternative approach known as robust optimization is gaining traction. Here random variables are captured in uncertainty sets. The model optimizes an objective function subject to satisfying all constraints for all parameter values in the uncertainty sets. In some sense this is a maximin optimization. See Bertsimas and Thiele (2006) for a tutorial on the subject.

Portfolio theory also uses the concepts of value at risk and conditional value at risk. Say a portfolio manager wants to limit the worst case outcomes by choosing a portfolio that maximizes the value of a cutoff point (minimizing the loss) on the left tail of a probability distribution where the left tale covers 5% of the possible outcomes. This is the value-at-risk optimization. Conditional value at risk minimizes the expected value of the losses below the value-at-risk loss. For readers with an inventory background, value at risk corresponds to setting reorder points and safety stocks at a level that targets a given probability of stock out. Conditional value at risk is the equivalent of setting reorder points at a target number of expected units short during lead time. There are convexity issues with value at risk. Rockafellar and Uryasev (2000) show how to formulate the conditional value at risk optimization as a linear program.

The interconnection between economics and operations research in game theory can be illustrated by the Averch-Johnson hypothesis (1962). This hypothesis states that regulated firms have a bias to overinvesting in capital rather than labor. They demonstrated their results by evaluating the Kuhn-Tucker conditions of an optimization model, where a firm with a monopoly maximizes profits subject to a rate-of-return constraint. With the deregulation of many industries, it is now known that these firms were not only overcapitalized, but they also had too much labor, in violation of the Averch-Johnson hypothesis.

The problem with the Averch-Johnson model was it presumed that the firm was a single entity and could optimize its behavior. However, one must not treat the firm as the atom. Instead, one must look at how the agents within the firm interact and look further into the nature of the behaviors of the agents who make up the firm. Figuring out the underlying incentives of the members of a firm and analyzing their behavior relative to the interests of stockholders is known as principal agent theory, an important area of microeconomics. For example, one could explain the behavior of regulated firms as follows: managers increase their importance by increasing the number of employees under them and buying labor peace by

paying high wages to unionized employees. Given the extent to which electric utilities were perceived as highly inefficient in the Averch-Johnson sense, it is interesting to see the differences in performance of investor-owned generators that were restructured, those that were not, and municipal utilities. Fabrizio et al. (2007) find only modest improvement.

Studying the behavior of economic agents and other individuals has a long tradition in economics and is the essence of game theory. Since little data exists for numerically evaluating game models, almost all studies examine the qualitative properties of the resulting games. Economists have focused mostly on markets (Shubik 1959). Indeed, outside of von Neumann's early work on parlor games, the book by von Neumann and Morgenstern (1944) was the first major treatment of the subject and focused on economics. Operations researchers have studied other types of games such as war games and invented some of what are now the classics like the prisoner's dilemma game (Poundstone 1992). The center for this work was RAND. An example of a strategic game that was studied was the stability of mutual assured destruction as a defense against nuclear war. Schelling (1980) presented an analysis of these strategic games. Both groups study the generic properties of games that abstract common situations. Shubik presented an interesting example of someone who does both strategic and economic games. As part of his examination of strategic issues, he used the dollar auction game to describe games of escalation such as war and lawsuits (Poundstone 1992).

Part of the reason for the common interest of economists and operations researchers in game theory is its universality in understanding conflict and cooperation. Political scientists and sociologists have become involved in game theory for the same reason. The link between political science and games is direct through the games already mentioned and the use of game theory concepts in negotiation. Sociologists use games to understand social interactions. The prisoner's dilemma game has been used repeatedly to explain the behavior of individuals in social situations and social structures. Thus, the notions of game theory have moved beyond the disciplines in which they were developed and influence important areas of the social sciences.

Rational decision making encompasses game theory. Indeed, a still invaluable work on the subject that treated both together is the book by Luce and Raiffa (1957). What is a rational decision is subject to debate. To explore the subject, von Neumann and Morgenstern developed the concept of expected utility. Utility is a simple concept in many situations when the goal can be clearly stated as with maximizing profits. However, in real life an individual faces many trade-offs. Examples include the willingness to bear risk, how to value income versus leisure, what value is in the products consumed, and how to value the future over the present. In the decision-making literature, Keeney and Raiffa (1976) explored the issues associated with multi-attribute utility in decision making. The notion of multi-attribute utility is central to the study of negotiation (Raiffa 1982), as the different parties in a negotiation generally value different aspects of the subject under discussion, creating an opportunity for joint gain.

As in other areas, economists have not focused on making actual decisions except in the general properties that can be understood from the decision-making process, as in Arrow (1951). In his seminal work on social choice, Arrow posits a set of axioms that define rationality and then shows how group interactions and voting processes lead to irrational decisions even though the original actors have rational utility functions. Another example of this is the economics literature on rational expectations. In its most basic form, the question addressed in the context of macroeconomic models is: "How do the consequences of macroeconomic policy change when the participants in the economy have rational expectations about the effect of macroeconomic policies and adjust their decisions?" See Redman (1992) or Sargent (1993) for a discussion of this area of economics.

In much economics theory the literature presumes the agents have full information and know their utility functions. These are questionable assumptions and one should treat the theorems of economics as hypotheses rather than foregone conclusions. Econometricians historically have done the testing. Smith (1994, 2005) developed the field of experimental economics by laboratory studies in which the subjects were rewarded based on their performance in economic situations. He verified results such as the effect of risk aversion on auction designs. This kind of work does not appear in the operations research literature because operations research is prescriptive rather than descriptive.

Operations research has come to dominate the subject so far as making actual decisions. For an example of a detailed decision analysis in a corporation see Borrison (1995). Some of the most important literature has come from psychologists trying to understand peoples' thought processes. The psychology literature is aimed squarely at the rational actor hypothesis of economics and finds it wanting. For a book that examines the approaches of all three disciplines see Bell et al. (1988). For a description of experiments that bring out cleverly human thought processes, see Ariely (2008). An area known as happiness research is moving economics from standard utility theory. Researchers do surveys to understand what makes people feel better. For example, rich people are happier than poor people. However, as a society gets richer, happiness does not increase. This has implications for the kinds of measures that should be used to determine the wellbeing of a society. See Graham (2005) for an overview. Again, a key difference between economics and operations research is that the goal of economics is to be descriptive, a science that discovers what is, and operations research, which comes from the engineering tradition of achieving organization-specific goals such as improved supply chains, looks to engineer better decisions.

An operations researcher cannot be an effective modeler and analyst without a good understanding of the basics of economic theory. Here is a simple example. The simplex algorithm represents the behavior of a set of independent economic agents (activities) making decisions to act or not based on a set of incentives in the form of objective coefficients and resource prices (dual variables). And if they choose to act based on profitability (reduced costs), they proceed until they reach a resource limit or drive a competitor out of business (the replaced activity reaches zero in the simplex pivot). By looking at a solution from the perspective of the simplex algorithm as an economic process, one gains the deeper insights into the story contained in the solution that takes analyst beyond just the value of the objective function and the level of activities.

The basic notions of substitutes and complements in production processes determines the character of optimization models. Network linear programs are models of pure substitution. Whereas, a product-mix model consists of activities that have inputs that are pure complements. The vast majority of the constraints in an LP can be classified as supply, demand or material-balance constraints. Greenberg (1981) used the notions of substitutes and complements to gain deeper insights into linear- programming models and their solutions.

Concluding Remarks

Economics and operations research have common roots. The fields often use the same tools, such as the Karush-Kuhn-Tucker conditions. In economics these conditions are used for pricing, marginal analysis, as with the search for institutional distortions of the marketplace in the Averch-Johnson hypothesis, and for such uses as the derivation of cost functions from production functions. Operations researchers exploit these conditions to improve algorithms and use the actual duals and ranges for evaluating the stability of the model results, estimating the effects of uncertainty in the coefficients on the solution, and determining the costs of constraints with an eye towards adding or reducing resources when making decisions within the firm.

Typically, the fields use these tools differently for different purposes. This reflects the different professional goals of the individuals involved in these fields. Operations researchers focus on making specific decisions and economists study the consequences of different organizational and market structures and policies through an assumption of rational decision making. Both groups are interested in understanding rational decision making and the consequences of rational decisions. This can be seen in the different views of the firm. The traditional economic theory of the firm is really a theory of the interactions of firms or constituencies within the firm. Whereas, operations research models provide a theory of decision making within the firm and are an important component of a theory of the internals of the firm. The operations research models do not provide a complete theory of decision making in the firm because operations researchers, although commenting on conflicts in the firm, tend to not focus on the incentives and structures that create these conflicts. This is where agency theory fits in and one of the places where game theory links both fields. The result is that operations research models tend to be most successful in capital intensive firms where the issues are managing those assets rather than large numbers of people.

The fields are now distinct because operations research takes an engineering perspective: the goal is to invent improved ways for making decisions, and in the process of doing this, inventing new models and algorithms as needed. In fact, the very success of operations research in building and using large computer-based models for planning and operations has led to the complaints of Ackoff (1987) that the field has abdicated its role in corporate strategy and public policy. Economics, instead, is a social science where the goal is to understand the existing world and study the consequences of policies that affect this world using the basic theme of exploring the consequences of rational self-interest. The two fields come together when there is the need to change the rules of the marketplace or when the marketplace creates opportunities to engineer new products that provide a profit as in finance and electricity markets. Both fields have their distinct niches, yet will always be connected by their tools and history. Each will continue to enhance the other field, an example of the economic notion of joint gain through comparative advantage.

See

- ▶ Banking
- ▶ Corporate Strategy
- ▶ Data Envelopment Analysis
- ▶ Decision Analysis
- ▶ Econometrics
- ▶ Game Theory
- ▶ Input–Output Analysis
- ▶ Karush-Kuhn-Tucker (KKT) Conditions
- ▶ Linear Programming
- ▶ Portfolio Theory: Mean-Variance Model
- ▶ Public Policy Analysis
- ▶ Quadratic Programming
- ▶ RAND Corporation
- ▶ Utility Theory

References

Ackoff, R. (1987). Presidents' symposium: OR, a post mortem. *Operations Research, 35*(3), 471–474.

Andrews, S., Wang, X., Murphy, F. H., & Welch, S. (1996). Modeling crude oil lightering in delaware bay. *Interfaces, 26*(6), 68–81.

Ariely, D. (2008). *Predictably irrational: The hidden forces that shape our decisions*. New York: Harper Collins.

Arrow, K. (1951). *Social choice and individual values*. New York: Wiley.

Arrow, K., Karlin, S., & Scarf, H. (1958). *Studies in the mathematical theory of inventory and production*. Stanford, CA: Stanford University Press.

Averch, H., & Johnson, L. (1962). Behavior of the firm under regulatory constraint. *American Economic Review, 52*, 369–372.

Beckmann, M., McGuire, C. B., & Winsten, C. B. (1956). *Studies in the economics of transportation*. New Haven, CT: Yale University Press.

Bell, D., Raiffa, H., & Tversky, A. (1988). *Decisionmaking, descriptive, normative and prescriptive interactions*. Cambridge: Cambridge University press.

Bertsimas, D., & Thiele, A. (2006). Robust and data-driven optimization: Modern decision making under uncertainty. In M. P. Johnson, B. Norman, & N. Secomandi (Eds.), *Tutorials in operations research*. Hanover, MD: INFORMS.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, 637–659.

Bodie, Z., Kane, A., & Marcus, A. J. (2005). *Investments*. New York: McGraw-Hill Irwin.

Borrison, A. (1995). Oglethorpe power corporation decides about investing in a major transmission system. *Interfaces, 25*(2), 25–36.

Brooke, A., Kendrick, D., & Meeraus, A. (1993). *GAMS: A user's guide*. Redwood City, CA: Scientific Press.

Carino, D., Kent, T., Myers, D., Stacy, C., Sylvanus, M., Turner, A., Watanabe, K., & Ziemba, W. (1994). The Russell-Yasuda Kasai model: An asset liability model for a Japanese insurance company using multi-stage stochastic programming. *Interfaces, 24*(1), 29–49.

Charnes, A., Cooper, W. W., & Mellon, B. (1952, April). Blending aviation gasolines—a study in programming interdependent activities in an integrated oil company. *Econometrica, 20*(2).

Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operations Research, 2*, 429–444.

Dantzig, G. (1951a). Maximization of a linear function of variables subject to linear inequalities. In T. C. Koopmans (Ed.), *Activity analysis of production and allocation*. New York: Wiley.

Dantzig, G. (1951b). A proof of the equivalence of the programming problem and the game problem. In T. C. Koopmans (Ed.), *Activity analysis of production and allocation*. New York: Wiley.

Dantzig, G. (1963). *Linear programming and extensions*. Princeton, NJ: Princeton University Press.

de Vries, S., & Vohra, R. (2003). Combinatorial auctions: A survey. *INFORMS Journal on Computing, 15*(3), 284–309.

Dixit, A. K., & Pindyck, R. S. (1994). *Investment under uncertainty*. Princeton, NJ: Princeton University Press.

Dorfman, R. (1953). Mathematical or 'linear', programming: A nonmathematical exposition. *American Economic Review, 43*, 797–825.

Dorfman, R., Samuelson, P., & Solow, R. (1958). *Linear programming and economic analysis*. McGraw Hill, New York.

Energy Information Administration. (2009). *The national energy modeling system: An overview*. www.eia.gov/oiaf/aeo/overview/.

Epstein, J. M., & Axtell, R. (1996). *Growing artificial societies*. Brookings and MIT Press.

Fabrizio, K., Rose, N. L., & Wolfram, C. A. (2007). Do markets reduce costs? Assessing the impact of regulatory restructuring on US electric generation efficiency. *American Economic Review, 97*(4), 1250–1277.

Forrester, J. W. (1961). *Industrial dynamics*. MA/New York: MIT Press/Wiley.

Gale, D., Kuhn, H., & Tucker, A. (1951). Linear programming and the theory of games. In T. C. Koopmans (Ed.), *Activity analysis of production and allocation*. New York: Wiley.

Graham, C. (2005). The economics of happiness insights on globalization from a novel approach. *World Economics*, *6*(3), 41–55.

Greenberg, H. J. (1981). Measuring complementarity and qualitative determinacy. In H. J Greenberg & J. S. Maybee (Eds.), *Computer-assisted analysis and model simplification* (pp. 497–522). Academic Press.

Gustafson, R. L. (1958). Carryover levels for grains. *US Dept. of Agriculture Technical Bulletin*, p. 1178.

Hazell, P. B. R. (1986). *Mathematical programming for economic analysis in agriculture*. New York: Macmillan.

Henderson, J. M. (1955, November). A short-run model of the coal industry. *The Review of Economics and Statistics, 37*.

Herman, R. (1992). Technology, human interaction, and complexity: Reflections on vehicular traffic science. *Operations Research, 40*, 199–211.

Hitchcock, F. (1941). The distribution of a product from several sources to numerous localities. *Journal of Mathematical Physics, 20*, 224–230.

Hogan, W. W. (1975). Energy policy models for project independence. *Computers and Operations Research, 2*, 251–271.

Kantorovitch, L. (1939). *Mathematical methods in the organization and planning of production*, Leningrad State University. Translated in *Management Science*, *6*(1960), 366–422.

Keeney, R., & Raiffa, H. (1976). *Decisions with multiple objectives*. New York: Wiley. Reprinted in 1993 by Cambridge University Press, New York.

Klemperer, P. (1999). Auction theory: A guide to the literature. *Journal of Economic Surveys, 13*(3), 227–286.

Koopmans, T. (1951). *Activity analysis of production and allocation*. New York: Wiley.

Land, A. (1956). A problem in transportation. *Journal of the Operations Research Society of America, 4*(1), 132–133.

Luce, D., & Raiffa, H. (1957). *Games and decisions*. New York: Wiley.

Luenberger, D. (1998). *Investment science*. Oxford: Oxford University Press.

Malkiel, B. (1973). *A random walk down Wall Street*. New York: W.W. Norton.

Manne, A. S., & Markowitz, H. M. (1961). Studies in process analysis: Economy-wide production capabilities. *Proceedings of a conference sponsored by the Cowles Foundation*, April 24–26, Yale University.

Markowitz, H. M. (1952, March). Portfolio selection. *Journal of Finance*, *7*(1).

Markowitz, H. M. (1954). Industry-wide, multi-industry, and economy-wide process analysis. In T. Barna (Ed.), *The structural interdependence of the economy, proceedings of an international conference on input–output analysis* (pp. 121–150).

Murphy, F. H., & Shaw, S. H. (1995). The evolution of energy modeling at the federal energy administration and the energy information administration. *Interfaces, 25*(5), 173–193.

Murphy, F. H., Conti, J., Sanders, R., & Shaw, S. (1988). Modeling and forecasting energy markets with the intermediate future forecasting system. *Operations Research, 36*, 406–420.

Murphy, F. H., Mudrageda, M., Soyster, A., Saric, A., & Stankovic, A. (2010). The effect of contingency analysis on nodal prices and the day-ahead market. *Energy Policy, 38*(1), 141–150.

Nagurney, A. (1993). *Network economics, a variational inequality approach*. Dordrecht: Kluwer.

O'Neill, R. P., Sotkiewicz, P. M., Hobbs, B. F., Rothkopf, M. H., & Stewart, Jr W. R. (2005, July). *European Journal of Operational Research*, *164*(1), 269–285.

Poundstone, W. (1992). *Prisoner's dilemma*. New York: Double-day.

Raiffa, H. (1982). *The art and science of negotiation*. Cambridge: Harvard University Press.

Redman, D. A. (1992). *A reader's guide to rational expectations*. Hants, England: Edward Elgar.

Rockafellar, R. T., & Uryasev, S. (2000). Optimization of conditional value at risk. *Journal of Risk, 2*, 21–41.

Rothkopf, M. H., Pekec, A., & Harstad, R. M. (1998). Computationally manageable combinatorial auctions. *Management Science, 44*(8), 1131–1147.

Samuelson, P. A. (1952). Spatial price equilibrium and linear programming. *American Economic Review, 42*, 283–303.

Sargent, T. J. (1993). *Rational expectations and inflation*. New York: Harper Collins.

Scarf, H., & Hansen, T. (1973). *The computation of economic equilibria*. New Haven, CT: Yale University Press.

Schelling, T. (1980). *The strategy of conflict*. Cambridge: Harvard University Press.

Shubik, M. (1959). *Strategy and market structure*. New York: Wiley.

Shubik, M. (1983). Auctions, bidding, and markets: An historical sketch. In R. Engelbrecht-Wiggans, M. Shubik, & J. Stark (Eds.), *Auctions, bidding, and contracting* (pp. 33–52). New York University Press.

Smith, V. L. (1994). Economics in the laboratory. *The Journal of Economic Perspectives, 8*(1), 113–131.

Smith, V. L. (2005). Behavioral economics research and the foundations of economics. *Journal of Socio-Economics, 34*(2), 135–150.

Stigler, G. (1945). The cost of subsistence. *Journal of Farm Economics, 27*(2), 303–314.

Stoft, S. (2002). *Power system economics: Designing markets for electricity*. IEEE Press.

Tesfatsion, L., & Judd, K. L. (Eds.). (2006). *Handbook of computational economics*. Elsevier.

Tobin, J. (1958). Liquidity preference as behavior toward risk. *The Review of Economic Studies*, 65–86.

von Neumann, J., & Morgenstern, O. (1944). *Theory of games and economic behavior*. Princeton, NJ: Princeton University Press.

Wardrop, J. G. (1952). Some theoretical aspects of road traffic research. *Proceedings Institute of Civil Engineers, Part II*, 325–378.

Weidlich, A. (2008). *Engineering interrelated electricity markets: An agent-based approach*. Heidelberg: Physical-Verlag.
Whitin, T. (1957). *The theory of inventory management* (2nd ed.). Princeton, NJ: Princeton University Press.

Edge

(1) An edge is the line segment joining two extreme points of a polyhedron such that no point on the segment is the midpoint of two other points of the polyhedron not on the segment. (2) A line connecting two nodes in a graph (network).

See

- ▶ Graph Theory
- ▶ Network Optimization
- ▶ Polyhedron

Efficiency

In statistics, an unbiased estimator's efficiency is the relative size of its variance compared to other unbiased estimators.

See

- ▶ Data Envelopment Analysis
- ▶ Efficient Solution

Efficiency Frontier

- ▶ Data Envelopment Analysis

Efficient Algorithm

- ▶ Computational Complexity

Efficient Point

- ▶ Efficient Solution
- ▶ Multiobjective Programming

Efficient Solution

For a maximizing multi-objective problem, a solution $\boldsymbol{x}^0$ is efficient if $\boldsymbol{x}^0$ is feasible and there exists no other feasible solution $\boldsymbol{x}$ such that $\boldsymbol{cx} \geq \boldsymbol{cx}^0$ and $\boldsymbol{cx} \neq \boldsymbol{cx}^0$. An alternative definition is that a feasible $\boldsymbol{x}^0$ is efficient if and only if there exists no other feasible $\boldsymbol{x}$ such that $\boldsymbol{c_k x} \geq \boldsymbol{c_k x}^0$ and $\boldsymbol{c_k x} > \boldsymbol{c_k x}^0$ for at least one $\boldsymbol{k}$. An efficient solution is a feasible solution for which an increase in value of one objective can be achieved only at the expense of a decrease in value of at least one other objective. Efficient solutions are also called nondominated solutions or Pareto-optimal solutions.

See

- ▶ Multiobjective Linear-Programming Problem
- ▶ Multiobjective Programming
- ▶ Pareto-optimal Solution

Eigenvalue

- ▶ Analytic Hierarchy Process
- ▶ Matrices and Matrix Algebra

Eigenvector

- ▶ Analytic Hierarchy Process
- ▶ Matrices and Matrix Algebra

Electric Power Systems

Oliver S. Yu
Star Strategy Group, Los Altos Hills, CA, USA

Introduction

An electric power system is designed for reliable, economic, and socially acceptable production and delivery of electricity to individual customers. It involves many interrelated elements: generation stations, control centers, transmission lines, distribution substations, and distribution feeders. With an intricate system structure, complex economic and social objectives, and numerous reliability, safety, and resource constraints, power system planning and operation has long been an ideal field for the development and application of operations research and management science (OR/MS) techniques. These developments and applications continue to expand and evolve with the advancements in power technologies and changes in the utility industry. In the following sections, electric power generation system planning and operation is first used as a prototype example to present some basic concepts. This is then followed with a discussion of trends and challenges in the electricity industry to provide a glimpse of the enormous opportunity for OR/MS applications to power systems.

Overview

Electric power systems have been planned, constructed, and operated to supply electricity to the general public by regulated utilities. As regulated entities, these utilities are allowed to recover their capital investments and operating costs for supplying electricity, with an allowance for reasonable returns by collecting revenues from customers in the form of electric rates. To assure the economic efficiency of the utilities, regulatory commissions in general have required utilities to minimize their total revenues required for electricity supply. Therefore, electric power system planning and operation has been a classical OR/MS problem of minimizing utility revenue requirements to meet projected electric demand growth over a future time period at a given level of reliability.

An electric power systems have three major parts: generation, transmission, and distribution. Because power flows in transmission and distribution are still difficult to be estimated accurately and economically, most applications of optimization techniques have been in generation system planning and operation. Specifically, the generation system planning and operation problem is to select a combination of power plants and unit dispatch schedules to minimize the present worth of the total capital, fuel, and operations and maintenance expenditures for meeting future electric demand while satisfying generally agreed-upon generation system reliability standards.

Optimal Generation System Reliability

Setting generation system reliability standards is itself an optimization problem, because too low a standard would cause economic losses to the customers from frequent electric supply interruptions, while too high a standard would cause low capacity utilization of power plants and thus high electricity costs.

In the past, a commonly accepted empirical generation system reliability standard has been the one day in 10 year loss of load probability, that is, the daily electric peak load not to exceed available generating capacity of each day by more than one day in 10 years. However, with increasing technical capability, computationally efficient procedures have been developed to enable utilities to assess generation system reliability in detail (Yu 1978). Further-more, cost/benefit approaches have been used to derive the optimal reliability standard for a given socio-economic environment by determining the appropriate tradeoffs between the cost of power supply shortage and disruption and the cost of over-capacity to the customers (Kaufman 1975; Telson 1975; Keane and Woo 1992).

Optimal Dispatch of Generating Units

Because electric load varies by hour, day, and season in a year, the dispatching of generating plants to meet daily load requirements is also itself an optimization

problem. This so-called production costing problem strives to determine the plant dispatch schedule that will minimize the fuel as well as operations and maintenance costs for meeting the load. In a broader context, the production costing problem also involves power purchases from neighboring utilities for either low cost or backup capacity. Therefore, for optimal generation system planning, a solution to the operational sub-problem of production cost optimization must first be found.

A classical generation system operation optimization problem is the combined scheduling of hydroelectric and thermal power plants. Specifically, ineffective use of hydropower will increase the use of high operating cost thermal plants. The power system operator's problem is, therefore, to minimize the total cost of generation system operations for a given time period with uncertainties in load requirement and water availability.

Generation System Expansion Planning

In a simplified form, the generation system expansion planning problem for a time horizon [0, T], may be expressed as follows (Anderson 1972):

$$\text{Minimize} \quad c_f(x_1, x_2, \ldots, x_n) + c_v[y_1(t),\ y_2(t),\ \ldots,\ y_n(t)]$$

$$\text{such that} \quad \sum_i y_i(t) \geq L(t) \quad \text{for } t \text{ in } [0, T]$$

$$0 \leq y_i(t) \leq d_i x_i$$

x_i	is the capacity of plant i;
$y_i(t)$	is the capacity of plant i used at time t;
$L(t)$	is the load at time t;
d_i	is the derating of plant i because of random forced outages;
c_f	is the present worth of the fixed costs; and
c_v	is the present worth of the variable costs.

In a more sophisticated formulation, the random nature of plant outage is taken into account by replacing the first constraint with:

$$\Pr\left\{\sum_i y_i(t) < L(t)\right\} \leq p \quad \text{for } t \text{ in } [0, T]$$

Probabilistic simulation models have been used to determine the optimal dispatch schedule accurately (Stremel et al. 1980; Sidenblad and Lee 1981). With the solution of the production costing subproblem, a number of OR/MS techniques, including linear and nonlinear programming, can be used to solve the overall generation system planning problem.

Another level of sophistication is to require x_i to be integer-valued. In this case, either mixed integer programming (Benders 1962) or dynamic programming (Jenkins and Joy 1974) can be used for generation system planning solutions.

A comprehensive application of OR/MS and other engineering-economic analysis techniques to generation system expansion planning has been the Electric Generation Expansion Analysis System (EGEAS) developed by the Electric Power Research Institute (EPRI 1983).

Optimal Maintenance Scheduling of Generating Units

Another optimization problem in generation system operations is unit maintenance scheduling. Each generating unit has a set period each year for preventive maintenance. The objective of optimal maintenance scheduling is to minimize the overall production cost while meeting the generation system reliability standards throughout the year. The problem is somewhat similar to the knapsack problem in OR/MS. Because utility business requirements vary, often a heuristic approach is required to find a solution for maintenance scheduling for a specific power system (Yu and Freddo 1978).

Fuel Inventory Planning

One other area in generation system planning amenable to OR/MS application is fuel inventory planning. Chao et al. (1989) have developed an optimization computer program that performs formal cost-benefit analysis of the following problems:

- uncertain fuel deliveries and fuel burn;
- seasonality in fuel use and fuel supply;
- supply disruption of varying severity, warning times, and duration; and
- nonlinear shortage costs.

Utility Resource Planning

Over the past years, there have been major changes in the electric utility industry in the United States. As a result, utility resource planning objectives have also evolved (Yu and Chao 1989).

In the 1970s, growing environmentalism imposed an additional tradeoff between environmental control cost and generation system reliability. A major application of OR/MS techniques was the development of the Over/Under Capacity Expansion Model funded by the Electric Power Research Institute (EPRI 1987).

In the 1980s, prevailing energy conservation ethics has given rise to the widespread adoption of the Least Cost Planning concept, also referred to as Integrated Resource Planning. Under this planning concept, in addition to an economic comparison among themselves, electric supply alternatives are to be further compared with demand-side management options, that include energy conservation and load management. As the research management arm of the U.S. electric utility industry, EPRI has also funded the development of a number of major optimization tools in this area (EPRI 1988), including the Multi-objective Integrated Decision Analysis and Simulation (MIDAS) model and the Utility Planning Model (UPM).

Trends and Challenges

There has been a fundamental changing trend in the electricity industry worldwide. This trend is largely driven by the ideological popularity of market economy and the development of low-cost high-efficiency gas turbine combined cycle generation technology. In this changing trend, the traditionally vertically integrated electricity industry is disintegrated and restructured so that electricity generation and retailing become competitive businesses while transmission and distribution systems remain regional monopolies.

In this restructured electricity industry, the applications of OR/MS tools become more important than ever to effectively manage the complexity and uncertainty in the competitive business environment and ensure the fairness of the market and the profitability of the investors. The following are a few major examples of these applications.

Transmission System Congestion Charge Allocation

In the restructured industry, a transmission system becomes a common carrier for all generating units. In addition to transmission charges, these units also need to pay a charge for the right to ship power under congestion. How to accurately estimate and properly allocate the congestion charge is a complex task that requires a systematic and quantitative approach. One such an approach that has received increased acceptance has been proposed by Chao and Peck (1996). Their idea is to allocate a fixed and finite set of transmission capacity rights to electricity suppliers according to a trading rule for the short-term leasing of these rights. The holders will set a price that maximizes its profit, while the rights are assigned to market participants that value them the most highly. As a result, this approach is efficient and involves a series of optimization analysis.

Game Theory Applications in Generation Competition

In the competitive generation market, there are many opportunities for game theory applications. In this market, a common bidding rule is the so-called second price bidding rule. By this rule, electricity suppliers bid on an hourly basis the amounts and the prices they would be willing to supply power. These prices are staggered from low to high together with their respective amount to form a electricity supply curve of that hour. At the same time, the power purchasers would bid the amounts and prices they would be willing to pay at the given hour. Their prices are staggered from high to low together with their respective amounts to form the electricity demand curve for the hour. The price at which supply and demand curves meet is the market clearing price, that will be paid to all suppliers and paid by all purchasers. With game theory, market participants can develop bidding strategies by anticipating the actions and reactions of the competitors. On the other hand, regulators who are responsible for the fairness and effectiveness of the market can also use game theory to detect potential dominance of the market by a small number of participants and possible collusive behavior among

participants. Several technical papers about these applications have collected by the Institute for Electrical and Electronics Engineers (IEEE 1999).

Risk Management through Option Theory

In the competitive market, investments in generation plants entail unprecedented risks, because they are no longer protected by economic regulation for guaranteed returns. Similarly, there are considerable risks as well as profitability in the retailing of electricity based on a combination of long-term contract and short-term market trading. As a result, both investors and retailers need to effectively manage their risks while maximize profitability. In addition to the usual decision and risk analysis techniques, a new approach for these investment and portfolio strategies, called real option has been developed through the extension of option theory from the financial field. In this approach, to provide flexibility in a highly uncertainty business environment, investors will make small initial investments to secure the rights for future large-scale investment options and retailers will negotiate option contracts to manage uncertainty and hedge against risks. This area is rapidly expanding, see Trigeorgis (1996) for an introduction and review. In addition, Smith and Nau (1995) provides an insightful comparison between option pricing theory and decision analysis.

The Smart Grid

Electric power system has always been information intensive. Because electricity generally cannot be stored, the operation center of a power system must continuously monitor changes in electricity demand and matches it by increasing or decreasing power supply in accordance with the merit order of the marginal cost of generation. In the meantime, the center also needs to constantly maintain the reliability of the system by stabilizing the system voltage and frequency and re-routing electricity in response to breakdowns and outages in the system. These functions require large-scale information acquisition, data analysis, and system control functions, which are the basis of the Supervisory Control and Data Acquisition/Energy Management System (SCADA/EMS) of modern power systems.

With the rapid advances in information technology (IT), it has been envisioned that such SCADA/EMS can be expanded far beyond the existing power delivery network that supplies electricity to an uncontrolled demand. Instead, advanced IT—so-called Smart IT because of its ability to actively acquire, process, communicate, analyze, and display information to provide interactive and adaptive control of both the system inputs and responses—will enable the power system to optimally manage both electricity supply and demand through real-time electricity pricing signals; interactive end-use sensors, communications, and controls; and environmentally based energy-trading mechanisms, as well as to effectively integrate distributed energy resources, including renewable, energy storages, and electric vehicles. Through advanced data analysis by the Smart IT, the power system can further anticipate future changes and develop appropriate responses, including self-healing in case of system breakdown and emergency control in case of natural or man-made disasters. In other words, Smart IT can provide constant information acquisition, processing, analysis, communication, and control to create a Smart Grid that simultaneously optimizes all generation, transmission, distribution, and end uses to continuously adapt to changing supply conditions, customer demands, and environmental and other policy requirements. This is a vision that has caught the imagination of both researchers and business people and gained the popularity among technical professionals and government officials alike throughout the world.

However, as IT continues its revolutionary changes, the Smart Grid vision also continues to evolve. To develop a Smart Grid based on a given set of Smart IT would not only be a long and expensive undertaking, but the results could also become technically obsolete as soon as it is completed. (A prominent example was the development of a Customer Information System by a major U.S. electric utility in the 1990s, which took 5 years at a cost of over 300 million dollars, but became outdated as soon as it was completed.) Therefore, in all countries in the world, Smart Grid concepts have been partially implemented in various segments of the power system, such as smart end-use management, two-way communications and interactions between utilities and customers, utility asset management with smart substations, and smart regular and emergency operation centers.

At the same time, these concepts have been also implemented more comprehensively in a smaller-scale local power grid, called the microgrid, in which there are often many independent power generators with renewable energy sources. Both these incremental developments of large-scale smart grid and the interactions between the more fully developed microgrids with the main power grid pose great challenges, but at the same time provide enormous opportunities for research in OR/MS in the optimization of resource allocation and system operation, see (Chakrabortty and Ilić 2011; Vasant et al. 2011) for some initial summaries.

See

- ▶ Decision Analysis
- ▶ Game Theory
- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Nonlinear Programming

References

Anderson, D. (1972). Models for determining least cost investments in electricity supply. *Bell Journal of Economics and Management Sciences, 3*, 267–299.

Benders, J. R. (1962). Partitioning procedures for solving mixed-variable programming problems. *Numerische Mathematik, 4*, 238–252.

Chakrabortty, A., & Ilić, M. (Eds.). (2011). *Control and optimization methods for electric smart grids*. New York: Springer.

Chao, H., Chapel, S. W., Morris, P. A., Sandling, M. J., Fancher, R. B., & Kohn, M. A. (1989). EPRI reduces fuel inventory costs in the electric utility industry. *Interfaces, 19*(1), 48–67.

Chao, H., & Peck, S. (1996). A market mechanism for electric power transmission. *Journal of Regulatory Economics, 10*, 25–59.

Electric Power Research Institute. (1983). *EGEAS, the electric generation expansion analysis system, EL-2561*. Palo Alto, CA.

Electric Power Research Institute. (1987). *Over/under capacity planning model, version 3, P-5233-CCM*. Palo Alto, CA.

Electric Power Research Institute. (1988). *EPRI products, volume 8, planning*. Palo Alto, CA.

Ikura, Y., Gross, G., & Hall, G. S. (1986). PG&E's state-of-the-art scheduling tool for hydro systems. *Interfaces, 16*(1), 65–82.

Institute for Electrical and Electronics Engineers. (1999). *Game theory applications in electric power markets, TP-136-0*. Piscataway, NJ.

Jenkins, R. T., & Joy, D. S. (1974). *WIEN automatic system planning package (WASP) — an electric utility optimal generation expansion planning computer code, report ORNL-4945*. Oak Ridge, TN: Oak Ridge National Laboratory.

Kaufman, A., (1975). *Reliability criteria — a cost benefit analysis, report* 75–9, New York Department of Public Service.

Keane, D. M., & Woo, C. K. (1992). Using customer outage cost to plan generation reliability. *Energy, 17*, 823–827.

Sidenblad, K. M., & Lee, S. T. Y. (1981). A probabilistic production costing methodology for systems with storage. *IEEE Transactions on Power Apparatus and Systems, 100*, 3116–3124.

Smith, J. E., & Nau, R. F. (1995). Valuing risky projects: Option pricing theory and decision analysis. *Management Science, 41*, 795–816.

Stremel, J. P., Jenkins, R. T., Babb, R. A., & Bayless, W. D. (1980). Production costing using the cumulant method of representing the equivalent load curve. *IEEE Transactions on Power Apparatus and Systems, 98*, 1947–1956.

Telson, M. L. (1975). The economics of alternative levels of reliability for electric power generation system. *Bell Journal of Economics, 6*, 679–694.

Trigeorgis, L. (1996). *Real options: Managerial flexibility and strategy in resource allocation*. Cambridge, MA: MIT Press.

Vasant, P., Barsoum, N., & Webb, J. (Eds.). (2011). *Innovation in power, control, and optimization, IGI global*. Pennsylvania: Hershey.

Yu, O. S. (1977). *An efficient approximation computational procedure for generation system reliability, technical report, mid-American interconnection network*. Chicago, IL.

Yu, O. S., & Chao, H. (1989). Electric utility planning is a changing business environment: Past trends and future challenges. *Proceedings of Stanford-NSF Workshop on Electric Utility Planning Under Uncertainty*, Stanford, CA, 253–272.

Yu, O. S, & Freddo, W. (1978). *An Efficient Electric Power Generation Maintenance Scheduling Procedure, presentation at ORSA/TIMS National Meeting*. San Francisco, California.

Electronic Commerce

Edgar H. Sibley[1] and Abeer A. Al-Hassan[2]
[1]George Mason University, Fairfax, VA, USA
[2]Kuwait University, Kuwait City, Kuwait

Introduction

Electronic Commerce (EC) has become a significant way of selling products and services because of the major improvements in electronic and Information Systems (IS) technology. This technology, however,

can only be exploited effectively by specialized business structures and organizational practices. At the same time, new laws, treaties, and international standards were enacted to respond to the worldwide changes needed to control and yet ease the movement of goods and data over borders. Because of its effectiveness, the term electronic commerce has been expanded to encompass types of systems that use similar technology, but have only marginal commercial aspects, such as governmental systems that interface with citizens. For example, the Internal Revenue Service (IRS) has a site that provides electronic forms for income tax submission and social systems that provide interfaces to members of social or professional organizations.

The definition of EC may be derived from that of normal commerce, which is as follows:

Commerce *(noun): the exchange or buying and selling of commodities on a large scale involving transportation from place to place* (*Merriam-Webster*).

Electronic Commerce is thus the use of information technology (IT) to provide an infrastructure with interfaces that allow communication between people and organizations for business or commercial purposes. It is buying, selling, or exchanging of products, services and information via an electronic network (generally the Internet, Intranet, and its extensions) (Efraim et al. 2010).

As a result of Information Technology (IT), most large stores include a pure brick (physical) operation and an on-line presence. The combination is termed "brick and click" and allows a seller to reach more customers and a world-wide audience, as well as compete with other businesses. Today, there are also stores that reside purely online (e.g., selling digital media, such as books, music, or operating systems) and rely completely on the Internet as its interface with customers. More businesses are expected to move online in order to stay competitive.

Using the Internet model, the process of buying goods and services becomes more transparent. EC allows the buyer to browse vendor offerings and prices.

Some History of Electronic Commerce

In the United States, starting in the 1960s major organizations used computers to automate previously manual operations, thereby improving the speed and accuracy of business computations. By 1970, most large enterprises had automated their simple manual business functions, such as payroll, and some early database management systems facilitated data sharing. Soon after this, the first internal networks were implemented by large firms. Additionally well-informed managers started interconnecting their geographically distributed computer systems via phone lines. Within years, the Internet was born and by the mid-1980s, there had been major adoption of network technology. Moreover, the advantage of communicating between consumers and suppliers (business to business or B2B) had led to the idea of electronic data interchange (EDI) where orders could be placed electronically, according to nationally defined and contractually approved methods. Electronically controlling inventory significantly reduced the costs of supply chain management. Because of the advent of gateways between networks and countries, the World Wide Web became a reality. The Internet had therefore joined phones and fax machines as a medium for commerce at a distance. All seemed well until the 1990s when the EC industry suddenly faced loss of public confidence. Banks and investors became doubtful of EC as a profitable venture and stopped funding EC startups. Consequently, the so-called dot com bubble burst. In the following years, however, EC sites gradually recovered from this setback and some e-businesses returned profits. As a result, today, most large US stores and businesses have effective Web sites with personal and corporate customers accessing them. This confirmed EC as an effective way of doing business.

Fields Associated with Electronic Commerce

EC technology changes the way that merchants conduct business; manufacturers develop relationships with other merchants; and consumers access information and obtain goods and services. Their IS must be adapted to take advantage of the opportunities of EC, as well as satisfy the expectations of those who establish a relationship with or purchase goods and services from the organization. This online medium redefines the ways in which companies and customers communicate.

As EC becomes a new source of customers and a means for the expansion and creation of business opportunities, many business and related disciplines become, directly or indirectly, an important part of the business model. Some affected business related areas are:

- Law and Policy: Legislators have passed new laws and are finding it necessary to modify or reinterpret old ones to deal with EC. The hierarchy of laws and regulations includes control over all transactions between governments, organizations, and people, thus incorporating international, national, and local laws. Major organizations must also negotiate for special considerations with various regulatory bodies throughout the world. Governments must also influence business processes to gain taxation and other revenue, protect national values and culture, and maintain relevance, while regulatory bodies have to deal with the effect of case law. Such issues of governance must be solved; within the firm, policy makers must therefore react by creating new rules and procedures that deal with the effect of the new modes of business.
- Finance and Accounting: This discipline must deal with new means of payment worldwide, therefore understanding the many different taxes (e.g., purchase tax, value added tax, etc.), currencies, and accounting requirements are important.
- Managerial: Corporate management must understand the cultural and political issues that affect the profitability and general operation of their organization around the world. The new way of conducting business must cope with many cultures and religions, whose members may find some content offensive or even blasphemous. The opportunity for illegal activities or even mischief also has increased; thus, ethical behavior plays a major role in the development of the EC Web site and also of the care of the information passing into and out of the Web site. Of course, this also implies a need for businesslike behavior within the Web site.
- Information Technology: The chief information officer, who is in charge of the IT systems and their maintenance, must deal with many issues including development of the Web site and its database and their maintenance. The Web site must be protected against the depredations of the cyber criminal or hacker.
- Psychology/Organizational Behavior: The system designers must be aware of the behavior of a customer while shopping online. Attention to this can aid in customer retention; navigation within a site, searching for products, and ensuring general ease of operation. Analyzing the differences between the feel or playfulness of Web sites are important to the customer who has the site as his or her major interface with the store.
- Marketing: This discipline must adapt to a new type of customer or face potential loss of market share. There is seldom any salesperson available on the site and thus the designer or architect of the site should provide a surrogate to aid the customer in navigating the site and learning of other people's feelings about the goods. The marketing consultant should, therefore, use new techniques such as data mining to find how to attract customers and implement new advertising techniques.
- Economics and Operations Research: Economists study markets, products, and commodities. As the Internet becomes an integral part of today's economy, supply and demand, as altered by EC, become important. Supply chain management is a particular example of the application of operations research (OR) principles to EC systems.

The Classification of Electronic Commerce

EC can be characterized by the types of entities that are communicating; these include Businesses (B), Consumer or Citizens (C), Governments (G) and Employee or Exchange (E). Obviously, these are paired and thus are referenced Business to Consumer (B2C) implying the use of specific types of transactions between Bs and Cs. The most important in terms of volume and financial commitments is B2B, where linked transactions take place between two or more companies resulting in supply chains. Some other classifications discussed by Efraim et al. (2010) include:

- Consumer-to-Consumer (C2C) – transactions between customers. Examples of such a site would be in person to person sale of artwork or second hand equipment;
- Business-to-Employee (B2E) – transactions within an organization, e.g., a business allowing its employees to perform transactions such as submitting time sheets online;

- Government-to-Citizen (G2C) – transactions where citizens make payments or apply for forms online;
- Mobile Commerce (M-COMMERCE), transactions that take place via a network, possibly using a phone or personal digital assistant (PDA) as the input device;
- Location Commerce (L-COMMERCE) – transactions that locate the user and solicit attention, such as the use of global positioning satellites (GPS) data in mashups on the PDA or cell phone or by sending a text message to a potential customer near a restaurant.

The Process of Electronic Commerce

Customers using an EC site experience a similar procedure to that when shopping at a traditional store. The normal steps are:

Step 1. Search for the Store of Interest
Similar to using a phone book's business directory, a potential buyer will often browse, or surf, the World Wide Web (WWW) to window shop or find a product. This surfer may use a search engine to locate stores, expand the search to worldwide, and establish conversation/dialogue with stores. One particularly important aspect of operation in the WWW is in the use of intelligent autonomous software agents or bots. These may be spawned by a site as a means of gathering information about possible suppliers or competitors. The bots are able to search the Web (mimicking real person inquiries) via search engines and visiting possible sites of interest, and may determine which sites offer the best prices or service. This can help the purchaser make a choice based on their criteria.

Step 2. Navigation and Selection
Similar to walking within a store, a surfer moves within a Web site to find items of interest. This may result in a customer selecting a particular item because of its price, the site's terms and conditions, and the seller's overall reputation. For chosen items, a selection is made by moving the product or service into a so-called shopping cart. A transaction may also depend on negotiating a price between the buyer and seller in some medium of exchange or barter. The seller may thus present the product or service for sale and reduce the asking price until a sale occurs, particularly in retail situations where the costs of maintaining inventory may erode the profit margin at the asking price or when the inventory has a limited lifetime (like fruit and vegetables or seasonal material). This devaluation is usually hidden from the customer except at discount outlets.

Step 3. Checkout (Payment)
Similar to payment via a checkout counter, payment within an EC site involves forms of electronic payment, including ways of submitting e-cash or billing systems. The actual exchange of money may occur either before or after delivery, with intermediaries (such as credit and debit card companies or banks) aiding in the exchange. For EC to work, efficient and paperless settlement processes must exist; each charges the merchant a relatively small transaction cost.

Step 4. Delivery
Similar to a traditional buying technique, once the sale has been completed, the parties must agree on the time and method of delivery of the product or service. For products, online merchants sometimes maintain small inventories of the most popular products and coordinate with suppliers for the delivery of the remainder. This coordination is contracted in advance and the consumer is unaware of the inventory of the merchant. Since business consist of moving quantities of inventory from the manufacturer to the consumer, lowering inventory costs at each step is important. Just-in-time (JIT) operations can reduce inventory cost to zero (if the EC site is merely acting as an intermediary in the transaction) or close to zero. The vendor will often send a confirmation that includes information so that the buyer can track the process of delivery.

The Elements of an Electronic Commerce Site

Any business with an online Web site should consider how certain important elements are to be implemented. The most common sections/elements on EC sites (Efraim et al. 2010) are:

- About Us section – this contains information about the business. Some include the history of the company, its vision, mission, and objectives, while others include management team and major hierarchies (departments, etc.) of the company and even a message from its Chief Executive Officer (CEO).
- Terms of Use – in which the company provides information on its overall policies, especially its

view of and care in ensuring security and thus how it stores and preserves private information provided by its customers when making a purchase.
- Customer Database/Customer Account – almost every company creates an account for its customers. The customer account normally allows the user to give a username and password for selecting and tracking purchases. However, if a customer is only window shopping or navigating an electronic catalog, this is seldom required, though the EC business could use such information in future promotion campaigns.
- Electronic Catalog – an electronic catalog offers information about the products or services provided. They can be displayed in the form of text, pictures, or even videos. A customer can be sent a customized catalog compiled from knowledge stored in the Web site database about the customer's prior purchases.
- Search Engine – for large companies, a special search engine is often provided to customers to help them narrow and speed up their Web site navigation by typing in keywords that should help them find information on possible products or services that appear to suit their needs.
- Shopping Cart – a shopping cart or shopping bag where shoppers can accumulate items that they are considering or intend buying.
- Payment Gateway – a check out process where a customer pays for products and services on the Web site.
- Contact Us – a section that usually gives information on ways that a customer may contact the company: toll-free numbers, email addresses, and even live chats are available on most sites, thereby providing a surrogate for the missing salesperson in an automated site.

Non Traditional Business Models and EC

Social Networking and EC

Social networks are analogous to traditional town meetings; they can be viewed from two revenue perspectives:

1. Through their Business Model, where their revenue was mainly achieved by assessing an annual or shorter term fee or subscription from participants.
2. As an electronic billboard, displaying advertisements, such as pop-up, banners, or sidebars that provide income.

They provide regular meeting space where members can create groups or sell products simply by posting them, the electronic equivalent of classified ads in a C2C format.

Auctions and EC

Alternatives to traditional auction houses may be found in EC form. English (forward) or Dutch (reverse) auctions, for example, may be completed online.

Bartering and EC

Some EC sites now use pure bartering methods or pass merchant money (scrip) as a way of charging and paying for goods or services. The site revenue is then obtained from traditional advertising methods or the income of the Web site developer from users purchasing the scrip at a slight cost from the site owner for use in exchanging or bartering. One advantage to barter participants is the lack of a trail showing any equivalent to a monetary transaction; this makes such transactions difficult to track by local or governmental authorities who find it difficult to impose taxation.

Revenue Models

Revenue models in EC describe "how the firm will earn revenues, produce profits, and produce a superior return on invested capital" (Laudon and Traver 2010). The most common revenue models are:

- Advertising – receiving revenue from companies advertising on the site.
- Subscription – a fee charged for offering content or service to members of a special site, such as a professional organization or social club.
- Transaction fee – charging a fee for a particular transaction on a site, similar to charging for a newspaper classified advertisement.
- Sales – revenue based on the profit from selling goods, services, or information.
- Affiliate – revenue from a referral fee resulting from redirecting a consumer to another site at which he or she made a purchase.

Web Economics

The area of economics is somewhat expanded by adding commerce over a network. The term Webonomics was coined to show a difference and underline new effects. It can be considered as the study of the effect of the WWW on production, distribution, and consumption of goods and services. The world is now dealing with an information-based economy and this redefines the way that the network can be exploited.

At least four major groups profit from using a Web:

- Consumers – The buyer and the individual or group wishing to trade something of value. The consumer will be interested in an online product or service only if it is the best, cheapest, or most convenient alternative and is from a trustworthy source; lack of safety and privacy standards hamper growth.
- Content Creators – The providers of raw materials and manufacturers. They are vying to attract consumers and often maintain their own Web sites, which must be as attractive and as up-to-date as possible; they are hoping to build their image among the millions of surfers.
- Marketeers – The marketers of products, facilitating the barter, trade, and purchasing processes bringing content creators to consumers and vice versa.
- Infrastructure companies – The engineers of the Web who build and sell computing and communication hardware, such as Web servers, routers, etc., as well as the software for building and maintaining sites.

Internet Marketing and Electronic Commerce

The use of EC allows the vendor to utilize new methods for marketing with no immediate relationship to non-EC marketing. Four of these are:

- Banner advertising – This is often found on pages regularly read by buyers, including the header page of a search engine, news service, or local news site.
- Preference profiles – By using tracking mechanisms, such as cookies, a vendor can determine the likes and dislikes of prior customers and target them for future special offers. Cookies are text files placed on the user's computer hard drive when visiting some Web sites; they contain data that was collected from a purchase and stored for future use. During subsequent visits, these text files are retrieved by the site software and used to personalize advertising. Thus, the user has to enter less data when making a new purchase and sellers can tailor marketing efforts via data mining.
- Leveraging information about a community – The seller can use push technologies, based on market assessment data, such as expected income or sociological characteristics of an area. These allow the seller to provide unsolicited information about a product or service that may be attractive such as winter cruises to warm climates for retired people living in relatively high priced localities; the buyers provide (sometimes unwittingly) a profile of their interests. For example, a book seller will send e-mail to customers when a previously purchased author has had a new book published.
- Broadcast desired product requirements – These provide a virtual tradingfloor where the buyer presents requirements and solicits bids from vendors. The buyer can then contact the most promising vendor and initiate a purchase. This is comparable to newspaper classified ads.

Technological Aspects

All EC sites can be considered to consist of two layers: the connectivity layer and the modules.

The Connectivity Layer

- EC has, as its base, a communication network, which is the way that organizations and people communicate. The medium for the network may vary in complexity from:
- Direct Connection – similar to a phone line between organizations or parts of an organization. This was the original way that early B2B systems were implemented and it is still in use. EC for some vending machines and modern parking meters, which use cash or credit card as a method of payment (micropayments); these transactions are for less than $10.00. However, more locations such as airports and shopping malls have vending machines that sell electronic devices such as cameras worth $200.00 or more. And

- Internet Connections – through a public network, like the Internet, with a need for some way of protecting the transmission of data from interception, corruption, or altering, such as encryption of the data on and off the network.

Direct Connection Networks were originally built and maintained by large manufacturing corporations primarily to support their supply chains. These were effective, but the cost of building and maintaining private networks can seldom be justified, except when the security needs of the organization are very high. Some examples of such special needs are in national security systems (the Department of Defense, etc.) or industries with extremely high competitive environments. These may require substantial protection over technology or process and thus need internal data retention with limited sharing of data with members of its supply chain (such as leading members of the aircraft and electronics industries). Most commercial systems and private users find it sensible to use an external, public service that provides an interface to the internet (an so-called Internet Service Provider or ISP). The speed of access and transmission of ISP services has risen markedly through the years due to the demand of users and the availability of faster network components. The economies of scale make the large systems provided by ISPs much cheaper and, thus, few organizations can no longer afford the luxury of a private network. Virtual Private Networks (VPN), however, are becoming popular and new architectures, such as cloud computing, seem to be viable.

The VPN is a special service of an ISP that provides high security by encrypting and authenticating technology to protect data entering and leaving the storage facilities. Thus, its users can experience a similar effect to that of a private network except that there is some delay due to the time spent in encrypting and decrypting the data. The security can be tailored to the organization, group (such as branches or divisions), or individual at the cost of servicing sets of encryption keys.

Cloud computing involves the use of the Internet as a massive, but inexpensive, storage device provided by an ISP. While it has little security (unless encryption is added in much the same way as in a VPN), the economies of scale again make this architecture very cheap. Many social networks are therefore using a cloud for storing their data, but with little or no expectation of security on the part of their users.

The Modules

- There are four modules that interface with the connectivity layer. They communicate through the layer and to the Internet. The modules are:
- Customer Information – This contains all customer related data such as name, address, and credit card information, possible interests, latest purchases, etc.
- Product and Services Information – This contains information about all products and services that are currently available for sale, including their prices and discount, detailed description of the product, and possible graphics. This module presents the information to customers in an easy to read and pleasing format.
- Payment Information – This module interfaces with the seller's financial institution so that it can authenticate the buyer's solvency prior to completing a sale.
- Shipping Information – This contains information about the various shipping methods available and their costs. The information is sent to customers who then select their preferred method and the consequent price is added to their payment information.

Trust in Electronic Commerce

Without trust no one will purchase from a store; the customer must believe that the product is well made and the seller must believe that the payment is not fraudulent. Similarly, in EC there must be trust between the parties, except that they are usually separated by some geographic distance and are potentially unknown to one another. Consequently, the primary challenge to the proliferation of EC has been in bolstering trust in the site and ensuring trust in the payment. The first entails assuring the customer of the security of the site and the vendor's intent to exclude hackers and crackers, while the second is provided by the banking system by authenticating the payment. Thus, security requires mechanisms that ensure non release and safe retention of data while privacy ensures that personal and other important data is not released to other people or organizations without the customer's explicit permission that the vendor may do so.

A lost credit card can result in significant loss to its owner or the credit card company. If someone steals

a card and publishes its number on the Web, there is likely to be a very rapid loss of cash, more than that from a single criminal who holds one stolen credit card.

When a direct link is established between a private network and the Internet, a company's internal network and all company resources are extremely vulnerable to hackers. Thus, the more a company allows access to its sites (describing products, services, and processes), the more vulnerable the enterprise is to information loss (stealing). For this reason, most systems use one of the many forms of encryption, which allows them to provide the necessary services of:

- Non-Repudiation – proof that the person or organization really participated in an electronic exchange of information resulting in a financial commitment;
- Confidentiality – mechanisms to ensure that any message cannot be read by unauthorized individuals;
- Authorization – internal functions to ensure that the internal staff are assigned access based on their role in the organization;
- Integrity – mechanisms to make sure that data are not modified by unauthorized persons during their storage or transmittal; and
- Authentication – Because an Internet transaction does not involve face-to-face interaction, this must guarantee that the person participating is not a fraud. The authentication function verifies that the offer of goods or services is from a registered provider and validates the identity of the purchaser, while ensuring and monitoring that each party is adhering to the terms of agreement. There are two basic models for authentication: digital signatures, and certificates.

"Web site owners are using a terms of use/terms of service agreement posted at their Web site to allocate, limit, reduce, mitigate, avoid, and otherwise manage potential risks in cyberspace," (Westermeier et al. 2007). Almost all Web sites contain a Terms of Use section. Such service agreements add to the confidence in or trust that a customer feels for a Web site.

Furthermore, some users and vendors are concerned with (i) transaction or credit card details stolen in transit, (ii) customers' credit card details stolen from a merchant's service, or (iii) merchants or customers masquerading as legitimate buyers or sellers (Chaffey et al. 2007). Thus the Terms of Use section explains how the site will provide security for concerned users by discussing how it deals with authentication, unauthorized access, and hardware and software used to provide security (e.g., its use of firewalls). Similarly the section provides Web site users information on how the site deals with their privacy by stating that the customer information will be kept confidential, that there are a restricted set of operations allowed on the data (such as data mining for customer interests), the use of cookies, and any third party to whom data maybe released, such as a bank.

Validation in Electronic Commerce Research and Development

Research on EC Web site effectiveness generally involves the collection of customer feelings about a specific aspect of a Web site and its consequent use. As such, the research tends to be psychological in nature, requiring the development of a model of the process being investigated, determining the variables that interact in that process, and using this model to help in positing hypotheses. It is then necessary to examine the variables for any overlap or interactivity between them and decide how to measure them. This leads to the construction and validation of survey instruments that measure the feelings or perceptions of customers about the Web site or its architecture.

Thus, the research requires the development of validated methods of measuring the customer's reaction to the Web site followed by collection of data from a relevant population of Web site users. The data from the survey must then be analyzed utilizing one of the many statistical methods, such as Structural Equation Modeling (SEM), to derive relationships between the data and thus prove or disprove the hypotheses.

References

Chaffey, D. (Ed.). (2009). *Internet marketing: Strategy, implementation, and practice* (4th ed.). Essex, England: Pearson Education.

Laudon, K., & Traver, C. (2010). *E-commerce: Business, technology and society* (5th ed.). Upper Saddle River, NJ: Pearson Prentice Hall.

Turban, E., King, D., & Lang, J. (2010). *Introduction to E-Commerce*. Upper Saddle River, NJ: Pearson Prentice Hall.
Westermeier, J., Plave, L., & Halpert, J. (2007). *E-business: The E-business legal survival kit*, Piper Rudnick LLP, 1200 Nineteenth Street, NW, Washington, DC.

Elementary Elimination Matrix

A square nonsingular matrix obtained by replacing a column of the identity matrix by some vector. Every pivot operation on a system of linear equations is equivalent to multiplication of the system from the left by an elementary elimination matrix. In the simplex method, such a pivot matrix is called an eta-matrix.

See

- ▶ Matrices and Matrix Algebra
- ▶ Simplex Method (Algorithm)
- ▶ Simplex Tableau

Elimination Method

- ▶ Gaussian Elimination

Ellipsoid Algorithm

The first polynomial-time algorithm for linear programming. The ellipsoid algorithm was originally developed by Shor, Udin and Nemirovsky as a method for solving convex programming, but it was Khachian who showed that this method can be adapted to give a polynomial-bounded algorithm for linear programming. The basis of the ellipsoid algorithm is a method for finding a feasible solution to a set of linear equalities. This method constructs a sequence of ellipsoids of shrinking volume, each of which contains a feasible point (if one exists). If the center of one of these ellipsoids is feasible to the system of inequalities, the algorithm terminates. If not, then after a known (polynomial) number of iterations the volume of the ellipsoid will be too small to contain a feasible point, and hence the system is infeasible. This method for solving inequalities can be used to solve linear-programming problems in polynomial time, by writing the primal and dual feasibility constraints and the equality of the primal and dual objectives as a system of inequalities. Despite its tremendous theoretical importance, the ellipsoid algorithm appears to have little practical significance, since its computational performance has been very poor.

See

- ▶ Computational Complexity
- ▶ Interior-Point Methods for Conic-Linear Optimization
- ▶ Simplex Method (Algorithm)

References

Schrijver, A. (1986). *Theory of linear and integer programming*. New York: John Wiley & Sons.

ELSP

Economic lot scheduling problem.

See

- ▶ Production Management

Embedded Markov Chain

- ▶ Imbedded Markov Chain

Embedding

(1) The drawing of a graph on a surface without edge crossings. (2) The use of a subsidiary stochastic process to solve a larger one in which the subsidiary is contained.

See

- ▶ Imbedded Markov Chain
- ▶ Queueing Theory

Emergency Services

Kenneth Chelst
Wayne State University, Detroit, MI, USA

Introduction

Emergencies range in scope from the routine—situations involving limited police, fire, and/or medical personnel—to the catastrophic—such as large-scale natural or man-made disasters. The latter category includes Hurricane Andrew and Hurricane Katrina, the Exxon Valdez and Deepwater Horizon oil spills, nuclear power plant meltdowns in Chernobyl and northeast Japan, earthquakes in Haiti and New Zealand, and tsunamis in Indonesia and Japan.

Whether emergencies occur hundreds of times each day, once a decade, or once in a lifetime, the planning and management of emergency services are complicated by various uncertainties:

1. The time and location of the emergency
2. The scope of the emergency
3. The type of call and the personnel and equipment needed to handle the initial emergency
4. The type of call and the personnel and equipment needed to handle the aftereffects of the initial emergency

Because of the differences in frequency and scope between the more routine basic emergency and the rare large-scale disaster, different strategic, tactical, and operational planning approaches are required.

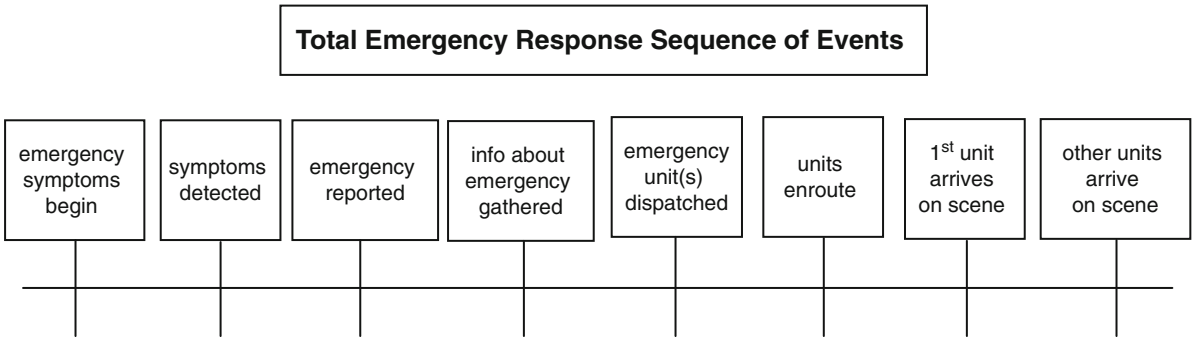
Common Emergencies: Police, Fire, and Emergency Medical Services

Police, fire, and emergency medical services (EMS) all operate in a complex, unpredictable year-round 24-7 environment. Typically, managers of these emergency services function within severely constrained budgets and face two common complex operational questions:

1. How many emergency service vehicles should be staffed each hour of the day?
2. Where should these vehicles and personnel be located?

In an ideal world, decision makers would focus on their ultimate goals when addressing these questions. Police officials would evaluate strategies in terms of their relative effectiveness in reducing crime and the fear of crime. Fire service administrators would allocate equipment and personnel so as to reduce fire damage. EMS managers would compare alternatives with regard to lives saved and disabilities avoided. Unfortunately, the relationship between many decisions and their ultimate impact are not fully understood. For example, the impact of a 10% increase in patrol cars on crime levels cannot be predicted. Similarly, the number of lives saved as result of building an additional fire station cannot be estimated. And, further, the impact of ambulance service on survival rates following automobile accidents is not fully understood. In contrast, there has been significant progress in understanding the relationship between the number and type of emergency medical units in a community and the likelihood that ambulance service will save the life of a person in full cardiac arrest (Erkut et al. 2008).

As a result, operations research (OR) models designed to assist emergency service decision makers use the surrogate measure of response time, as well as workloads, when structuring resource allocation decisions. Figure 1 illustrates the total response pattern to an emergency. It begins with the onset of symptoms of an emergency (e.g., chest pains, smoke, suspicious persons). There is a delay until the symptoms are recognized as an emergency and a further delay until the emergency is reported. The dispatcher must process the call and find an emergency unit to dispatch. It may take time for the unit with its personnel to leave its location and begin traveling to the emergency. The final response time delay is the travel time to the scene of the call. For fire services and ambulance services housed at stations, there could be an additional delay between the time the station is notified of the call and the time it takes for the vehicle to be in motion on the street. If multiple units are dispatched to the same call, there will be multiple arrival times. OR models focus on two components of



Emergency Services, Fig. 1 Total emergency response sequence of events

response time: (1) the queueing delay, the time from call receipt until an emergency unit is available for dispatch, and (2) the travel time to the scene of an emergency.

In the 1970s, faculty at M.I.T. and researchers at the New York City RAND Corporation (a joint venture between New York City's government and the RAND Corporation) developed models that address these two basic deployment questions, with funding from the National Science Foundation and the Department of Housing and Urban Development (Larson 1972; Walker et al. 1979). One specific relationship that developed from this research was the square-root law that estimates the average travel distance to a random call:

$$\text{Average Travel Distance} = c^*\{A/[N(1-b)]\}^{1/2}$$

where A = area, N = the number of emergency service stations or vehicles, b = the proportion of time an emergency service unit is busy, and c = a constant of proportionality (usually between 0.6 and 0.7) calibrated from real-world data. The parameters a and b are city-specific, while N represents the key decision: the number of units to deploy. This simple square-root formula is utilized to make aggregate decisions on deployment. It predicts that quadrupling the number of available emergency units reduces average travel distance by only 50%. The estimated travel distance is converted into travel time by either assuming an average speed or by using regression analysis to define a nonlinear relationship between distance and time. Police deployment models use average speed, while fire models tend to use regression analysis.

A major problem in planning emergency services is that the number of calls per hour is a random variable. Thus, decision makers use the Poisson process, a probabilistic model for forecasting the number of calls in a designated period of time. These forecasts are the input into simulation or queueing models that estimate the average delay between the arrival of a call and the dispatch of an emergency unit. The average queueing delay is added to average travel time to determine total average response time.

Police Services: The allocation of police patrol resources is complicated by a number of factors. First, the number of police patrol units on the street is usually much larger than the number of fire stations or ambulances. Second, these police units respond to a wide variety of calls with different levels of urgency (e.g., armed robbery in progress versus loud music next door). In small and medium-size cities and suburbs, quality-of-life calls and traffic enforcement are often the bulk of the police workload. Lastly, police call rates vary significantly by time of day and day of week. Because of this added complexity, planning in a large city might proceed in three phases.

In the first phase, police officials use aggregate data and a model such as PCAM (Patrol Car Allocation Model) to determine the number of patrol units to assign to different parts of the city for different periods during the week. Next, police commanders use a descriptive model such as the hypercube or simulation model to design patrol beats for individual patrol units (Larson 1974). These models forecast response times for different priorities of calls, as well as workloads of patrol units. Police managers can then apply a mathematical- programming model to design a work schedule for personnel to ensure that there is enough manpower on duty to staff the proposed street personnel plan (Green and Kolesar 2004).

In all decision making regarding emergency services, there is a tension between efficiency and equity. A deployment plan that minimizes average response time will tend to concentrate resources in high-risk areas, leaving other areas with response times significantly longer than average. This conflict has been addressed in a variety of ways. To achieve equity, PCAM allows the decision maker to establish minimum response time standards for each region of the city. Once enough patrol units have been assigned to each region to achieve these minimum standards, PCAM can allocate excess patrol resources to minimize citywide average response time (Chaiken and Dormont 1978). Alternatively, concepts of multi-attribute utility and group-decision making can be used to assist decision makers in exploring tradeoffs between equitable deployment plans and efficient ones.

The police are responsible for more than simply responding to emergencies. One goal of routine or directed patrol is to deter crimes. In addition, police patrol units are often the first investigators at a crime scene. (See the entry "► Crime and Justice" to explore the link between police and crime).

Local Fire Services: Fire service deployment is the least complex of the three services to analyze. Fire call rates are typically low; firemen spend less than an hour a day responding to calls. Response times are generally affected by the number of fire stations and whether firefighters are awake or asleep, rather than variations in the average call rate over the course of a day. Consequently, most municipalities tend to staff their fire stations at the same manpower levels around the clock to provide a constant level of fire protection. Some locales, such as Loudon County, Virginia, vary the mix of volunteers and full-timers by time of day and day of week. This is more a function of the availability of volunteers than a reflection of varying call rates.

The low workload simplifies the analysis of fire station locations. Fire station planning models make the reasonable assumption that the nearest fire unit will be available to dispatch immediately when a need arises. Among deterministic optimal location models that are used to situate fire stations, coverage and *p*-median models stand out. A coverage model locates fire stations so as to maximize the number of people or houses situated within, for example, four minutes' travel time of the nearest station. This is consistent with International Association of Fire Fighters (IAFF) standards that are reported in terms of coverage. The *p*-median model locates stations so as to minimize the average response time to a target population at risk.

In some complex environments, such as New York City, operations researchers have used a descriptive model that allows the decision maker to add or delete fire stations and then assess the impact (Walker et al. 1979). The descriptive model provides a wider range of performance statistics than the coverage and *p*-median models, and predicts response times for the arrival of a second and third fire engine; it also differentiates between equipment with different roles (e.g., engine trucks and ladder trucks).

As a result of the 2007–2009 recession, many cities have seriously considered substantial cutbacks in fire services. Such cuts are only possible because city officials generally staff for relatively rare events in emergency services. For example, the International City/County Management Association (ICMA) found that in one city of 300,000, there were still five units available for immediate response even during the busiest hour of the year.

Thus the reduction of staff without compromise in service has been an area of close inquiry. Although the IAFF standard is four men per engine, a National Institute of Standards and Technology (NIST) study found that four men were only 5% more effective than three, using standard equipment and water (Averill et al. 2010). Furthermore, fire hoses that operate with compressed air foam rather than water are much lighter and easier to handle, thus requiring fewer personnel than calculated in the IAFF standard. Models for staffing and overtime related to providing coverage for vacations, unscheduled absences, and retirements have also been studied.

Emergency Medical Services: In cities and densely populated counties, ambulance services operate at higher workloads than fire services. Often they are included as part of the fire service. In those instances, EMS calls often account for more than 80% of the total call volume of the fire/EMS. Utilization rates of 15% to 30% are not uncommon, and, in busy periods during the week, workloads exceed 50%. Consequently, ambulance location models that assume the nearest stationed ambulance will not be busy when a call comes in are oversimplifying reality. This has led to the development of more

complex coverage models that incorporate concepts such as backup coverage and workload adjustment factors. By doing so, these models approximate probabilistic concepts while maintaining the capability of using a deterministic optimal location model to place ambulances. An alternative to these optimal location models is the hypercube queuing model. This model is primarily descriptive and can be used to evaluate ambulance placement plans. Heuristic search algorithms have been appended to the hypercube model to facilitate the search for better solutions (Morabito et al. 2008).

Unlike fire equipment, ambulances and other EMS vehicles may be stationed on the street and easily relocated. State-of-the art dispatch systems facilitate the dynamic relocation of these vehicles. The relocation may be triggered by an overload in one area that has tied up the local ambulances, leaving a region uncovered. Alternatively, these vehicles may be redeployed as a result of established patterns of demand according to area, time of day, and day of the week. Both integer-programming and dynamic-programming models have been developed to assist in this redeployment (Maxwell et al. 2010).

The final stage of emergency medical treatment occurs in an emergency room. Simulation is frequently used to study alternative emergency room designs, staffing levels, and triage policies within this department (Wang et al. 2009).

Policy Questions: The models described above focus on developing plans to improve day-to-day operation of emergency services. OR models have also played an important role in exploring a number of policy questions with regard to cost benefits. For example: (1) which is more cost-effective, to spend public dollars on more fire stations to reduce response time or to subsidize the placement and maintenance of smoke detectors to reduce the delay until the detection of the fire? (2) Which is more cost-effective, more but less costly basic life support ambulances or fewer but more expensive advanced life-support ambulances? (3) What is the cost-benefit ratio for automatic defibrillators? (4) What are the relative benefits of one- and two-officer patrol units? (Chelst 1981).

Cities with populations as large as 70,000 have trained public safety officers to handle both police and fire emergencies. This has proven cost-effective in some locales but not in others. OR models have been used to study such mergers of emergency services, thus enabling decision makers to assess the potential impact on fire and police response times and cost (Matarese and Chelst 1991).

Despite the success of OR models in key operational and policy questions, relatively few cities employ them on a regular basis in their decision making regarding emergency services. Success stories from various cities still appear periodically in the literature, but there is no critical mass of researchers or operations research analysts dedicated to improving the performance of these types of emergency services. Instead, the typical local study involves one or more faculty members working with a local area emergency service to solve a particular problem. Such work generally entails no more than minor variations on existing research.

Operations researchers whose area of specialization is location theory continue to enhance their models. This has included expansion of the definition of coverage models, incorporation of multiple objectives, and inclusion of probabilistic issues that relax the assumption that the nearest emergency service responder is available when a call comes in. These enhancements, however, have not generated any greater practical interest from administrators. One interesting development is a software system, ALIAS, that was designed to assist ambulance location decisions. Its strength is that it integrates geographic information system (GIS) technology into a descriptive decision-support system. The system enables a decision maker to consider multiple objectives while making specific location decisions.

The recession that started in 2007 and the subsequent housing collapse have led to serious shortfalls in financing emergency services. City managers around the U.S. are taking unprecedented steps in requesting substantial cuts in emergency service personnel. In 2007, ICMA began providing city officials with a clear assessment of the workload of their police and fire departments, as well as an analysis of response time. A typical police report overlays workloads of all types against deployment levels. In one instance, a city council decided to reduce its patrol force by 30%, returning it to a level last seen in the late 1970s.

The most important change in policing since 1990 involves descriptive analytics. New York City pioneered the use of timely crime data to focus police resources. These data were used in regularly

scheduled, wide-ranging meetings in which precinct commanders accepted responsibility and were held accountable for implementing strategies to reduce crime. This concept was called COMPSTAT and copied in a number of major cities (McDonald 2001). With the spread of computerized mapping, even smaller cities can routinely produce crime maps to identify hot spots. Researchers have now moved beyond mapping and developed predictive models of crime patterns (Gorr et al. 2003). In Great Britain, researchers are making progress in estimating the likelihood of an arrest for a burglary reported in progress as a function of response time.

Disaster Planning and Management

In contrast to police, fire, and emergency medical services, disaster planning, in terms of both frequency and scope, is at the extreme end of the spectrum of emergency planning and management. Disasters of limited scope, such as commercial airplane crashes, may occur once a year. Calamities on the scale of the Exxon Valdez oil spill, Hurricane Andrew, the Bhopal-Union Carbide disaster, or the Chernobyl nuclear accident were thought to occur no more frequently than once a decade. However, perceptions may be changing as four major disasters occurred in just one ten year period between 2002 and 2011. In any case, when they do occur, they can affect the lives of millions of people and despoil millions of square miles of the environment. As a result of a series of massive disasters, research in disaster planning and management has gained added urgency, and decision-support systems to help plan for and manage disasters are a growing area of research and software development (Altay and Green 2006).

This growth has been driven in part by regulatory requirements that mandate disaster plans, as well as recognition that prevention and planning are cost-effective. The International Emergency Management and Engineering Society (TIEMES) was formed in 1994, and, in 1998, a special issue of IEEE Transaction on Engineering Management was dedicated to this topic. The most visible U.S. governmental agency in this area is the Federal Emergency Management Agency (FEMA). With respect to past disasters, there is also increasing global cooperation in developing early warning systems. A global warning system for tsunamis was developed in reaction to the 2004 massive tsunami that led to catastrophic loss of life in multiple countries in South Asia. In addition, national public health agencies cooperate in tracking the spread of epidemics.

Emergency Services, Table 1 Disaster categories

Category 1	Category 2	Category 3
Locally contained disasters	Localized disasters with potential for spread	Massive disasters
Aircraft or shuttle disaster	Forest fire	Hurricane
Mine explosion	Toxic chemical leak (train wreck or tanker overturn)	Tsunami
Disaster at sea (ship or ferry)	Hazardous oil spill	Major snow or ice storm – large-scale power outages
Major terrorist attack	Nuclear power plant accident	Floods
Mudslides or avalanche		Earthquake
High-rise fire		Volcanic eruption

Locally Contained Disasters: Disasters can be grouped into three categories that reflect their scope and duration (see Table 1). The first category covers disasters of limited duration, such as an airplane crash, space shuttle explosion, mine explosion, cave-in, or bombing. These disasters involve a single short-lived catastrophic event. The impact is usually limited to the individuals in the vicinity of the disaster, though it may, like that of 9/11, affect life far beyond the immediate focus of the terrorist attacks. The primary opportunity for risk mitigation in Category 1 disasters is in disaster prevention. Another strategy is to contain or dissipate the initial force of the disaster and work to increase the possibility of survival. Rapid evacuation may also be important if there is a risk of secondary explosions or continuing collapse, as in the case of a mine or building explosion.

Operations research has played a significant role in understanding and avoiding the risks associated with air transportation (Machol 1995). Collision-risk models have been used to establish safe and efficient separation standards. Simulation models have been used to design better runway configurations to avoid takeoff and landing disasters, while maximizing the efficiency of airports. In the case of the space shuttle, probabilistic models were used to identify the greatest sources of potential catastrophic failure, and

organizational pressures were identified that also contributed to increased risk of failure.

Evacuation has been an area of modeling at every level of disaster (Yi and Özdamar 2007). Simulation has been used to study mine-fire escapes and as a training tool to prepare miners to make the right decisions in an emergency (Cole et al. 1998). Evacuation of buildings is the subject of a number of papers that model the escape routes as a network. If probabilistic issues are factored in, a queueing network model can be used to study evacuation capacity. Early warning systems increase the potential effectiveness of evacuation procedures. Florida has developed effective strategies for evacuating large areas when a hurricane is forecast. In addition, Florida counties have plans to locate disaster recovery centers to assist victims of a local disaster. Unfortunately, there was no effective evacuation strategy or assistance for the New Orleans area as Hurricane Katrina approached. (See the entry "► Fire Safety Modeling and Applications" where evacuations are discussed in more detail).

Localized Disasters with Potential for Spread: The second category involves disasters of longer duration with potential to spread. This category includes forest fires, oil spills, hazardous waste leaks, and nuclear power plant accidents. Forest fires have drawn significant attention from operations researchers in large part because the USDA Forest Service has used analytic models in a wide range of areas. Decision analysis has been used to study different strategies and tactics to clear away underbrush so as to limit the scope of fires. A simulation model known as FARSITE calculates how a fire will spread under specific prevailing weather and environmental conditions (Finney 2004). Understanding and planning for forest fire spread is a continuing area of critical research. The USDA Forest Service developed the National Fire Management Analysis System (NFMAS) to assist in planning attacks on forest fires (Donovan et al. 1999). The system is a simulation that is used iteratively to determine the effect of different decisions designed to reduce the economic impact of a specific forest fire.

With regard to hazardous waste, the focus has been on models to route shipments of hazardous materials so as to reduce the risk in transit and to determine where to locate unpopular facilities (List and Turnquist 1998). The Hazardous Materials Transportation Uniform Safety Act of 1990 has improved the routing of shipments of hazardous materials and the planning for emergency response. Because there are different stakeholders affected by the routing decisions, the models are generally multi-objective. One set of OR models integrate the routing decision with the siting of emergency response teams.

Planning for massive oil spills begins with broad systemic assessment of risk so as to reduce its likelihood. This is followed by developing a plan to pre-position appropriate containment and cleanup equipment. Mathematical programming has been the modeling technique of choice. Once an accident has occurred, the key task is to forecast damage spread as a function of weather and environmental conditions. The challenge then is to rapidly assemble needed people and equipment resources so as contain the accident as soon as possible. A mathematical-programming model was developed to model the dispatch of cleanup equipment (Iakovou et al. 1996).

If the forecast indicates that there are people in harm's way, they have to be evacuated as rapidly as possible. The Nuclear Regulatory Commission requires utilities to develop and update evacuation plans. MASSVAC is a software package that models traffic movement during evacuation and has a user equilibrium assignment algorithm to improve evacuation time. Once evacuation has been accomplished, the next responsibility is to clean up the mess or repair the damage. The goal is to either return the environment to its original state or, more likely, to bring it to an equivalent state of environmental health. OR's role in the long-term problem of restoring the environment has been limited to waste water contamination problems.

Massive Disasters: The third category involves massive disasters that can encompass a region, a state, or even a whole country, such as Haiti. These include hurricanes, major snow and ice storms, massive earthquakes, large volcanic eruptions, or extensive flooding. These massive natural events may not be preventable, but governmental strategies that encourage better planning can reduce the impact of the disaster. In 2011 a number of cities along the Mississippi River benefited from this type of planning. They were saved from being deluged by floods when overflows were diverted to less populated areas. Also, cities and individuals can be encouraged to build away from a flood plain or construct buildings and structures more resistant to earthquakes and hurricanes. One of the earliest public sector applications of decision analysis

with imperfect information involved seeding hurricanes to reduce their impact.

Improvements in forecasting have helped people carry out short-term tactics, such as boarding up windows, evacuating from the area of an impending disaster, and stockpiling emergency supplies. One decision support system (but without OR models) is designed simply to convert National Weather Service forecasts into forecasts of specific emergency hazards (Subramanian and Kerpedjiev 1998). The first major application of this system was the Basin Rainfall Monitoring System, intended to predict in simplified and useful fashion the risk of flash flooding.

Once the immediate disaster is over, local, state, and national agencies must cooperate to address short-term, day-to-day needs, to create and implement stop-gap solutions such as temporary roads and bridges, and then to work to help a region rebuild its housing and industrial infrastructure (Beresford and Pettit 2009). Rebuilding should be implemented with an eye toward reducing the chance of a future similar disaster. One integrated application of OR techniques involved managing emergency repair logistics for an electric utility (Zografos et al. 1998). The integrated framework included: (1) a data management module with GIS to track service restoration by geography, (2) an information system module to monitor vehicles and communications, and (3) an analytic module to optimize the division of the region into districts and a simulation to study and improve dispatch operations.

Operations research models in this domain tend to emphasize centralized and integrated planning. This planning would include dispatching vehicles around a transportation infrastructure that has been compromised and locating facilities to assist people in need. Often, however, these massive events seriously undermine communications, making it difficult to implement a comprehensive plan. It has been proposed that a more flexible localized effort be the key element of any strategy.

See

- ▶ Crime and Justice
- ▶ Disaster Management: Planning and Logistics
- ▶ Facility Location
- ▶ Fire Safety Modeling and Applications
- ▶ Hypercube Queueing Model
- ▶ Network
- ▶ Queueing Theory
- ▶ RAND Corporation

References

Altay, N., & Green, W. G., III. (2006). OR/MS research in disaster operations management. *EJOR, 175*, 475–493.

Averill, J. D., Moore-Merrell, L., Barowy, A., Santos, R., Peacock, R., Notarianni, K. A., & Wissoker, D. (2010). *Report on residential fireground field experiments*. NIST Technical Note 1661, NIST, Washington, DC.

Beresford, A., & Pettit, S. (2009). Emergency logistics and risk mitigation in Thailand following the Asian tsunami. *International Journal of Risk Assessment and Management, 13*, 7–21.

Chaiken, J. M., & Dormont, P. (1978). A patrol car allocation model: Capabilities and algorithms. *Management Science, 24*, 1291–1300.

Chelst, K. R. (1981). Deployment of one- vs. two-officer patrol units: A comparison of travel times. *Management Science, 27*, 213–230.

Cole, H. P., Vaught, C., Wiehagen, W. J., Haley, J. V., & Brnich, M. J., Jr. (1998). Decision making during a simulated mine fire escape. *IEEE Transactions on Engineering Management, 45*, 153–161.

Donovan, G., Rideout, D. B., & Omi, P. N. (1999). *The economic efficiency of the national fire management analysis systems (NFMA) and FIREPRO*. Proceedings of the symposium of fire economics, planning and policy: Bottom lines. USDA Forest Service, PSW-GTR-173.

Erkut, E., Ingolfsson, A., & Erdo˘gan, G. (2008). Ambulance deployment for maximum survival. *Naval Research Logistics, 55*, 42–58.

Finney, M. A. (2004). FARSITE: Fire area simulator–model development and evaluation. Research Paper RMRS-RP-4 Revised. Ogden, UT: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station.

Gorr, W., Olligschlaeger, A., & Thompson, Y. (2003). Short-term forecasting of crime. *International Journal of Forecasting, 19*, 579–594.

Green, L. V., & Kolesar, P. J. (2004). Improving emergency responsiveness with management science. *Management Science, 50*, 1001–1014.

Iakovou, E. R., Ip, C. M., Douligeris, C., & Korde, A. (1996). Optimal location and capacity of emergency cleanup equipment for oil spill response. *EJOR, 96*, 72–80.

Larson, R. C. (1972). *Urban police patrol analysis*. Cambridge, MA: MIT Press.

Larson, R. C. (1974). A hypercube queuing model for facility location and redistricting in urban emergency services. *Computers and Operations Research, 1*, 67–95.

List, G. F., & Turnquist, M. A. (1998). Routing and emergency-response-team siting for high-level radioactive waste shipments. *IEEE Transactions on Engineering Management, 45*, 141–151.

Machol, R. E. (1995). Thirty years of modeling midair collisions. *Interfaces, 25*(5), 151–172.

Matarese, L. A., & Chelst, K. R. (1991). Forecasting the outcome of police/fire consolidations. *MIS Report* 23, 4, 1–22, International City Management Association.
Maxwell, M. S., Restrepo, M., Henderson, S. G., & Topaloglu, H. (2010). Approximate dynamic programming for ambulance redeployment. *INFORMS Journal on Computing, 22*, 266–281.
McDonald, P. P. (2001). *Managing police operations: Implementing the NYPD crime control model using COMPSTAT*. Wadsworth.
Morabito, R., Chiyoshi, F., & Galvão, R. D. (2008). Non-homogeneous servers in emergency medical systems: Practical applications using the hypercube queueing model. *Socio-Economic Planning Science, 42*, 255–270.
Subramanian, C., & Kerpedjiev, S. (1998). Dissemination of weather information to emergency managers; a decision support tool. *IEEE Transactions on Engineering Management, 45*, 106–114.
Walker, W. E., Chaiken, J. M., & Ignall, E. J. (Eds.). (1979). *Fire department deployment analysis*. Amsterdam: North Holland.
Wang, T., Guinet, A., Belaidi, A., & Bescombes, B. (2009). Modelling and simulation of emergency services with ARIS(TM) and Arena(TM). Case study: The emergency department of Saint Joseph and Saint Luc Hospital. *Production Planning & Control, 20*, 484–495.
Yi, W., & Özdamar, L. (2007). A dynamic logistics coordination model for evacuation and support in disaster response activities. *EJOR, 179*, 1177–1193.
Zografos, K. G., Douligeris, C., & Tsoumpas, P. (1998). An integrated framework for managing emergency-response logistics: The case of the electric utility companies. *IEEE Transactions on Engineering Management, 45*, 115–126.

EMS

Emergency medical services.

See

► Emergency Services

Engineering Applications

Reuven R. Levary
Saint Louis University, St. Louis, MO, USA

Introduction

Different engineering disciplines have unique characteristics and problems. Operations research (OR) is a field made up of many conceptually different methods and algorithms, each suited to a specific environment. Thus, some specific OR methods and algorithms are better suited to the solution of certain types of engineering problems than others. Applications of OR methods and algorithms to problems in those engineering disciplines best suited to such applications are described below.

Communication Systems Engineering

Operations research methods are widely used in various aspects of communication system planning, design, manufacturing, and implementation (Daigle and Langford 1988). Examples that illustrate the diversity of applications of OR methods to communication systems include a filter design which uses game theory (Kazakos 1983) and the use of Markov chains to model a synchronous time division multiplexing frame synchronization algorithm (Liu and Hammond 1980). The use of OR methods in the design and analysis of computer communication networks has attracted significant attention in the literature, since the design and analysis of these networks depends almost entirely on OR methods. Sauer and McNair (1983) used simulation to analyze computer communication systems. Queueing theory and stochastic processes were used by Hayes (1984) and by Stuck and Arthurs (1985) to analyze such systems. Marcus and Papatoni-Kazakos (1983) analyzed a multi-access protocols problem via dynamic programming. Self-healing communication networks that allow re-routing of demands through switching processes at designated nodes have been designed using a node-path linear-programming approximation to the multi-commodity network formulation (Saniee 1996). Luna et al. (2008) used a grid-based generic algorithm to solve automatic frequency planning problems. These problems occur in global systems for mobile communications networks. Several evolutionary algorithms applied to a class of communication network design problems were evaluated by Nesmachnow et al. (2007). A brief description of two of the most important problems in the design of contemporary communication systems follows.

Statistical Multiplexing — The design of a communication system based on high-speed links

that transmit information generated by many users generally has an economic advantage over a design based on separate links between pairs of users (Daigle and Langford 1988). Multiplexing and concentration are terms used to characterize methods of assigning the capacity of a link among many users. Multiplexing represents the case where the capacity of a link is divided into frequency bands and each user is assigned to a specific band. Concentration represents the case where the capacity of a link is smaller than that needed to accommodate simultaneous requests by all users, and, thus, a line sharing approach is necessary. In that case, a queueing system is used to control the traffic of user requests for communication.

Statistical multiplexing, also called asynchronous time-division multiplexing, is an approach for link sharing that combines characteristics of both multiplexing and concentration (Daigle and Langford 1988). The users' messages need to be transmitted over a communication link. While a users' message is divided in some systems into packets, the message remains intact in others. When statistical multiplexing is implemented, the users' messages are kept in a queue before being transmitted over the communication link. Depending on the volume of messages, queueing delays may occur. The delays obviously present an inconvenience to the users, and should be considered in evaluating the appropriateness of statistical multiplexing for a particular environment.

Statistical multiplexing systems have been designed and analyzed using queueing theory, Markov-chain models and renewal theory. Poisson arrivals are assumed in many cases to simplify the analysis. The Poisson-arrivals-see-time-averages (PASTA) property is often used to obtain analytical results (e.g., see Wolff 1982). The Chapman-Kolmogorov equation is generally used to analyze discrete-time systems, while equilibrium balance equations are frequently used to analyze continuous-time systems (Daigle and Langford 1988).

Topological Design of Local Distribution Networks — The problem of designing the topological aspects of local distributed networks is composed of five subproblems (Gavish 1982). The subproblems include determination of: the number of concentrators; the location of concentrators; the method of connecting the concentrators to the switch; the assignment of terminals to concentrators; and the interconnection between terminals and concentrators. The subproblems can be solved simultaneously (Daigle and Langford 1988).

In the terminal layout problem, a tree structure is sought that minimizes the total cost of connecting all the terminals to the packet switch, that is, terminal or node 1 (Daigle and Langford 1988). It is assumed that the number of terminals or nodes is $n - 1$, and are numbered 2 through n. The cost c_{ij} of connecting terminal i to terminal j for $1 \leq j \leq n$ and $1 \leq i \leq n$ is assumed to be given. It is also assumed that c_{ij} is infinite for $i = j$. It is further assumed that the measure of traffic associated with each terminal identifies the communication flow between the terminal and the switch. Daigle and Langford (1988) discussed solution methodologies that have been applied under the following conditions: when trees are unconstrained, degree-constrained, capacity-constrained or both capacity and degree constrained.

Structural Engineering

Mathematical programming (MP) has been extensively used to determine the optimal structural design of various engineering systems. Applications of MP to the design of civil engineering structures are surveyed below. MP has also been used to design structures undergoing free or forced vibrations. An overview of this subject is also provided below.

Designing Civil Engineering Structures — While many feasible designs that satisfy functional requirements often exist, a trial-and-error procedure may be needed to obtain the optimal solution (Kirsch 1988). The efficiency of the structural design process can be improved by automating portions of the design process. Such automation is possible because of the progress made in computer technology, structural analysis, and optimization methods (Atrek et al. 1984; Soares 1986).

Kirsch (1988) suggested that an automated design process be considered an iterative process. The iterative cycle, in this process, is composed of two main steps: (1) analysis of the current structural design, and (2) redesign, that is, modification of the design by optimizing an objective function subject to the pertinent constraints.

Some of the parameters characterizing a structure are fixed during the automated design process.

The parameters that are not fixed are the design variables. The design variables represent the following characteristics of the structure:

1. Physical properties of the materials
2. Topology
3. Configuration or geometric layout
4. Cross-sectional dimensions.

The design of the structure is defined once the value of the design variables are obtained. From analyzing the structure, one can determine the forces and displacements. Structural optimization problems often deal with cross-sectional design variables. These variables are usually assumed to be continuous despite the fact that they actually can obtain only discrete values (Kirsch 1988).

Truss structure is the subject of most structural optimization studies. An illustrative example of such optimization was given by Kirsch (1988). Grillage is a flexural system composed of beam elements. Moses and Onoda (1969) illustrated that grillages introduce significant design difficulties, in addition to those usually encountered in the optimal design of any indeterminate structure. The design difficulties typical to grillages include multiple local optima and internal forces sensitive to changes in the design variables. Kirsch (1988) provided an example that illustrates the possibility of multiple optimal topologies in flexural systems. The optimal design of reinforced-concrete structures was formulated by Kirsch (1988) as a four-level MP problem. Krishnamoorthy and Murno (1973) formulated a linear-programming (LP) model to satisfy compatibility and limited ductility, equilibrium and serviceability conditions as constraints. The model has a linearized objective function for the total volume of steel reinforcement. Kirsch (1988) formulated a typical prestressed system design problem using MP.

Designing Optimal Vibrating Structures — The objective functions used in designing structures that are subject to dynamic loads include both maximization of eigen-frequencies and minimization of dynamic deflection and/or stress (Adali 1988). The resonance range of structures undergoing free vibrations are increased by maximizing either the fundamental or higher-order frequencies. Dynamic deflections and/or stresses are minimized for structures undergoing forced vibration to improve the structure's service performance. Structures may be subject to both free and forced vibrations during operational life. For this reason, lower bounds are generally imposed on the eigen-frequencies and upper bounds are imposed on the dynamic deflection and stresses (Adali 1988). In this case, the resulting design problem is a constrained optimization problem. Such problems are routinely solved using penalty function methods. The problems, however, can also be formulated as multicriteria optimization problems.

Beam design problems for the case of continuously vibrating structures were formulated by Adali (1988) using MP. The objective function in such problems is maximization of the fundamental frequency. The use of MP in formulating beam design problems for discrete structures like frames and trusses is fairly obvious.

Some structures may be subject to forced vibrations caused by environmental forces such as wind, waves, and earthquakes. The objective in designing such structures is to minimize the deflection and/or stresses caused by dynamic loads (Adali 1988). Structures under dynamic loads can be effectively designed using MP (Khot et al. 1986).

The penalty-function method is an effective optimization method used to design structures when there are constraints on stress, deflection, and maximum frequency. Several penalty-function techniques useful in designing structures having constrained beams and frames are discussed by Adali (1988).

MP has been successfully used to optimally design composite, lightweight structures like those typically used in the aerospace industry. After studying the designs of laminated plates and shells undergoing free vibrations, Adali (1988) demonstrated how to use MP in designing such optimal composite structures.

Results of sensitivity analysis were used by Hsieh and Arora (1986) for satisfying state-variable constraints. Sensitivity results for eigen-frequencies were obtained for frames and composite plates. Their use was illustrated by Adali (1988) via specific examples. Evolution strategies were applied by Hasanceb (2007) to optimize the design of truss bridges. This optimization problem involves identification of the bridge's shape and topology configurations, as well as sizing of the structural members for minimum weight. Cagdas and Adali (2007) developed optimum designs for

clamped-clamped columns under concentrated and distributed axial loads. The design objective was the maximization of the buckling load subject to volume and stress constraints. Barreled cylinders and domes of generalized elliptical profile were optimized by Blachut and Smith (2007) for their buckling resistance when loaded by static external pressure. The optimum shells were found using either a static or adaptive tabu search method.

Chemical Engineering

Many applications of OR techniques can be found in the design and operation of chemical plants. These application include refinery planning, equipment design optimization, and optimization of equipment networks. Solutions to the complex nonlinear optimization problems typical of chemical engineering have been gradually improved due to the continuous progress made in nonlinear programming (NLP), mixed integer-linear programming (MILP), scheduling, and simulation techniques, as well as to the improvement made in related software (Biegler et al. 1988).

Many chemical engineering optimization problems are much too large and complex to be solved by direct application of generic OR algorithms (Biegler et al. 1988). By examining both the unique physical characteristics of each system and the unique structure of each optimization problem, however, it is possible to reduce very large and often highly combinatorial problems to manageable size (Biegler et al. 1988).

Biegler et al. (1988) demonstrated that when addressing the optimal design/operation problem of continuous processes, an understanding of model construction and execution must precede algorithm selection and tuning. Modeling is typically done using process simulation. By taking advantage of the unique composition of the system being modeled, one is in a better position to successfully optimize system design. The NLP applications from optimizing design/operation problems dealing with continuous processes are discussed by Biegler et al. (1988), and illustrate the advantages of both the unique analysis of each problem and the intelligent application of generic optimization algorithms.

The combinatorially complex problem for synthesis of process structures was discussed by Biegler et al. (1988). Problems related to utility systems, energy recovery systems, and total processing systems were considered. The problems related to utility systems can be formulated as MILP. The binary variables in such problems are the structural choice variables. It is possible to significantly reduce the number of binary variables by taking advantage of the uniqueness of constraints and problem structure (Biegler et al. 1988). A reduction in the number of binary variables makes it possible to use software for MILP. Biegler et al. (1988) showed that a complex mixed integer-nonlinear programming (MINLP) can be broken down into a series of subproblems having well-structured solutions. This decomposition is accomplished through use of thermodynamic insights and the system's structural features. Beigler et al. (1988) applied these ideas to a total processing system that consisted of a utility plant, a heat-recovery network, and the chemical plant itself.

The design and scheduling of noncontinuous processes generally involves multi-products. Noncontinuous processes involving multi-products are typical to some high-value-added chemical and biochemical products. Problems in these areas are discussed by Biegler et al. (1988). Two types of facilities were considered: the network flow shop configuration and the multi-purpose plant. These facilities can be modeled using MINLP. Because of the complexity of the models, approximation and decomposition methods are necessary to solve them (Biegler et al. 1988). Design problems in a multipurpose plant involve some aspects of scheduling and therefore add to the complexity. While some portions of these design problems can be formulated and solved as MILP scheduling and sequencing problems, most portions require the development of specialized approximation algorithms. The application of combined continuous-discrete simulation concepts to modeling and analysis of continuous batch chemical processes was presented by Biegler et al. (1988).

Churi and Achenie (1997) developed an MINLP model for single component refrigerant design . The design objective is to build a refrigerant molecule that has desired physical properties and performance characteristics. Scheduling of chemical batches using stochastic integer programming was reported by Urselmann et al. (2007). The scheduling algorithm combined evolutionary algorithm and mixed-integer programming into two-stage stochastic programming problem.

Aerospace Engineering

The design of an aerospace vehicle is a multidisciplinary project that involves the following disciplines: control theory, solid mechanics, fluid mechanics, dynamics, electronics, and computer engineering. Optimization theory has a significant role in aerospace engineering since it is extremely important to achieve optimal design in every aerospace project.

Aerospace Structures — The motivation for applications of optimization methods to the design of aerospace structures is the need for efficient light-weight structures. The decision variables of a typical problem involved in optimizing an aerospace structure are the structural member dimensions. The objective of such a problem is to minimize structural weight subject to constraints on limit stresses in members-nodal displacements, on natural frequencies of the structure, and on the stability of the eigenvalues (Hajela 1988).

One way of formulating and solving optimization problems related to aerospace structures is to first define an optimality criterion and then attempt to satisfy the criterion via a numerical algorithm (Berke and Khot 1974). This approach, however, is very restrictive as the definition of optimality criterion is problem dependent (Hajela 1988).

The application of NLP methods to the design of aerospace structures is more flexible than application of optimality criterion. NLP methods, however, are computationally inefficient for design problems involving large numbers of variables and constraints (Hajela 1988). A more practical way of achieving an optimum aerospace structural design is by using numerical optimization.

Sobieski (1988) showed how numerical optimization could be used in aerospace structural design. He also described the role of decision variables, objective function, constraints, and the design space in numerical optimization. Sobieski (1988) illustrated how a search procedure composed of modules (e.g., search algorithms, approximate analysis, and sensitivity analysis) could improve the design of aerospace structures. He also elaborated on several ways of combining modules into an effective comprehensive optimization procedure. Sobieski (1988) described ways of dealing with the unique difficulties involved in large-scale aerospace structural design problems.

Aerodynamic Characteristics — The geometry of the airfoil (that is, wing cross-section) affects configuration lift and drag characteristics. Design variables of an airfoil include thickness, distribution over the chord-wise dimension, radius of curvature of the airfoil's nose, and the camber or arch of the airfoil chord (Hajela 1988). The values of these variables determine the lift and drag of the aerospace vehicle given the characteristics of the flow in which the airfoil is expected to operate. Flow can range from low subsonic speeds to high supersonic speeds. Atmospheric re-entry vehicles can experience hypersonic flow. The entire spectrum of airfoil shapes and their respective lift and drag for low speed flow have been documented (Hajela 1988). Simulated flow conditions in a wind tunnel can be useful in designing new shapes for airfoils that must operate under changing flight conditions. The use of wind tunnels is expensive, however, because it calls for the development of a model and because delays are inevitable in scheduling the use of a wind tunnel. Furthermore, only a limited number of design alternatives can realistically be examined in a wind tunnel.

Airfoil shapes can be designed using numerical computation techniques for fluid flow such as the finite-difference method or the finite-element method (Hajela 1988). Aidala (1988) developed the concept of obtaining optimum airfoil shape by using a weighted sum of standard airfoil shapes. The weight constants, in that case, are the decision variables. Their values must be selected to optimize desired aerodynamic characteristics. Aidala (1988) also elaborated on another design approach in which airfoil geometry was refined to obtain desired pressure distribution over the airfoil. Variable-complexity methods were applied by Thokala et al. (2007) to aerodynamic shape design problems. The objective was to reduce the total computational cost of the optimization process. Transonic airfoil design problems considering inviscid and viscous flow solvers were solved by Shahrokhi and Jahangirian (2010). They used a surrogate assisted evolutionary optimization method to solve the problems.

The aerodynamic design of an aircraft must allow for a mechanically and aero-elastically acceptable wing and must also satisfy several different performance requirements (Aidala 1988). The Lagrange multipliers method can be used to achieve

an optimal aerodynamic design when linear theories are sufficient to characterize the design problem (Aidala 1988). The typical approach in the more general nonlinear case has been to integrate analysis codes with a general nonlinear optimization method (Aidala 1988). A general nonlinear optimization approach to aerodynamic design is the most frequently used in such optimization problems and described by Aidala (1988). To obtain computationally efficient results, however, numerical optimization must be used.

Aerospace Vehicle Performance — A unique characteristic of aerospace vehicles is that there are no external forces like buoyant (static) lift or ground reaction forces that can support a vehicle (Kidwell 1988). Thus, an aerospace vehicle must be designed to generate the forces needed for longitudinal, lateral, and vertical translation (Kidwell 1988). The forces must be proportional to the vehicle's mass and to the requirements for acceleration. Growth factor (Kidwell 1988) is a measure of the overall increase in the aircraft gross weight caused by adding one pound of weight (from payload, fuel, structural weight, etc.). The growth factor indicates the compounding effect of weight. Any increase in the requirement for weight necessitates increase in one or more of the following: wing area to provide the necessary additional lift; engine size to compensate for the increase in drag; and fuel requirements that are a consequence of the greater engine thrust needed to counter drag (Kidwell 1988). Every component of an aerospace vehicle must be designed with minimal weight in mind, as any change in the weight of a particular component has a significant effect on the performance of the entire vehicle. Kidwell (1988) discussed several methods useful in designing aircraft that meet desired performance criteria. He also showed how numerical optimization could be use to achieve an optimal aircraft design having acceptable computational effort.

Optimal locations of dual trailing-edge flaps was determined by Viswamurthy and Ganguli (2007). The objective was to achieve minimum hub vibration levels in helicopters. Chang, et al. (2008) developed a deductive top-down estimation methodology for assessing the reliability of aircrafts propulsion system.

Electrical Circuit Design

The filter design problem was formulated by Lasdon and Waren (1966) as a nonlinear program. Optimal filters were obtained using an interior penalty function in a problem formulation having a minimax type of objective function. Lasdon, Waren et al. used NLP in designing optimal antenna arrays, cascade crystal-realizable lattice filters, and optimal filters (Waren et al. 1977).

The use of optimization methods in the design of filters and in the modeling of active electrical devices is reviewed by Temes and Calaham (1967). Temes and Zai (1969) designed active equalizers based on least pth approximation. The theory of generalized least pth approximation was developed by Bandler and Charalambous (1972). This approach allows one to use any value of p while solving several minimax problems.

Madsen et al. (1975) developed a minimax electrical network optimization algorithm based on successive linear approximation of a nonlinear objective function. Applications of their algorithm include the design of transmission-line transformers and the design of microwave filters (Bandler and Rizk 1979).

Optimization methods have been routinely used in the design of digital filters (Bandler and Rizk 1979). Optimization methods such as LP, NLP and integer programming (IP) were used in the design of a nonrecursive filter based on minimax responses (Helms 1971). Steiglitz (1970) showed how optimization methods could be used to design recursive digital filters. Recursive digital filters with optimum word length were designed by Bandler et al. (1975).

Optimization methods have been used extensively in the design and operation of power networks (Bandler and Rizk 1979). The problem of minimizing the cost of fuel for thermal power plants, for example, is solved on-line every few minutes. The solution to this problem is periodically used to adjust the output of the power plant (Bandler and Rizk 1979). The problems of optimizing hydroelectric systems are generally much larger than those of thermal power plants. Gagnon et al. (1974) developed an NLP model for solving such problems.

Computer System Design

Several different aspects of computer system design are described below.

Selecting an Optimum Network Configuration — To optimize the configuration of a computer system, the system can be modeled as a closed queueing network. Consider, for example, the problem of configuring a batch-oriented, nonpaged, multiprogramming system (Trivedi and Kinicki 1980). The central server queueing network given in Sarma et al. (1988) can be used to model such systems. The system characteristics that are considered include variables related to both the size of the main memory and the speed of a fixed number of secondary storage devices. $m + 1$ service facilities (i.e., facility 0,. . ., facility m) in a closed queueing system represent active resources of the system such as the CPU and I/O channels. Each service facility is composed of a single server when the ith facility has a processing capability of b_i work units per unit of time. The degree of multiprogramming is denoted by n. The network is composed of n stochastically equivalent programs that switch back and forth between service at the CPU and service at the I/O channels. After processing by the CPU is complete, service from the ith I/O channel is requested by a program with probability p_i ($i = 1, 2, \ldots, m$).

The device speed vector $\boldsymbol{b} = (b_0, b_1, \ldots, b_m)$ in this design example (see Sarma et al. 1988) is considered to be the decision variable. The branching probabilities p_i ($i = 1, \ldots, m$) are assumed to be parameters. The design problem is formulated as follows:

$$\begin{aligned} &\text{Maximize } T(\boldsymbol{b}) \\ &\text{subject to } F(\boldsymbol{b}) + M(n) \le B \\ &\qquad\qquad b_i \ge 0, \quad i = 0,\ 1,\ 2,\ \ldots,\ m \end{aligned}$$

where $T(\boldsymbol{b})$ is the system throughput; $F(\boldsymbol{b})$ is the cost of $m + 1$ service facilities; $M(n)$ is the cost of the main memory; and B is the budget available for the purchase of the necessary hardware. For the method of solving such a problem, see Sarma et al. (1988).

Design of Fault-Tolerant Computer Systems — The design optimization problem under consideration includes measures of system dependability, performance and cost. More specifically, a shared and replicated resources architecture designed for high reliability applications is considered. Real-time control typifies an application that necessitates high reliability. The modular multi-processor system (MMPS) (Pedar and Sarma 1983) is an example of this architecture.

One characteristic of the MMPS is that the system's computational capability, as well as the corresponding memory contention, change when the processor and the memory unit fail (Sarma et al. 1988). Trivedi (1982) showed that system reliability can be analyzed using Markov models if the failure rates of the processors and of the memory modules are assumed to be constant. The state, in this case, represents the number of operational processors and the number of operational modules at a given time. It is assumed that the program memory is divided into several stages. The decision variables include the number of stages, the number of modules in each stage, the replication factor in a give stage, and the number of processors. The problem faced by the computer design engineer is to minimize the total cost of the system subject to constraints on unreliability of the system's computations.

Task Allocation in Distributed Computer Systems — Given a program composed of several modules, the problem under consideration is the allocation of the modules to the individual processing elements of a multiprocessor system. This problem can be formulated as a constrained optimization problem (Sarma et al. 1988). The objective of such problems is to minimize the cost function that depends on the specific allocations. The constraints include restrictions on the speed of the processors and the limits on memory capacity. Allocation problems are generally computationally intensive (i.e., NP-complete). The three most widely used approaches for solving these problems are graph theoretic, integer programming, and approximation algorithms. Graph theoretic algorithms are practical in solving only problems involving a small number of processors. While integer programming is more widely used in solving allocation problems, it is difficult to predict the amount of computations necessary to obtain the needed accuracy. Approximation algorithms are generally considered the most practical method for solving allocation problems since they can provide acceptable solutions at low computational cost

(Sarma et al. 1988). OR methods have been applied to other aspects of computer systems design. One example is the use of optimization algorithms for motion recovery in computer vision (Lee and Park 2008).

Mechanical Engineering

OR methods have been successfully applied to various mechanical engineering design problems. A few examples follow.

Engine design–Aittokoski and Miettinen (2008) applied a simulation-based optimization to the design of a two-stroke combustion engine. The optimization approach involves interactive multiobjective optimization. A hybrid evolutionary algorithm consisting of a generic algorithm and particle swarm optimization was applied by Jeong et al. (2008) to the geometry design of diesel engine combustion chamber. The new design resulted in reducing exhaust emissions.

Yixian et al. (2009) developed a topology optimization approach for the design of compliant actuators using mesh-free methods in which the thermo-mechanical multi-physics modeling and geometrically non-linear analysis were included. The optimization problem was formulated as a nonlinear programming-problem to which a sequential convex programming method was applied. The mechanical structure of a five-bar parallel robot was designed by Villarreal-Cervantes et al. (2010) using a constraint-handling differential evolution algorithm to solve the nonlinear dynamic optimization problem.

See

- ▶ Computer Science and Operations Research Interfaces
- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Markov Processes
- ▶ Networks of Queues
- ▶ Nonlinear Programming
- ▶ PASTA
- ▶ Queueing Theory
- ▶ Telecommunication Networks

References

Adali, S. (1988). Optimal design of vibrating structures by mathematical programming. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 201–225). New York: North-Holland.

Aidala, P. V. (1988). Optimization theory for aerodynamic design. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 263–275). New York: North Holland.

Aittokoski, T., & Miettinen, K. (2008). Cost effective simulation-based multiobjective optimization in the performance of a internal combustion engine. *Engineering Optimization, 40*(7), 593–612.

Atrek, E. R., Gallagher, R. H., Ragsdell, K. M., & Zienkiewicz, O. C. (Eds.). (1984). *New directions in optimum structural design*. New York: John Wiley & Sons.

Bandler, J. W., & Charalambous, C. (1972). Theory of generalized least *p*th approximation. *IEEE Transactions on Circuit Theory, CT-19*, 287–289.

Bandler, J. W., & Rizk, M. R. M. (1979). Optimization of electrical circuits. In M. Avriel & R. M. S. Dembe (Eds.), *Engineering optimization, mathematical programming study 11* (pp. 1–64). Amsterdam: North-Holland Publishing.

Bandler, J. W., Bardakjian, B. L., & Chen, J. H. K. (1975). Design of recursive digital filters with optimized word length coefficients. *Computer Aided Design, 7*, 151–156.

Berke, L., & Khot, N. S. (1974). Use of optimality criteria methods for large scale systems, AGARD Lecture series, no. 70. *Structural Optimization*, October 1974.

Biegler, L. T., Grossmann, I. E., & Reklaitis, G. V. (1988). Application of operations research techniques in chemical engineering. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 317–468). New York: North-Holland.

Blachut, J., & Smith, P. (2007). Tabu search optimization of externally pressurized barrels and domes. *Engineering Optimization, 39*(8), 899–918.

Cagdas, I. U., & Adali, S. (2007). Optimization of clamped columns under distributed axial load and subject to stress constraints. *Engineering Optimization, 39*(4), 453–469.

Chang, K. H., Cheng, C. H., & Chang, Y. C. (2008). Reliability assessment of an aircraft propulsion system using IFS and OWA trees. *Engineering Optimization, 40*(10), 907–921.

Churi, N., & Achenie, L. E. K. (1997). On the use of a mixed integer non-linear programming model for refrigerant design. *International Transactions in Operational Research, 4*(1), 45–54.

Daigle, J. N., & Langford, J. D. (1988). Operations research methods in the communication fields. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 644–682). New York: North-Holland.

Gagnon, C. R., Hicks, R. H., Jacoby, S. L. S., & Kowalik, J. S. (1974). A nonlinear programming approach to a very large hydroelectric system optimization. *Mathematical Programming, 6*, 28–41.

Gavish, B. (1982). Topological design of centralized computer networks-formulations and algorithms. *Networks, 12*, 355–377.

Hajela, P. (1988). Optimization applications in aerospace engineering. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 252–262). New York: North-Holland.

Hasanceb, O. (2007). Optimization of truss bridges within a specified design domain using evolution strategies. *Engineering Optimization, 39*(6), 737–756.

Hayes, J. F. (1984). *Modeling and of analysis computer communication networks*. New York: Plenum Press.

Helms, H. D. (1971). Digital filters with equiripple or minimax responses. *IEEE Transactions on Audio Electroacoustics, AU-19*, 87–94.

Hsieh, C. C., & Arora, J. S. (1986). Algorithms for pointwise state variable constraints in structural optimization. *Computers & Structures, 22*, 225–238.

Jeong, S., Obayashi, S., & Minemura, Y. (2008). Application of hybrid evolutionary algorithms to low exhaust emission diesel engine design. *Engineering Optimization, 40*(1), 1–16.

Kazakos, D. (1983). New results on robust quantizations. *IEEE Transactions on Communications, COM-31*(8), 965–974.

Khot, N. S., Venkayya, V. B., & Eastep, F. E. (1986). Optimal structural modifications to enhance the active vibration control of flexible structures. *American Institute of Aeronautics and Astronautics Journal, 24*, 1368–1374.

Kidwell, G. H. (1988). Aircraft design-performance optimization. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 276–293). New York: North-Holland.

Kirsch, U. (1988). Applications of mathematical programming to the design of civil engineering structures. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 174–200). New York: North-Holland.

Krishnamoorthy, C. S., & Munro, J. (1973). Linear program for optimal design of reinforced concrete frames. *International Association for Bridge and Structural Engineering, 33*, 119–141.

Lasdon, L. S., & Waren, A. D. (1966). Optimal design of filters with rounded, lossy elements. *IEEE Transactions on Circuit Theory, CT-13*, 175–187.

Lee, S., & Park, F. C. (2008). Cyclic optimization algorithms for simultaneous structure and motion recovery in computer vision. *Engineering Optimization, 40*(5), 403–419.

Liu, S. S., & Hammond, J. L., Jr. (1980). A method for modeling and analysis of reframing performance of multilevel synchronous time division multiplex hierarchies. *IEEE Transactions on Communication, COM-28*(8), 1219–1228.

Luna, F., Nebro, A. J., Alba, E., & Durillo, J. J. (2008). Solving large-scale real- world telecommunication problems using a grid-based genetic algorithm. *Engineering Optimization, 40*(11), 1067–1084.

Madsen, K., Nielson, O., Schjaer-Jacobsen, H., & Thrane, L. (1975). Efficient minimax design of networks without using derivatives. *IEEE Transactions on Microwave Theory and Techniques, MTT-23*, 803–809.

Marcus, G. D., & Papatoni-Kazakos, P. (1983). Dynamic scheduling protocols for a multi-access channel. *IEEE Transactions on Communications, COM-31*(9), 1046–1055.

Moses, F., & Onoda, S. (1969). Minimum weight design of structures with application to elastic grillages. *International Journal for Numerical Methods in Engineering, 1*, 311–331.

Nesmachnow, S., Cancela, H., & Alba, E. (2007). Evolutionary algorithms applied to reliable communications network design. *Engineering Optimization, 39*(7), 831–855.

Pedar, A., & Sarma, V. V. S. (1983). Architecture optimization of aerospace computing systems. *IEEE Transactions on Computers, C-32*, 911–922.

Saniee, I. (1996). Optimal routing designs in self-healing communication networks. *International Transactions in Operational Research, 3*(2), 187–195.

Sarma, V. V. S., Trivedi, K. S., & Reibman, A. L. (1988). Optimization methods in computer systems design. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 683–705). New York: North-Holland.

Sauer, C. H., & McNair, E. A. (1983). *Simulation of computer communication systems*. Englewood Cliffs, NJ: Prentice-Hall.

Shahrokhi, A., & Jahangirian, A. (2010). A surrogate assisted evolutionary optimization method with application to the transonic airfoil design. *Engineering Optimization, 42*(6), 497–515.

Soares, M. C. A. (Ed.). (1986). NATO advance study Institute. *Proceedings on Computer Aided Optimal Design*, Troia, Portugal.

Sobieski, J. S. (1988). Optimization in aerospace structures. In R. R. Levary (Ed.), *Engineering design: Better results through operations research methods* (pp. 294–316). New York: North-Holland.

Steiglitz, K. (1970). Computer-aided design of recursive digital filters. *IEEE Transactions on Audio Electroacoustics, AU-18*, 123–129.

Stuck, B. W., & Arthurs, E. (1985). *A computer communications network performance analysis primer*. Englewood Cliffs, NJ: Prentice-Hall.

Temes, G. C., & Calaham, D. A. (1967). Computer-aided network optimization-the state-of-the art. *Proceedings of the IEEE, 55*, 1832–1863.

Temes, G. C., & Zai, D. Y. F. (1969). Least *p*th approximation. *IEEE Transactions on Circuit Theory, CT-16*, 235–237.

Thokala, P., Joaquim, R. R., & Martins, A. (2007). Variable-complexity optimization applied to airfoil design. *Engineering Optimization, 39*(3), 271–286.

Trivedi, K. S. (1982). *Probability and statistics with reliability, queueing and computer science, applications*. Englewood Cliffs, NJ: Prentice-Hall.

Trivedi, K. S., & Kinicki, R. E. (1980). A model for computer configuration design. *IEEE Computer, 13*, 47–54.

Urselmann, M., Emmerich, M. T. M., Till, J., Sand, G., & Engell, S. (2007). Design of problem-specific evolutionary algorithm/ mixed- integer programming hybrids: Two-stage stochastic integer programming applied to chemical batch scheduling. *Engineering Optimization, 39*(5), 529–549.

Villarreal-Cervantes, M. G., Cruz-Villar, C. A., Alvarez-Gallegos, J., & Portilla-Flores, E. A. (2010). Differential evolution techniques for the structure-control design of a five-bar parallel robot. *Engineering Optimization, 42*(6), 535–565.

Viswamurthy, S. R., & Ganguli, R. (2007). Optimal placement of trailing-edge flaps for helicopter vibration reduction using response surface methods. *Engineering Design, 39*(2), 185–202.

Waren, A. D., Lasdon, L. S., Stotts, L. B., & McCall, D. C. (1977). Recent developments in nonlinear optimization and

their use in engineering design. In A. Wexler (Ed.), *Large engineering systems*. Oxford, England: Pergamon.
Wolff, R. W. (1982). Poisson arrivals see time averages. *Operations Research, 30*, 223–231.
Yixian, D., Zhen, L., Qihua, T., & Liping, C. (2009). Topology optimization for thermo-mechanical compliant actuators using mesh-free methods. *Engineering Optimization, 41*(8), 753–772.

Entering Variable

The non-basic variable chosen to become basic in an iteration of the simplex or similar linear-programming algorithm.

See

► Simplex Method (Algorithm)

Environmental Systems Analysis

Charles ReVelle
The Johns Hopkins University, Baltimore, MD, USA

Introduction

Within a decade after the emergence of operations research at the end of World War II, civil and environmental engineers were already adapting the remarkable mathematical tools that had evolved in the defense sector during the war. They quickly applied these tools to the solution of important societal problems relating to environmental protection. Applications of operations research to urban and regional water management began in the late 1950s primarily under the leadership of Lynn and Charnes. Solid wastes management using the tools of OR began in the mid-1960s led by Liebman and engineers at Berkeley. The development of air pollution management models followed not long after. Parallel to these engineering-based investigations of environmental issues were applications in forestry/timber management/recreation as well as game and fisheries management. The three engineering-based areas of environmental OR application will be reviewed.

Urban Water Management

First, in historical sequence of environmental applications of operations research, was urban and regional water management. Urban and regional water management encompasses many activities, some of which have been approached with OR tools and others which have seen very little application activity. Water resources management, a parallel activity to urban and regional water management, focuses on the operation of reservoirs and systems of reservoirs for purposes of water supply, recreation, flood control, irrigation, hydropower, and navigation. It also deals with aquifer management, conjunctive use of ground and surface waters, and interbasin transfers. The activities of urban and regional water management, in contrast, are principally concerned with the local delivery of water, with treatment of water, with disposal of wastewater, and with the quality of receiving waters, although quantity and quality intersect in a number of problem settings. Water resources management is discussed in this encyclopedia under its own heading.

Here, water is followed (1) out of the reservoir to the water treatment plant that produces drinking water, (2) through the distribution system to the consumer, (3) from the consumer through the sewer system (4) to the wastewater treatment plant (sewage treatment plant), and (5) into the receiving water body, where the pollution content of the treated wastewaters of many communities interact with stream dissolved oxygen resources.

The water treatment plant is designed to produce drinking water that is free of disease-causing bacteria, viruses and protozoa. The water should be attractive (clear) and palatable with little in the way of objectionable tastes, odors, or color. The processes in a typical water treatment plant are designed to achieve these criteria. The design and arrangement of the component processes are probably susceptible to a cost optimization in which constraints are placed on the final concentrations of various contaminants. The design would constitute the first stage of applications of systems analysis to urban water management. However, very little in the way of OR/systems analysis has been done in water treatment plant design.

From the water treatment plant, the water enters a distribution system for delivery to customers. In this second stage of systems applications, linear programming and nonlinear programming have been applied to the design of water distribution systems although the nonlinearity in the equations has generally been approached by iterative application of linear programming to approximate an unknown multiplicative term. Decisions include which links of the system to build, the diameters of the pipes, the water flows in each link of the system, and the pressure heads at junctions of the system. A series of papers have appeared on this topic beginning in the late 1960s. A major difficulty in this design problem is the tradeoff between cost and redundancy (needed for reliability) and the lack of a good operational measure of redundancy. A unique article that explicitly compares a number of the approaches to this problem was jointly prepared by many of the researchers in the area of pipe network optimization and appears under the title "Battle of the Network Models," (Walski et al. 1987).

Residential consumers, as well as industry and commerce, receive the water from the distribution system, use it and abuse it for washing, bathing, lawn watering, irrigation and manufacturing processes. As a consequence of the use, the quality of the water is degraded, principally by the presence of organic contaminants, but also with micro-organisms and inorganic chemicals. Treatment at a sewage treatment plant is required to restore the water to a level of quality that will not impair the water in the receiving body.

To reach the sewage treatment plant, the wastes from residences, commerce and industry enter a wastewater collection system, the sewer system, which conveys them to the plant. The design of the sewer system represents a third stage of application of systems analysis to urban water management. A number of optimization models have been built to determine which links to build, diameters of individual sewer lines, the depth at which each line is placed and the slopes of each line. A representative work in this field is that of Walters (1985).

The sewers transport the wastes from their origins to the treatment plant which itself requires design. The design of the wastewater treatment plant represents a fourth stage of application of systems analysis to urban water management. The treatment plant typically consists of ordinary biological processes which remove organic wastes that are in solid form as well as organic wastes that are dissolved in the waste stream. Organic wastes are removed from wastewater because they will otherwise be degraded by microbes when they reach a lake or river and that biodegradation would remove dissolved oxygen from the water. Because fish and other stream aquatic organisms require adequate levels of oxygen to survive, it is imperative that sufficient amounts of organics be removed from wastewater to protect the dissolved oxygen resource that sustains the fish and other aquatic life. Depending on the requirement of the receiving water, the treatment plant could also include physical-chemical processes to remove not only nitrates and phosphates, but also the small fraction of organics which is resistant to biological treatment. The design of treatment processes using optimization methodology was begun in the early 1970s. A research paper on the subject which usefully refers to past works is that of Tang et al. (1987).

From the treatment plant (where the removal of organics takes place), the restored wastewater may enter a river or lake where it mixes with receiving waters. The concentration of organic wastes ultimately discharged into the receiving waters will decrease as the degree of treatment/level of removal at the wastewater treatment plant increases. Without much treatment, the amount of dissolved oxygen in the lake or river consumed by oxidation of the organic wastes will be relatively large, making the water environment inhospitable to fish and other desirable aquatic organisms. Predictive models for the removal of the oxygen resource and biological decay of organic wastes were first developed in the 1910s and have become increasingly descriptive and encompassing since that time. A text reference that clearly describes these numerous models is Thomann and Mueller (1987). These differential and difference equation models describe the response of the receiving waters to inputs of organic and other wastes.

While the response of the receiving water to the input of a single stream of organic wastes had been largely modeled by the late 1950s, the response of a river or lake to a number of spatially separated waste streams had not been described analytically. If wastes from multiple treatment plants are entering a river, then an optimization problem arises in which one seeks the least cost set of treatment plant efficiencies (level of treatment or degree of removal)

that can ensure the dissolved oxygen concentration everywhere in the river remains above a desired level or standard. The desired level or standard reflects the uses of the water body whether for fishing, swimming, boating, etc. Of course, it is possible and desirable to develop the tradeoff between the system cost of wastewater treatment and the dissolved oxygen standard. The dissolved oxygen standard represents the value of the lowest level of dissolved oxygen that occurs anywhere along the length of the river. Linear and dynamic programming models have been developed for this optimization problem which is really a problem in linear optimal control — in the sense that the dissolved oxygen responses are governed by differential equations.

After manipulation, which can be extensive, of the governing equations that describe the system and the constraints on performance, these models can be converted to the optimization form

$$\textit{Water Pollution Abatement Model}:$$

$$\text{Minimize} \quad \sum_{i=1}^{n} c_i e_i$$

$$\text{s.t.} \quad \sum_{i \in I} a_{ij} e_i \geq S \quad \forall j \in J$$

$$0 \leq e_i \leq 1 \quad \forall i \in I$$

i, I = the sources and set of sources from which organic pollutants are discharged;

j, J = the points and set of points at which the dissolved oxygen standard must be met;

c_i = cost per unit of removal efficiency at source i;

e_i = the removal efficiency at source i;

a_{ij} = the amount of dissolved oxygen that is protected or is allowed to be present in the stream at point j per unit of removal efficiency at point i; and

S = the dissolved oxygen standard that must be met at all monitoring points in the river.

In an estuarine situation where tidal movements cause pollutants to mix upstream and downstream of their point of discharge, all a_{ij} coefficients are non-zero and positive. In contrast, in a non-tidal river, only those a_{ij} are nonzero and positive for which the point j is downriver from source i. That is, in a non-tidal river, pollution from a source i has negligible effect on a point of measurement upstream from that source.

It is useful to develop the multi-objective tradeoff curve between total treatment cost and the dissolved oxygen standard because costs may increase rapidly in some portion of the curve, suggesting that further gains in quality can be obtained only at considerable expense. The river basin optimization model that chooses treatment efficiencies for each of many waste sources is a fifth stage in the application of systems analysis to urban water management (see ReVelle and Ellis 1994; McGarity 1997).

While realistic and relatively complex, the treatment plant/river basin optimization models do not completely describe the options for designing a pollution abatement program for the waste sources on a river. Their lack centers around the assumption that each discharge of treated wastewater enters the receiving water body at a known and prespecified point somewhere along the river, usually at its point of origination. Thus, another fundamental problem, and a sixth stage of application, is the siting of wastewater treatment plants along a river. That is, the previous model selected removal efficiencies but assumed that the flows from each of the wastewater treatment plants entered the river at the same geographical position as the community or industry that generated the flow. In contrast, the problem of siting wastewater treatment plants assumes a single prespecified and high removal efficiency for all the plants on the river, but seeks the positions for discharges which minimize the total treatment cost. The single treatment level is presumed to be sufficiently high that dissolved oxygen standards are not violated along the river. At one extreme, discharges may still occur at each community or industry along the river. At the other extreme, discharges may be consolidated into a single regional wastewater treatment plant. Most likely, however, is the partial consolidation of wastewater flows with some at-source discharges and some flows merging at regional plants for treatment and discharge.

The motivation of this problem setting is that economies of scale in treatment may be captured when wastewater flows are combined and treated together. Working against this cost advantage in consolidating flows are the additional costs of piping and pumping that are incurred when wastewater flows are merged at central points. Thus, the objective of the regional wastewater treatment plant problem is to minimize the sum of treatment costs and piping/pumping costs. The more dispersed that communities are along a river, the less likely will be consolidation in regional plants. Much work has been done on this

problem since the early 1970s, most of it focusing on a fixed charge approximation of the concave costs of treatment. Zhu and ReVelle (1988) provided an efficient and exact solution to siting regional treatment plants along an essentially linear river via integer programming and refer to most previous published research on the problem. Whitlatch (1997) provides a review of the literature of regional treatment plant siting.

A variation of and combination of the two previous problems has not been well studied; this is the problem which seeks the least cost set of treatment efficiencies and the sites for regional wastewater treatment plants given that a dissolved oxygen standard is honored along the length of the river. Multiplicative nonlinearities and concave or fixed charge cost functions make the problem especially challenging.

A final and seventh stage of application of systems analysis to urban and regional water management is the problem of cost or burden sharing. The notion here is that a regional authority has been created whose goal is to stimulate cooperation in the solution of environmental problems. Cooperation takes the form of joint activities, e.g. regional wastewater plants vs. separate plants for each community — if the regional solution saves money. Since the authority is assumed to be unable to coerce the communities to cooperate, it must find the means to induce cooperation; the goal is to find an effective and attractive way to distribute the savings from joint undertakings. Such a distribution should be chosen in a way that makes every participating community better off than it would be if it were to treat its waste flow alone or to join some other non-optimal coalition. An article that refers to prior work in allocation of costs for regional environmental facilities is Zhu and ReVelle (1990), but a more expansive treatment of cost allocation across the many areas of water resources and water quality is given in Heaney (1997).

Solid Wastes Management

Management of the operation and of the design of urban and regional solid wastes systems constitutes an important environmental area for the application of optimization. Although the level of research activity in urban solid waste systems has decreased since the middle 1970s, challenging problems remain and would surely be addressed if research funding were to flow to this sector as it did twenty years ago. Regional solid waste management, especially with regard to hazardous wastes routing and siting, has been thriving.

The field of urban waste management may be roughly divided into two sectors: a collection/routing sector and a siting sector. In the first category, collection/routing, are two related classes of problem: routing within a district and the creation of districts. Routing within a prespecified district means either visiting at least once every link within the district with the least total length route (minimal retracing of links) or visiting each link twice (corresponding to collection on both sides of the street) with the least total length route (no triple tracing required). The principle of a routing that includes every link at least once is minimal retracing, a property that is achieved by minimal length matching of odd nodes (nodes with an odd number of incident links). In the routing that covers each arc twice (two-sided collection) there are no odd nodes so that matching is not required and the route can be completed with a total length equal exactly to two times the total length of links in the district. Alternatively, the best routing might be the one with the least total cost, where time is a major factor in the determination of cost. As a consequence, a route that included many left turns against a high volume of oncoming traffic might be inferior to a longer route with mostly right turns. Route design with consideration of cost, time and left turns remains a challenging problem.

The creation of collection districts from a large network of streets, however, requires a prior step in which links are assigned to each district in such a way that the total collection distance/time (and possibly volume or weight loading) in each district is within a preset bound. Once each district is created, the routing step would be undertaken next. However, the district creation step and the routing within a district step influence one another. That is, the length of the minimal length routing within the district is only finally determined by routing so that in theory the assignment of links to a district cannot be completed without knowing the length of the minimal route within the district. Heuristics have been used to resolve this problem (Liebman 1975).

In the second sector are siting problems. At least four types of siting problems can be identified: the central siting of collection vehicles, the siting of a central incinerator, the siting of sanitary landfills (of which there may be a number), and the siting of transfer stations (the stations where smaller trucks

offload to larger vehicles for more distant hauling). All of these siting problems have been studied, but not all are well solved.

A review of solid waste operations management is given in Liebman (1975), and a review of the siting of processing facilities and landfill sites is given in Gottinger (1988). Liebman (1997) revisited routing and siting in solid wastes management.

Regional solid waste management has received inspiration from the mounting problems of disposing of hazardous wastes. Issues of routing, scheduling and siting abound — with conflicting objectives. On the one hand, routes and sites should be chosen to minimize cost. On the other hand, routes and sites should be chosen to decrease risk and population exposure. A special issue of *Transportation Science*, edited by Turnquist and Zografos (1991), took up the issues in hazardous materials transportation in five articles. A review of the logistics of hazardous waste management is given in Turnquist and Nozick (1997).

Control of Air Pollution

The application of systems analysis to the control of air pollution dates from the late 1960s. Modeling efforts drawn from air pollution meteorology were then applied for the first time to develop predictive equations that could be applied in constraint form for air pollutant concentrations downwind from emission sources. From these predictive equations, transfer coefficients could be derived. Each such transfer coefficient provides the unit increment of pollutant concentration downwind (in say milligrams/cubic meter) at a particular point of measurement for each unit of emission (say tons per day) at each pollutant source. Thus, each ton/day of sulfur dioxide emitted from a specified source has a quantifiable impact on the atmospheric sulfur dioxide concentration at each of a number of downwind sites. With such transfer coefficients, as well as costs of abatement at the sources, and, with an atmospheric concentration standard to be met, it is possible to structure an optimization model. This air pollution management model chooses the least cost set of removal efficiencies, one removal level for every source, that achieves atmospheric concentrations of the pollutant at all specified points of concern at or below the standard.

A basic model that has been proposed for the management of acid rain is described next. The model follows the same form as the water pollution abatement optimization model described earlier. It should be noted that this model presumes no chemical reactions of pollutants, but only gradual dissipation of the pollutant.

The optimization problem is given as

$$\begin{aligned} &\textit{Acid Rain Management Model}: \\ &\text{Minimize} \quad z = \sum_{i \in I} c_i R_i \\ &\text{s.t.} \quad \sum_{i \in I} t_{ij} E_i R_i \geq \sum_{i \in I} t_{ij} E_i - S_j \quad (j \in J) \\ &0 \leq R_i \leq 1 \quad \forall\, i \in I. \end{aligned}$$

I, I = index and set of sources;

J,J = index and set of points at which concentrations are monitored;

R_i = fractional removal efficiency at source i;

E_i = tons per unit time of emissions at source i without any removal;

t_{ij} = transfer coefficients (mg/m^3/ton/day); the increment to the atmospheric concentration at j per unit of emissions at i;

S_j = atmospheric standard at receptor point j for the pollutant; and

c_i = cost per unit of pollutant removal.

This basic acid rain management problem can be manipulated in many ways. The transfer coefficients can be a single set of known numbers. They can also be random variables, and the constraints can then be either expected value constraints or chance constraints. Many possible sets of transfer coefficients can also be considered, leading to models that minimize maximum regret subject to investment in pollutant removal. Developments in air pollution management models, in general, and acid rain, in particular, were surveyed in reviews of water and air quality management by ReVelle and Ellis (1994) and Ellis (1997).

Concluding Remarks

The environmental models discussed here have much in common. Air and water pollution control models both have the equivalent of transfer coefficients which

translate upstream and upwind discharges and emissions into downstream and downwind concentrations. Water pollution control facilities as well as landfills and incinerators require siting and regionalization to minimize costs. Where power plant air emissions are part of the control equations, siting of power plants also becomes an issue in air quality management. Finally, burden sharing and cost allocation issues are common to all these areas of environmental management. An annotated bibliographic treatment of applications of mathematical programming to the many areas of environmental quality management is provided by Greenberg (1995). The problem set in environmental systems analysis is rich, diverse, challenging and important.

See

- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Location Analysis
- ▶ Natural Resources
- ▶ Vehicle Routing
- ▶ Water Resources

References

Ellis, J. (1997). Quality management, chapter 4. In C. ReVelle & A. McGarity (Eds.), *Design and operation of civil and environmental engineering systems*. New York: John Wiley.

Gottinger, H. (1988). A computational model for solid waste management with application. *European Journal of Operational Research, 35*, 350–364.

Greenberg, H. (1995). Mathematical programming models for environmental quality control. *Operations Research, 43*, 578–622.

Heaney, J. (1997). Cost allocation in water resources, chapter 13. In C. ReVelle & A. McGarity (Eds.), *Design and operation of civil and environmental engineering systems*. New York: John Wiley.

Liebman, J. (1975). Models of solid waste management, chapter 5. In S. Gass & R. Sisson (Eds.), *A guide to models in government planning and operations*. Potomac, MA: Sauger Books.

Liebman, J. (1997). Solid waste management, chapter 5. In C. ReVelle & A. McGarity (Eds.), *Design and operation of civil and environmental engineering systems*. New York: John Wiley.

McGarity, A. (1997). Water quality management, chapter 2. In C. ReVelle & A. McGarity (Eds.), *Design and operation of civil and environmental engineering systems*. New York: John Wiley.

ReVelle, C. (1999). Research challenges in environmental management. To appear in *European Journal of Operational Research*.

ReVelle, C., & Ellis, J. (1994). Models for air and water quality management, in operations research and public systems. In S. Pollock., A. Barnett., & M. Rothkopf (Eds.), *Handbooks of operations research* (Vol. 7). Elsevier.

Tang, C., Brill, E., & Pfeffer, J. (1987). Optimization techniques for secondary wastewater treatment systems. *Journal of Environmental Engineering Division, ASCE, 113*, 935–951.

Thomann, R., & Mueller, J. (1987). *Principles of surface water quality modeling and control*. New York: Harper and Row.

Turnquist, M., & Nozick, L. (1997). Hazardous waste management, chapter 6. In C. ReVelle & A. McGarity (Eds.), *Design and operation of civil and environmental engineering systems*. New York: John Wiley.

Turnquist, M., & Zografos, C. (Eds). (1991). Transportation science, Special Issue: *Transportation of Hazardous Materials*, *25*(1).

Wainright, J., & Mulligan, M. (Eds.). (2004). *Environmental modelling*. Chichester, England: John Wiley & Sons.

Walski, T., et al. (1987). Battle of the network models: Epilogue. *Journal of Water Resources Planning and Management, Division ASCE, 113*(2), 191.

Walters, G. (1985). The design of the optimal layout for a sewer network. *Engineering Optimization, 9*, 37–50.

Whitlatch, E. (1997). Siting regional environmental facilities, chapter 14. In C. ReVelle & A. McGarity (Eds.), *Design and operation of civil and environmental engineering systems*. New York: John Wiley.

Zhu, Z.-P., & ReVelle, C. (1988). A siting model for regional wastewater treatment systems. *Water Resources Research, 24*(1), 137–144.

Zhu, Z.-P., & ReVelle, C. (1990). A cost allocation method for facilities siting with fixed charge cost functions. *Civil Engineering Systems, 7*(1), 29–35.

EOQ

Economic order quantity.

See

- ▶ Economic Order Quantity
- ▶ Economic Order Quantity Model Extensions
- ▶ Inventory Modeling

Ergodic Theorems

Results giving the conditions for the time averages of a stochastic process to converge to its limiting or steady-state probability distribution.

See

- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Queueing Theory

Erlang

The unit of traffic load used in congestion analysis of telecommunication networks. The traffic load is the expected number of arrivals during an average service time. This quantity is dimensionless, but is refered to as the number of *erlangs* offered to the system. Named after the Danish mathematician/engineer A. K. Erlang, who founded modern queueing theory with his work on telephony in the early 1900s.

See

- ▶ Offered Load
- ▶ Queueing Theory

Erlang B Formula

The probability that all servers are busy in the multiserver queueing system M/M/c with Poisson input, exponential service and no waiting space, and thus that an arriving customer will be unable to enter the system (i.e., is blocked).

See

- ▶ Queueing Theory

Erlang C Formula

The probability that all servers are busy in the multiserver queueing system M/M/c with Poisson input, exponential service and infinite capacity.

See

- ▶ Queueing Theory

Erlang Delay Model

The multiserver queueing system M/M/c with Poisson input and identical exponential service for each server.

See

- ▶ Queueing Theory

Erlang Distribution

A continuous random variable is said to have an Erlang distribution if its probability density may be written in the form $f(t) = a(at)^{k-1}\ e^{-at}/(k-1)!$ where k is a positive integer and a is a positive real number. The constant k is called the shape parameter, while a (or various equivalents) is called the scale parameter. The Erlang distribution is a special case of the gamma distribution with integral shape parameter.

See

- ▶ Gamma Distribution

Erlang Loss Model

The multiple-server queueing system M/M/c with Poisson arrivals, exponential service times, c servers, but no additional space for holding customers.

See

- ▶ Erlang B Formula
- ▶ Queueing Theory

Error Analysis

- ▶ Numerical Analysis

Eta File

A sequential file storing the sequence of elementary elimination matrices used to obtain the *LU* decomposition of the basis matrix in the simplex method. Each elementary elimination matrix is represented by its eta vector.

See

- ▶ Revised Simplex Method

Eta Matrix

- ▶ Elementary Elimination Matrix

Eta Vector

The special column of a pivot (elementary elimination) matrix that is different from the corresponding column vector of the identity matrix. A pivot matrix is uniquely specified by its eta vector and its location in the matrix.

See

- ▶ Revised Simplex Method

Ethics in the Practice of Operations Research

Joseph H. Engel
Bethesda, MD, USA

Introduction

Ethics in the practice of operations research is the set of moral standards to which a practitioner of OR/MS should adhere in doing his or her work, so that the analyst can do relevant work responsibly and objectively, and be perceived as doing so.

The OR/MS worker must apply the basic principles of scientific methodology in such a way as to be transparent in the way the work is reported. A technically qualified but disinterested party should be able to verify that the work has been carried out in a valid manner, based on data that have been gathered and analyzed correctly.

Operations research, as distinguished from the physical sciences in general, deals with interactions between people and the systems they operate. With this in mind, the discussion here concerns the OR/MS analyst's ethical requirements operationally, in terms of beginning, conducting, and reporting a study (as covered in Caywood et al. 1971, 1129–1130).

In Beginning a Study

The OR/MS analyst should discuss thoroughly with the client the nature of the problem to be solved, and should become familiar with the system, so that the analyst and the client can reach agreement on the client's objectives in operating the system to be studied, measures of effectiveness in achieving the system objectives, and the boundaries of the system. Both parties need to "agree on what *will* and *will not* be done" (Caywood et al. 1971).

The careful delineation of the objectives of a system in planning how to begin a study is important in all cases, particularly where there are multiple objectives. All component objectives and measures of performance must be carefully defined.

Of comparable ethical importance in the formulation of the problem is the determination of

the extent of the system to be studied. This means that analyst and client should agree on which portions of the system can be affected by the operation of the system and what phenomena affected by the operation of the system are of concern to the client. Then, properly relevant analysis of possible operation of the system can lead to recommendations which should lead to the desired improvement.

It also is important for the analyst to understand the general nature of all of the effects that the system can have on the total environment, regardless of whether or not the system operator is directly interested in some of these effects. Uncovering unexpected effects may make it possible for the system operator to acquire a better understanding of the overall relationship of the system to its surrounding. This, perhaps, may lead to more useful results than might have otherwise been possible.

In Conducting a Study

Having selected measures of performance and having defined the system boundaries, the analyst must plan for data collection to ensure its maximum accuracy and relevance to the problem at hand, without interfering unduly with what the operating personnel are doing.

As in other people-oriented sciences, the OR/MS scientist often cannot conduct controlled experiments (because they inflict an undue burden of cost or damage on the operating personnel). The analyst may have to be content with observing a series of operational trials under what are hoped to be relevant field conditions. But, to the maximum extent possible, the number and nature of specific data collecting trials should be stipulated by the analyst, with the concurrence of the client, so as to ensure that a statistically valid amount of data covering all relevant facets of the operation will be collected.

This depends to a great degree on the nature of the mathematical model being used to describe the system being studied and how its performance is affected by factors under the control of the system operator, as well as by uncontrollable environmental factors. The time, personnel and equipment available and costs of conducting trials must also be taken into account in planning the quantity and extent of data collection.

The analyst should assure that qualified operators are trained in using appropriate data recording equipment and that they do record the data from the proper number of trials under the desired range of conditions that are to be recorded, or that they tell the analyst how many trials were really conducted, and under what conditions. Wherever possible, the analyst should observe the data collecting trials directly to determine whether they are conducted under true field conditions and to become aware of possible sources of error or inaccuracy in the data collection.

The analyst should not operate equipment being evaluated during data collection trials, because the analyst is not part of the operating system, and such participation can bias the results of the trials unexpectedly. This need not preclude the analyst from operating or observing the system on other convenient occasions to help build a good understanding of how the system is supposed to operate. It is generally preferable for the client's operating personnel rather than the analyst to collect the data. This is most desirable if the collection of such data is or ought to be part of the normal operating process (because such procedures are often valuable to the operators for training and self evaluation purposes).

Once the data are collected, the analyst must study it together with the mathematical model being used to describe system performance. The analyst must not arbitrarily omit data or add new or nonexistent data to develop results that are more to the liking of the analyst (or the client), and should make proper statistical, mathematical, and logical use of the data to derive valid conclusions. The analyst should try to reach an understanding of the nature of the conclusions, and why they agree (or fail to agree) with any prior opinions that may exist concerning the system being studied. The analyst should conduct sensitivity analyses of the effects of variations in key parameters or assumptions and should deal with possible limitations on the accuracy of observed data values and the effect of these limitations on the conclusions.

In studying a single objective system, the analyst uses a mathematical model appropriately to determine what combination of control variables will yield maximum performance effectiveness. Multi-objective systems are more difficult.

In the comparatively simple case of a single type of cost versus a single measure of performance, the analyst can treat the problem by first discovering how to maximize performance for a given cost, or, alternatively, by discovering the least expensive way to achieve a specified level of performance.

The decision maker (usually the client rather than the analyst) must decide on the maximum amount of money he or she wishes to spend, or the minimum level of performance desired. Then the analyst can solve the problem by recommending how to get the best performance at the desired expenditure level, or, alternatively, by recommending the smallest expenditure to achieve the desired level of performance.

The general solution in multi-objective systems is often difficult because there does not exist a mathematically rigorous method to find how to optimize the operation of any such system. In the special case when it can be shown that each of the objectives can be achieved more effectively when the system is operated in one certain way rather than in any other way, that way is a unique and dominating optimum solution to the problem. Similarly, multiple dominating solutions (each of them identically effective to all other dominating solutions with respect to each of the corresponding values of the component measures, and at least as effective in all of their components and more effective than some of the corresponding components of the non-dominated solutions) can be found. But dominating solutions do not always exist.

In such cases, it is necessary to consider possibly conflicting objectives in order to develop balanced procedures that deal with all important factors affecting performance. Such a problem might arise, for example, in optimizing the design of a military aircraft in terms of its range, cruising altitude, speed, payload weight, delivery accuracy, defensive ability, procurement and operating costs.

In general, a multi-objective index is used together with a sensibly designed mathematical model to be used to find out how to select values of control variables to maximize the value of the index. Such an index is usually structured to increase in value in a balanced way whenever any component measure indicates an improvement within an acceptable range (for example, the index might be a positively weighted sum of positive powers of each positive component measure). This is often hard to accomplish. If, for example, a system must be designed to optimize a time stream of expected short-run and long-term future costs and benefits, it is difficult to decide how to discount and balance long-term future costs and benefits against the short term. These long-range planning problems are not always dealt with properly, as witness frequent emphasis on ending each fiscal year in the black without paying enough attention to long-term profitability.

In Reporting a Study

Having performed the analysis and drawn conclusions, the analyst must report the findings and their possible limitations to the client in as complete and understandable a manner as possible.

It is well worth reviewing the mechanics, as well as the ethics of reporting a study. All aspects of the analysis, the data collection procedures, the fundamental assumptions and mathematical model used, including the values of any multi-objective index and each of its components, conclusions, recommendations and their limitations, should be reported and explained to the client. When an analysis is conducted using a multi-objective index, the analyst should explain to the client how the values of the coefficients and exponents used in the index have been chosen, including a thorough discussion of the implications inherent in their selection. At all costs, the analyst must avoid conducting and then reporting the analysis in such a way as to warp results intentionally so as to validate the analyst's own or anyone else's prior conclusions. Further, the analyst reports only to the client and nobody else without prior permission from the client. Leaks are unethical (Caywood et al. 1971).

The ethical problems connected with the reporting process revolve around the need to analyze and report relevantly, honestly, completely, clearly, and exclusively, so the client will understand what has been done as well as what has not been done. Failure to do so is an ethical failure, because the analyst will have failed to deliver what was contracted.

Concluding Remarks

Beyond the ethical requirement's of beginning, conducting, and reporting an OR study, as discussed above, there are a number of other ethical issues that analysts encounter in their research and applied activities. Many of these issues are similar to ones faced by all professionals, for example, data availability and computational reproducibility, peer review procedures, the handling of conflicts of interest. The book edited by Wallace (1994) covers the full range of such ethical issues. Further ethical concerns are discussed in the special issue on ethics and operations research of the journal *OMEGA* (37, 6, 2009); in particular, see the papers Gass (2009), Le Menestrel and Van Wassenhove (2009), and Walker (2009). Caywood et al. (1971) offers a valuable discussion of the concept and ethical issues of the analyst as an advocate. Brams (2002) discusses a multicriteria decision aid, the PROMETHEE–GAIA procedure, and shows how it could provide well-balanced solutions between rationality, subjectivity, and ethics.

See

- ▶ Implementation
- ▶ Multi-attribute Utility Theory
- ▶ Multiobjective Programming
- ▶ Practice of Operations Research and Management Science
- ▶ Verification, Validation, and Testing of Models

References

Brams, J.-P. (2002). Ethics and decisions. *European Journal of Operational Research, 136*, 340–352.

Caywood, T. E., et al. (1971). Guidelines for the practice of operations research. *Operations Research, 19*, 1123–1158.

Gass, S. I. (2009). Ethical guidelines and codes in operations research. *Omega, 37*, 1044–1050.

Kunsch, P. L., Kavathatzopoulos, I., & Rauschmayer, F. (2009). Modelling complex ethical decision problems with operations research. *Omega, 37*, 1100–1108.

Le Menestrel, M., & Van Wassenhove, L. (2009). Ethics in operations research and management sciences: a never-ending effort to combine rigor and passion. *Omega, 37*, 1039–1043.

Walker, W. (2009). Does the best practice of rational-style model-based policy analysis already include ethical considerations? *Omega, 37*, 1051–1062.

Wallace, W. A. (Ed.). (1994). *Ethics in modeling*. New York: Pergamon.

Euler Tour

In an undirected connected graph, an Euler tour is a cycle that starts at some node, visits each arc exactly once, and returns to the starting node.

See

- ▶ Chinese Postman Problem
- ▶ Combinatorics
- ▶ Graph Theory
- ▶ Integer and Combinatorial Optimization

EURO

Association of European Operational Research Societies.

See

- ▶ International Federation of Operational Research Societies (IFORS)

Evaluation

- ▶ Model Evaluation

Event-Driven Simulation

A computer simulation paradigm in which each simulated event is contained in a (logical) module of code (subroutine). Each module is executed when and only when other code determines that event should occur. Generally, event-driven simulations have stochastic (random) decisions that determine if and when (in model time) an event will occur.

See

- ▶ Simulation of Stochastic Discrete-Event Systems

Evolutionary Algorithms

Zbigniew Michalewicz[1] and Marc Schoenauer[2]
[1]The University of Adelaide, Adelaide, South Australia, Australia
[2]INRIA Saclay – Île-de-France, Orsay cedex, France

Introduction

The Evolutionary Computation (EC) techniques are stochastic algorithms whose search methods model some natural phenomena: genetic inheritance and Darwinian strife for survival. The idea behind Evolutionary Algorithms (EAs) is to do what nature does. Consider rabbits as an example: at any given time there is a population of rabbits. Some of them are faster and smarter than other rabbits. These faster, smarter rabbits are less likely to be eaten by foxes, and therefore more of them survive to do what rabbits do best: make more rabbits. Of course, some of the slower, dumber rabbits will survive just because they are lucky. This surviving population of rabbits starts breeding. The breeding results in a good mixture of rabbit genetic material: some slow rabbits breed with fast rabbits, some fast with fast, some smart rabbits with dumb rabbits, and so on. And on the top of that, nature throws in a "wild hare" every once in a while by mutating some of the rabbit genetic material. The resulting baby rabbits will (on average) be faster and smarter than these in the original population because more faster, smarter parents survived the foxes. (It is a good thing that the foxes are undergoing similar process — otherwise the rabbits might become too fast and smart for the foxes to catch any of them). So the metaphor underlying evolutionary algorithms is that of natural evolution. In evolution, the problem each species faces is one of searching for beneficial adaptations to a complicated and changing environment. The "knowledge" that each species has gained is embodied in the makeup of the chromosomes of its members. From the point of view of optimization, EC is a powerful stochastic zeroth order method (i.e., requiring only values of the function to optimize) that can find the global optimum of very rough functions. This allows EC to tackle optimization problems for which standard optimization methods (e.g., gradient-based algorithms requiring the existence and computation of derivatives) are not applicable. Moreover, most traditional methods are local in scope, thus they identify only the local optimum closest to their starting point.

An Algorithm

A general framework is introduced, which accounts for most existing Evolutionary Algorithms.

Let the search space be a metric space E, and let F be a function $E \to \mathbb{R}$ called the objective function. The task of evolutionary optimization is to find the maximum of F on E (the case of minimization is easily handled by considering $-F$).

A population of size $P \in \mathbb{N}$ is a set of P individuals (points of E) not necessarily distinct. This population is generally initialized randomly (at time $t = 0$) and uniformly on E. Then the fitness of each individual is computed (on the basis of the values of the objective function); a fitness value is represented as a positive real number—the higher the number, the better the individual. The population then undergoes a succession of generations; the process is illustrated in Fig. 1.

Several aspects of the evolutionary procedure require additional comments:

- **Statistics and stopping criterion**: The simplest stopping criterion is based on the generation counter t (or on the number of function evaluations). However, it is possible to use more complex stopping criteria which depends either on the evolution of the best fitness in the population along generations (i.e., measurements of the gradient of the gains over some number of generations), or on some measure of the diversity of the population.
- **Parental selection**: Choice of some individuals that will generate offspring. Numerous selection processes can be used, either deterministic or stochastic. All are based on the fitness of the individuals. Depending on the selection scheme used, some individuals can be selected more than once. At that point, selected individuals give birth to copies of themselves (clones).
- **Application of variation operators**: To each one of these copies some operator(s) are applied, giving birth to one or more offspring. The choice among possible operators is stochastic, according to

```
procedure evolutionary algorithm
begin
    t ← 0
    initialize population
    evaluate population
    while (not termination-condition) do
    begin
        t ← t + 1
        select individuals for reproduction
        apply variation operators
        evaluate newborn offspring
        select individuals for survival
    end
end
```

Evolutionary Algorithms, Fig. 1 The structure of an evolutionary algorithm

user-supplied probabilities. These operators are always stochastic operators; it is common to distinguish between crossover (or recombination) and mutation operators:

- Crossover operators are operators from E^k into E, i.e., some parents exchange genetic material to build up one offspring. In most cases, crossover involves two parents ($k = 2$), though more parents can be used.
- Mutation operators are stochastic operators from E into E.

– **Evaluation**: Computation of the fitnesses of all newborn offspring. As mentioned earlier, the fitness measure of an individual is directly related to its objective function value.
– **Survival selection**: Choice of which individuals will be part of next generation. The choice can be made either from the offspring only (in which case all parents die) or from both the offspring and the parents. In either case, this survival selection procedure can be deterministic or stochastic.

Sometimes the variation operators are defined on the same space as the objective function (called phenotype space or behavioral space); in other cases, an intermediate space is introduced (called genotype space or representation space). The mapping from the phenotype space in the genotype space is termed coding. The inverse mapping from the genotype space in the phenotype space is termed decoding. Genotypes undergo variation operators, and their fitness is evaluated on the corresponding phenotype. The properties of the coding mappings can greatly modify the global behavior of the evolutionary algorithm. For more (implementational) details related to the structure outlined in Fig. 1 and further discussion on the above aspects of the evolutionary procedure, see (Michalewicz and Fogel 2004).

An Example

The following example based on an example given in chapter 2 of (Michalewicz 1996) presents the action-steps of a standard genetic algorithm — the best-known paradigm within evolutionary algorithms — for a numerical optimization problem. These action-steps illustrate general structure of evolutionary algorithms given in Fig. 1. Here, without loss of generality, consider a maximization problem where the objective function f takes positive values on its domain.

Consider the maximization of a function of k variables, $f(x_1, \ldots, x_k) : \mathbb{R}^k \rightarrow \mathbb{R}$. Suppose further that each variable x_i can take values from a domain $D_i = [a_i, b_i] \subseteq \mathbb{R}$ and $f(x_1, \ldots, x_k) > 0$ for all $x_i \in D_i$. Assume the optimization specifies some required precision: suppose six decimal places for the variables' values is desirable.

It is clear that to achieve such precision each domain D_i should be cut into $(b_i - a_i) \cdot 10^6$ equal size ranges. Denote by m_i the smallest integer such that $(b_i - a_i) \cdot 10^6 \leq 2^{m_i} - 1$. Then, a representation having each variable x_i coded as a binary string of length m_i clearly satisfies the precision requirement. Additionally, the following formula interprets each such string:

$$x_i = a_i + decimal(1001...001_2) \cdot \frac{b_i - a_i}{2^{m_i} - 1},$$

where $decimal(string_2)$ represents the decimal value of that binary string.

Now, each individual (in genetic algorithms terminology, individuals are often called chromosomes) is represented by a binary string of length $m = \sum_{i=1}^{k} m_i$; the first m_1 bits map into a value from the range $[a_1, b_1]$, the next group of m_2 bits map into a value from the range $[a_2, b_2]$, and so on; the last group of m_k bits map into a value from the range $[a_k, b_k]$. A chromosome represents a potential solution to the problem.

To initialize the population, simply set some *pop_size* number of chromosomes randomly in

a bitwise fashion. However, if there is some knowledge about the distribution of potential optima, such information may be used in arranging the set of initial (potential) solutions.

At each generation (while loop, see Fig. 1), evaluate each chromosome (using the function f on the decoded sequences of variables), select new population with respect to the probability distribution based on fitness values, and alter the chromosomes in the new population by mutation and crossover operators. After some number of generations, when no further improvement is observed, the best chromosome represents an (possibly the global) optimal solution. Often the algorithm is stopped after a fixed number of iterations depending on speed and resource criteria. For the selection process (selection of a new population with respect to the probability distribution based on fitness values), a *roulette wheel* with slots sized according to fitness is used here; such a roulette wheel is constructed as follows:

- Calculate the fitness value $eval(v_i)$ for each chromosome v_i $(i = 1, \ldots, pop\_size)$.
- Find the total fitness of the population

$$F = \sum_{i=1}^{pop\_size} eval(v_i).$$

- Calculate the probability of a selection p_i for each chromosome v_i $(i = 1, \ldots, pop\_size)$:

$$p_i = eval(v_i)/F.$$

- Calculate a cumulative probability q_i for each chromosome v_i $(i = 1, \ldots, pop\_size)$:

$$q_i = \sum_{j=1}^{i} p_j.$$

The selection process is based on spinning the roulette wheel pop_size times; each time select a single chromosome for a new population in the following way:

- Generate a random (float) number r from the range [0..1].
- If $r < q_1$ then select the first chromosome (v_1); otherwise select the i-th chromosome v_i $(2 \le i \le pop\_size)$ such that $q_{i-1} < r \le q_i$.

Obviously, some chromosomes would be selected more than once. This is in accordance with the Schema Theorem (see Comparison section): the best chromosomes get more copies, the average stay even, and the worst die off.

Now, apply the recombination operator, crossover, to the individuals in the new population. As mentioned earlier, one of the parameters of a genetic system is probability of crossover p_c. This probability gives us the expected number $p_c \cdot pop\_size$ of chromosomes which undergo the crossover operation. Proceed as follows:

For each chromosome in the (new) population:

- Generate a random (float) number r from the range [0..1];
- If $r < p_c$, select given chromosome for crossover.

Now, mate selected chromosomes randomly: for each pair of coupled chromosomes generate a random integer number pos from the range $[1..m-1]$ (m is the total length — number of bits — in a chromosome). The number pos indicates the position of the crossing point. Two chromosomes

$$(b_1 b_2 \ldots b_{pos} b_{pos+1} \ldots b_m) \text{ and}$$
$$(c_1 c_2 \ldots c_{pos} c_{pos+1} \ldots c_m)$$

are replaced by a pair of their offspring:

$$(b_1 b_2 \ldots b_{pos} c_{pos+1} \ldots c_m) \text{ and}$$
$$(c_1 c_2 \ldots c_{pos} b_{pos+1} \ldots b_m).$$

The next operator, mutation, is performed on a bit-by-bit basis. Another parameter of the genetic system, probability of mutation p_m, gives us the expected number of mutated bits $p_m \cdot m \cdot pop\_size$. Every bit (in all chromosomes in the whole population) has an equal chance to undergo mutation, i.e., change from 0 to 1 or vice versa. So proceed as follows:

For each chromosome in the current (i.e., after crossover) population and for each bit within the chromosome:

- Generate a random (float) number r from the range [0..1];
- If $r < p_m$, mutate the bit.

Following selection, crossover, and mutation, the new population is ready for its next evaluation.

This evaluation is used to build the probability distribution (for the next selection process), i.e., for a construction of a roulette wheel with slots sized according to current fitness values. The rest of the evolution is just cyclic repetition of the above steps.

It is relatively easy to keep track of the best individual in the evolution process. It is customary (in genetic algorithm implementations) to store "the best ever" individual at a separate location; in that way, the algorithm would report the best value found during the whole process (as opposed to the best value in the final population) — this approach illustrates so-called elitist strategy.

Historical Paradigms

This section introduces the four historical paradigms that build what is today known as Evolutionary Computation.

Genetic Algorithms

In the canonical genetic algorithm (GA) (Holland 1975; Goldberg 1989), the genotype space is $\{0,1\}^n$. Note that the phenotype space can be any space, as long as it can be coded into bitstring genotypes. The parental selection is proportional selection (the best-known being the roulette wheel selection as discussed in the previous section): P random choices are made in the whole population, each individual having a probability proportional to its fitness of being selected. The crossover operators replace a segment of bits in the first parent string by the corresponding segment of bits from the second parent, and the mutation operator randomly flips the bits of the parent according to a fixed user-supplied probability. In the survival selection phase, all P offspring replace all parents (aka generational survival). The best fitness in the population can thus decrease: the original GA strategy is not elitist.

However, it rapidly became clear that the genotype space can be almost any space, as long as some crossover and mutation operators are provided (Radcliffe 1991; Michalewicz 1996). Moreover, proportional selection has been gradually replaced by comparison-based selections, from ranking (the selection is performed on the rank of the individuals rather than on their actual fitness), or tournament selection (one selects the best individual among a uniform choice of T individuals, T ranging from 2 to some small proportion of the population size) – see, e.g., (Chakraborty et al. 1996) for a discussion on these selection methods. Finally, most users use the elitist variant of survival selection, in which the best individual of generation t is included in generation $t+1$, whenever the best fitness value in the population decreases.

Evolution Strategies

Evolution Strategies (ESs) (Rechenberg 1972; Schwefel 1981) have been designed as parametric optimization algorithms, i.e., optimizing functions of floating-point variables. The original ES handles a population made of a single individual given as a real-valued vector. This individual undergoes a Gaussian mutation: addition of zero-mean Gaussian variable of standard deviation σ to each of the real variables. The fittest from the parent and the offspring becomes the parent of next generation. The critical feature is the choice of parameter σ: Originally, the so-called $1/5$ thumb rule (i.e., when more than $1/5$ mutation are successful (respectively unsuccessful), increase (respectively decrease) σ (Rechenberg 1972).

ES then evolved into population-based algorithms (Bäck and Schwefel 1993), termed $(\mu,\lambda)-$ES or $(\mu+\lambda)-$ES: μ parents generate λ offspring. (There is no parental selection, i.e., every parent produces λ/μ offspring on average). Moreover, the survival selection is deterministic, i.e., the best μ individuals become the parents of the next generation, chosen among the $\mu+\lambda$ parents plus offspring in the elitist $(\mu+\lambda)-$ES scheme, or among the λ offspring in the non-elitist $(\mu,\lambda)-$ES scheme (with $\lambda \geq \mu$).

The main operator remains mutation, generalized into multi-variate Gaussian mutation, i.e., defined by a full covariance matrix C and a scaling factor σ, also called the step-size mutation. In the 90s, the powerful paradigm of self-adaptive mutation was the rule: C and σ were added to the description of the individuals, and undergo mutation as well. The recent trend is to adapt C and σ based on the history of the evolution, leading to the state-of-the-art algorithm of Covariance Matrix Adaptation – Evolution Strategy, CMA-ES (Hansen and Ostermeier 2001; Auger and Hansen 2005). CMA-ES is almost parameter-free: it uses a $(\lambda/2,\lambda)-$ES survival selection, where the parameter λ is chosen as $4+\lfloor 3\log(n)\rfloor$ (Hansen and Ostermeier 2001), and increased in case of highly-multi-modal functions (Auger and Hansen 2005).

Evolutionary Programming

Evolutionary Programming (EP) is one of the oldest EAs, originally proposed to evolve finite state machines [27]. As in ESs, there is no parental selection, and every individual in the population generates one offspring. Moreover, the only evolution operator is mutation. Survival selection is what is today called a $(P + P)$-ES, i.e. the best P individuals among parents and offspring become the parents of the next generation.

As in the field of GAs, further works rapidly generalized the approach to handle any search space, still emphasizing the use of mutation as the only operator (Fogel 1995). Several variants were then introduced, from stochastic survival selection to self-adaptive mutations (similar to the ones from Evolution Strategies, though discovered completely independently (Saravanan et al. 1995)).

Genetic Programming

Genetic Programming (GP) as a method for evolving computer programs first appeared as an application of GAs to tree-like structures (Koza 1992). Original GP evolves tree structures representing LISP-like S-expressions. The strength of this representation is that a closed crossover operator can easily be defined: by swapping sub-trees between two valid S-expressions, one always gets a valid S-expression. Koza's original work used a steady state genetic algorithm evolutionary mechanism (Syswerda 1991): a parent is selected by tournament (of size 2 to 7 typically), and generates an offspring by crossover only. The offspring is then put back in the population using a reverse-tournament: T individuals are uniformly chosen, and the one with the worst fitness gets replaced by the newborn offspring.

Since then, most published work considered also mutation. Further, several variants of tree-based GP have been proposed (e.g., linear GP (Nordin 1997), Cartesian GP (Miller and Smith 2006), push-GP (Spector and Robinson 2002)). At present, Genetic Programming is viewed more generally as program evolution (Banzhaf et al. 1998).

Modern Trends: Hybrid Methods

Many researchers further modified evolutionary algorithms by adding some problem-specific knowledge to the algorithm. Several papers have discussed initialization techniques, different representations, decoding techniques (mapping from genetic representations to phenotypic representations), and the use of heuristics for variation operators. Davis [20] wrote (in the context of classical, binary GAs):

> It has seemed true to me for some time that we cannot handle most real-world problems with binary representations and an operator set consisting only of binary crossover and binary mutation. One reason for this is that nearly every real-world domain has associated domain knowledge that is of use when one is considering a transformation of a solution in the domain [. . .]. I believe that genetic algorithms are the appropriate algorithms to use in a great many real-world applications. I also believe that one should incorporate real-world knowledge in one's algorithm by adding it to one's decoder or by expanding one's operator set.

Such hybrid/nonstandard systems enjoy a significant popularity in the evolutionary computation community. Very often these systems, extended by the problem-specific knowledge, outperform other classical evolutionary methods as well as other standard techniques. For example, a system Genetic-2 N (Michalewicz 1996) constructed for the nonlinear transportation problem used a matrix representation for its chromosomes, a problem-specific mutation (main operator, used with probability 0.4) and arithmetical crossover (background operator, used with probability 0.05). It is hard to classify this system: it is not really a genetic algorithm, since it can run with a mutation operator only without any significant decrease of the quality of results. Moreover, all matrix entries are floating-point numbers. It is not an evolution strategy, since it did not use Gaussian mutation, nor did it encode any control parameters in its chromosomal structures. Clearly, it has nothing to do with genetic programming and very little (just matrix representation) with evolutionary programming approaches. It is just an evolutionary computation technique aimed at a particular class of problems.

Comparison

Many papers have been written on the similarities and differences between these approaches (Bäck and Schwefel 1993; Fogel 1995; Bäck 1995). Clearly, these similarities and differences can be discussed from different perspectives.

- **The representation issue**
  Original GAs, ESs and EP address only bitstrings, real numbers and finite state machines, respectively. However, recent tendencies indicate that this is not a major difference. Moreover, it is still far from trivial to select appropriate variation operators for the chosen representation and the objective function, aka the *fitness landscape* (Radcliffe 1991; Michalewicz 1996).
- **The usefulness of crossover**
  According to the Schema Theorem (Holland 1975; Goldberg 1989), GAs main strength comes from the crossover operator: better and better solutions are built by exchanging building blocks from partially good solutions previously built, in a bottom-up approach. The mutation operator is then considered as a background operator. On the other hand, the philosophy behind EP and ESs is that such building blocks might not exist, at least for most real-world problems. Also some experimental results contradict the building block hypothesis; for example, uniform crossover (Syswerda 1989) is very disruptive of short schemata whereas one and two-point crossover are more likely to conserve short schemata and combine their defining bits in offspring. This top-down view considers that selective pressure plus genotypic variability brought by mutation are sufficient. This discussion on significance of the crossover has been going on for a long time. However, even when crossover is experimentally demonstrated beneficial to evolution, it could be because it acts like a large mutation (Jones 1995). Yet another example of the duality between crossover and mutation comes from GP's history: the original GP algorithm (Koza 1992) used only crossover, with no mutation at all, with very large population size: the rationale is that all building blocks that are necessary to represent at least one sufficiently good solution are already present in the initial population, and only need to be recombined.
- **Mutation operators**
  Whereas the usefulness of crossover has been heavily discussed, that of mutation is acknowledged by all trends (with the historical exception of Koza's work in GP – Genetic Programming section). Indeed, mutation is the only operator that ca re-introduce diversity in the population, as crossover is de facto limited by the current population. From a theoretical point of view, only mutation can guarantee the ergodocity of the stochastic process (i.e. that all points of the search space can be reached whatever the current population). However, the way mutation operators are applied differ from one paradigm to another, and should be closely linked to the types of selection that are used. Traditional GAs use a low mutation rate, i.e. the average number of bits that are flipped remains small (though it can take large values, with very small probabilities). Because all offspring replace all parents without competition, high mutation rates would totally prevent any convergence to any optimal solution, as mutation gets more and more destructive as the population contains better and better solutions. Within ESs, on the other hand, all individuals undergo mutation, and only the strength of the mutation is varied (e.g., the standard deviation of the Gaussian mutation, in one dimension). Together with the fact that several offspring are generated from each parent, this nevertheless still allows the algorithm to converge. Furthermore, it can converge to any arbitrary precision provided the mutation strength is decreased accordingly: this was first demonstrated by self-adaptive ES (Schwefel 1981; Beyer 2001), and proved right for CMA-ES as well (Hansen and Ostermeier 2001). Indeed, adaptive and self-adaptive mutations can have a significant impact only when all individuals undergo mutation. Furthermore, it also requires some property of the mutation with respect to the fitness function called the strong causality principle emphasized in Rechenberg (1972): mutation should be parameterized in such a way that small mutations have small effects on the fitness. This property is not specific to floating-point optimization, and should be kept in mind when designing mutation operators for a particular representation. In particular, this property is not true when floating-point numbers are encoded into binary strings (as is the case in traditional-syle GAs).
- **The selection mechanisms**
  They range from the totally stochastic fitness proportional parental selection of GAs with generational survival, to the deterministic (μ, λ) survival selection of ES, through the stochastic, but elitist (i.e. preserving the best), tournament-based survival step of EP and the steady-state scheme used

in GP. Though some studies have been devoted to selection mechanisms (see e.g., (Chakraborty et al. 1996)), the choice of both selection steps for a given problem (fitness-representation-operators) is still an open question (and is very likely to be problem-dependent).

The current trend in the EC community is to mix up all these features on a very pragmatic basis, trying to best fit the application at hand: some ESs applications deal with discrete or mixed real-integer spaces (Bäck 1995), the early arguments (Antonisse 1989) against the dogma "binary is the best" have diffused in the community, and the Schema Theorem has been extended to any representation (Radcliffe 1991). Moreover, most ESs (including CMA-ES) use some crossover operator, mutation has been added to GP, etc. And the different selection operators are more and more being used now by the whole community.

On the other hand, such hybrid algorithms, by getting away from the simple original algorithms, also escape the few available theoretical results, and EAs can today only rely on successful applications (see Application Areas section) to demonstrate their usefulness as general-purpose optimization algorithms.

Theoretical Results

Theoretical studies of Evolutionary Algorithms can be roughly categorized into three different types.

- An Evolutionary Algorithm can be viewed as a Markov chain in the space of populations, as the population at time $t+1$ only depends on the population at time t. The full theory of Markov chains can then be applied. Some asymptotic results were obtained for general EAs (Eiben et al. 1991), and for more focused algorithms (Rudolph 1997), but for very specific and simple functions. Stronger results (convergence in finite time) were obtained using the powerful Friedlin-Wentzell theory (Cerf 1996), for very general functions, but a specific algorithm. However, those results have limited practical consequences as the real-world environment often requires short response times.
- The specific characteristics of Evolution Strategies allowed precise theoretical studies on the sphere function, i.e. the quadratic function. Early theoretical studies by the ES pioneers resulted in Rechenberg's 1/5th rule (Rechenberg 1972), and in Schwefel's first self-adaptive ES (Schwefel 1981). Those works were later pursued in Schwefel's group in Dortmund, with the precise study of the so-called progress rate, i.e. the improvement from one step of the algorithm to the next, by Beyer (2001). Finally, those results that had been obtained asymptotically for very high dimensions, have been rigorously justified in any dimension. In particular, all numerical values for optimal settings have been shown to actually be optimal, with the help of the theory of irreducible Harris recurrent Markov chains (Auger 2005; Auger and Hansen 2006).
- Last, but not least, classical Algorithm Complexity Theory has been used to study EAs on discrete spaces. The pioneer of this approach was Ingo Wegener, and most works in this area come from his group in Dortmund (and some spin-offs groups led by former Wegener PhD students). Here the algorithms under study are simple but actual algorithms (and they get less and less "simple" every year), while the fitness functions are simple functions or classes of functions. Results range from effective complexity for hitting the optimum of linear boolean functions with a (1 + 1)-ES (Droste et al. 1998) to general bounds for stochastic search algorithm in generic black-box scenarios (Droste et al. 2006).

However, one should keep in mind that all the above theoretical analyses address some simple models of Evolutionary Algorithms. As stated earlier, the modern trends of EC gave birth to algorithms working on poorly structured search spaces (see next section), or hybrid algorithms, for which no theory is applicable at the moment.

Application Areas

It is widely acknowledged that in Evolutionary Computation theory lags far behind practice. Indeed, lessons from successful applications are one of the main driving forces of EA research today. Several edited books are devoted to applications of EAs (e.g., see the recent (Yu et al. 2008)), and almost every event related to EAs has its own special session dedicated to applied works, and their proceedings provide a wide overview of actual applications (e.g., see several

special sessions of the annual IEEE Congress on Evolutionary Computation – CEC, and the Real-World Application track run every year during the ACM Genetic and Evolutionary Computation Conference – GECCO).

This section will quickly survey the preferred domains of application of EAs, and the different sub-domains will be distinguished according to the type of search space they involve.

Combinatorial Optimization

Hard combinatorial optimization problems (NP-hard, NP-complete) involve huge discrete search spaces, and have been studied extensively by the Operational Research community. Two different situations should be considered: academic benchmark problems and large real-world problems.

As far as benchmark problems are concerned, it is now commonly acknowledged that pure EAs alone cannot compete with OR methods (see all papers about combinatorial optimization in (Bäck et al. 1997) for instance). However, in the last decade, hybrid algorithms termed Genetic Local Search, or Memetic Algorithms where the EA searches the space of local optima with respect to some OR heuristic, have obtained the best-so-far results on a number of such benchmark problems (e.g., from (Merz and Freisleben 1999) to (Merz and Huhse 2009)).

The situation is slightly different for real-world problems: pure OR heuristics generally don't directly apply, and OR methods have to take into account problem specificities. This is true of course for EAs, and there are many success stories where EAs, carefully tuned to the problem at hand, have been very successful, as for instance in the broad area of scheduling (Paechter et al. 1998; Semet and Schoenauer 2006). A few commercial applications of EAs were also reported; some of them resulted in a significant return on investment (Michalewicz et al. 2005). However, these applications required also forecasting components, resulting in hybrid adaptive business intelligence systems (Michalewicz et al. 2006).

Parametric Optimization

The optimization of functions with floating-point variables has been thoroughly studied by practitioners, and many very powerful methods exists. Though the most well-known methods address linear convex problems, there are many other cases that can be handled successfully (Bonnans et al. 1997). However, the recently appeared CMA-ES (Covariance Matrix Adaptation Evolution Strategy) (Hansen and Ostermeier 2001; Auger and Hansen 2005) can be viewed as the leading Evolutionary Algorithms in the continuous domain, outperforming the best methods from both deterministic and stochastic domains for highly multi-modal, irregular, ill-conditioned and non-separable functions. An impressive list of applications of CMA-ES is maintained on its inventor's web page (Hansen 2009).

The situation is different when dealing with multi-objective problems: Multi-Objective Evolutionary Algorithms (MOEAs) are the only ones that can produce a set of best possible compromise (the Pareto set), and have recently received increased attention. MOEAs use the same variation operators than standard EAs, but the Darwinian components are modified to take into account the multi-valued fitness (Deb 2001; Coello et al. 2002). Note that MOEAs are discussed in the Parametric Optimization section because prominent application results have been obtained in that area (see e.g., (Obayashi 1997)), but apply on any search space, as only the Darwinian part of the algorithm is different from that of a single-objective EA.

Mixed Search Spaces

Mixed search spaces involve different types of variables, generally both continuous and discrete, and EAs are flexible enough to handle such search spaces easily. Once variation operators are known for continuous and discrete variables, constructing variation operators for mixed individuals is straightforward: crossover for instance can either exchange values of corresponding variables, or use the variable-level crossover operator. Many problem have been easily handled that way, like the optical filter optimization (Bäck and Schütz 1995; Martin et al. 1995), where one is looking for a number of layers, the unknown being the layer thickness (continuous variable) and the material the layer is made of (discrete variable). Furthermore, some platforms now exist that help the non-EC expert to implement generic EAs for a given problem involving different types of mixed search spaces, requiring only a structured description of potential solutions – and of course a routine to compute the fitness of those candidate solutions (Costa and Schoenauer 2009).

Artificial Creativity

A very promising area of application of EAs, where EAs can be much more than yet another optimization method, is that of design, where the ability of EAs to handle almost any search space allows programmers (and artists!) to unveil their wildest ideas. The idea of component-based representations can boost innovation in structural design (Gero 1998; Hamda and Schoenauer 2002), architecture (Rosenman 1999), as well as in many other areas including art (Bentley 1999). But the most original idea in that direction is that of emrbyogenies: the genotype is a program, and the phenotype is the result of applying that program to "grow an embryo"; the fitness of the genotype (the program) is obtained by testing that phenotype in a real situation. Such approach already lead to astonishing results in analog circuit design for instance (Koza et al. 1999) – though exploring a huge search space (a space of programs) implies a heavy computational cost. Note that those results were achieved using Genetic Programming, using both crossover and mutation, and a huge (distributed) population as well.

Concluding Remarks

Natural evolution can be considered as a powerful problem solver achieving Homo Sapiens from chaos in only a couple of billion years. Computer-based evolutionary processes can also be used as efficient problem solvers for optimization, constraint handling, machine learning and modeling tasks. Furthermore, many real-world phenomena from the study of life, economy, and society can be investigated by simulations based on evolving systems. Last but not least, evolutionary art and design form an emerging field of applications of the Darwinian ideas.

An interesting question asks for guidance on the types of problems for which evolutionary methods are more appropriate than, say, standard operations research methods. Real-world problems are usually difficult to solve for several reasons; these include (Michalewicz and Fogel 2004):

- The number of possible solutions is so large as to forbid an exhaustive search for the best answer.
- The evaluation function that describes the quality of any proposed solution is noisy or varies with time, thereby requiring not just a single solution but an entire series of solutions.
- The possible solutions are so heavily constrained that constructing even one feasible answer is difficult, let alone searching for an optimum solution.

Naturally, this list could be extended to include many other possible obstacles. For example, one could include noise associated with our observations and measurements, uncertainly about given information, and the difficulties posed by problems that have multiple and possibly conflicting objectives (which may require a set of solutions rather than a single solution). All these reasons are just various aspects of the complexity of the problem.

Note that every time a problem is solved, one is only finding the solution to a *model* of the problem. All models are a simplification of the real world — otherwise they would be as complex and unwieldy as the natural setting itself. Thus the process of problem solving consists of two separate general steps: (1) creating a model of the problem, and (2) using that model to generate a solution:

Problem $\Rightarrow$ Model $\Rightarrow$ Solution.

Note that the "solution" is only a solution in terms of the model. If the model has a high degree of fidelity, there is more confidence that the solution will be meaningful. In contrast, if the model has too many unfulfilled assumptions and rough approximations, the solution may be meaningless, or worse.

So in solving real-world problem there are at least two ways to proceed:

1. Try to simplify the model so that traditional methods might return better answers.
2. Keep the model with all its complexities, and use nontraditional approaches, to find a near-optimum solution.

In other words, the more complexity in the problem (e.g., size of the search space, evaluation function, noise, constraints), the more appropriate it is to use a nontraditional method, like Evolutionary Algorithms. One often has to choose between approximating a model or approximating the solution. And a large volume of experimental evidence shows that this latter approach can often be used to practical advantage.

One can also look at Evolutionary Algorithms from a broader perspective of population-based methods, where a set of potential solutions is being processed

in parallel. Apart from Evolutionary Algorithms, there are other population-based approaches that have been proposed over the last 20 years; these include Differential Evolution (Price et al. 2005), Artificial Immune Systems (Dasgupta and Niño 2009), Particle Swarm Optimization (Clerc 2006), Ant Colony Optimization (Dorigo and Stutzle 2004), and Cultural Algorithms (Cowan and Reynolds 2004).

See

- ▶ Heuristics
- ▶ Metaheuristics

References

Antonisse, J. (1989). A new interpretation of schema notation that overturns the binary encoding constraint. In J. D. Schaffer (Ed.), *Proceedings of the 3rd international conference on genetic algorithms* (pp. 86–91). Morgan Kaufmann.

Auger, A. (2005). Convergence results for (1, λ)-SA-ES using the theory of φ-irreducible Markov chains. *Theoretical Computer Science, 334*(1–3), 35–69.

Auger, A., & Hansen, N. (2005). A restart CMA evolution strategy with increasing population size. In *Proceedings of the CEC'05* (pp. 1769–1776). IEEE Press.

Auger, A., & Hansen, N. (2006). Reconsidering the progress rate theory for evolution strategies in finite dimensions. In M. Cattolico (Ed.), *Proceedings of the ACM GECCO'06* (pp. 445–452). ACM Press.

Bäck, T. (1995). *Evolutionary algorithms in theory and practice*. New-York: Oxford University Press.

Bäck, T., & Schütz, M. (1995). Evolution strategies for mixed-integer optimization of optical multilayer systems. In J. R. McDonnell., R. G. Reynolds, & D. B. Fogel (Eds.), *Proceedings of the 4th annual conference on evolutionary programming*. MIT Press.

Bäck, T., & Schwefel, H.-P. (1993). An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation, 1*(1), 1–23.

Bäck, T., Fogel, D., & Michalewicz, Z. (Eds.). (1997). *Handbook of evolutionary computation*. Oxford University Press.

Banzhaf, W., Nordin, P., Keller, R., & Francone, F. (1998). *Genetic programming — An introduction on the automatic evolution of computer programs and its applications*. Morgan Kaufmann.

Bentley, P. J. (Ed.). (1999). *Evolutionary design by computers*. Morgan Kaufman Publishers Inc.

Beyer, H. G. (2001). *The theory of evolution strategies*. Springer Verlag.

Bonnans, F., Gilbert, J., Lemarechal, C., & Sagastizbal, C. (1997). *Optimisation numrique, aspects thoriques et pratiques* (Vol. 23). *Mathematiques & Applications*. Springer Verlag.

Cerf, R. (1996). An asymptotic theory of genetic algorithms. In J. M. Alliot., E. Lutton., E. Ronald., M. Schoenauer, & D. Snyers (Eds.), *Artificial evolution: LNCS* (Vol. 1063, pp. 37–53). Springer Verlag.

Chakraborty, U., Deb, K., & Chakraborty, M. (1996). Analysis of selection algorithms: A Markov chain approach. *Evolutionary Computation, 2*, 133–168.

Clerc, M. (2006). *Particle swarm optimization*. Wiley.

Coello, C. A. C., Veldhuizen, D. A. V., & Lamont, G. B. (2002). *Evolutionary algorithms for solving multi-objective problems*. Kluwer Academic Publishers.

Costa, L. D., & Schoenauer, M. (2009). Bringing evolutionary computation to industrial applications with. In G. Raidl et al. (Ed.), *Proceedings of the GECCO'09*. ACM Press. To appear.

Cowan, G. S., & Reynolds, R. G. (2004). *Acquisition of software engineering knowledge: Sweep, an automatic programming system based on genetic programming and cultural algorithms*. World Scientific Publishing Company.

Dasgupta, D., & Niño, L. F. (2009). *Immunological computation*. CRC Press.

Davis, L. (1991). *Handbook of genetic algorithms*. New York: Van Nostram Reinhold.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. John Wiley.

Dorigo, M., & Stutzle, T. (2004). *Ant colony optimization*. The MIT Press.

Droste, S., Jansen, T., & Wegener, I. (1998). A rigorous complexity analysis of the (1 + 1) evolutionary algorithm for separable functions with boolean inputs. *Evolutionary Computation, 6*(2), 185–196.

Droste, S., Jansen, T., & Wegener, I. (2006). Upper and lower bounds for randomized search heuristics in black-box optimization. *Theory of Computing Systems, 4*, 525–544.

Eiben, A., Aarts, E., & Hee, K. V. (1991). Global convergence of genetic algorithms: A Markov chain analysis. In H. P. Schwefel, & R. Männer (Eds.), *Proceedings of the 1st parallel problem solving from nature* (pp. 4–12). Springer Verlag.

Fogel, D. B. (1995). *Evolutionary computation. Toward a new philosophy of machine intelligence*. Piscataway, NJ: IEEE Press.

Fogel, L. J., Owens, A. J., & Walsh, M. J. (1966). *Artificial Intelligence through simulated evolution*. New York: John Wiley.

Gero, J. (1998). Adaptive systems in designing: New analogies from genetics and developmental biology. In I. Parmee (Ed.), *Adaptive computing in design and manufacture* (pp. 3–12). Springer Verlag.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Addison Wesley.

Hamda, H., & Schoenauer, M. (2002). Topological optimum design with evolutionary algorithms. *Journal of Convex Analysis*, 503–517.

Hansen, N. (2009+). References to CMA-ES applications.

Hansen, N., & Ostermeier, A. (2001). Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, (2), 159–195.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan Press.

Jones, T. (1995). Crossover, macromutation and population-based search. In L. J. Eshelman (Ed.), *Proceedings of 6th ICGA* (pp. 73–80). Morgan Kaufmann.
Koza, J. R. (1992). *Genetic programming: On the programming of computers by means of natural evolution*. Massachusetts: MIT Press.
Koza, J. R., et al. (1999). *Genetic programming III: Automatic synthesis of analog circuits*. Massachussetts: MIT Press.
Martin, S., Rivory, J., & Schoenauer, M. (1995). Synthesis of optical multi-layer systems using genetic algorithms. *Applied Optics*, 2267.
Merz, P., & Freisleben, B. (1999). Fitness landscapes and memetic algorithm design. In D. Corne, M. Dorigo, & F. Glover (Eds.), *New ideas in optimization* (pp. 245–260). London: McGraw-Hill.
Merz, P., & Huhse, J. (2009). An iterated local search approach for finding provably good solutions for very large tsp instances. In G. Rudolph et al. (Ed.), *Proceedings of PPSN X, Number 5199 in LNCS* (pp. 929–939). Springer Verlag.
Michalewicz, Z. (1992–1996). *Genetic algorithms + data structures = evolution programs* (1st-3rd ed.). New-York: Springer Verlag.
Michalewicz, Z., & Fogel, D. (2004). *How to solve it: Modern heuristics* (2nd ed.). New-York: Springer Verlag.
Michalewicz, Z., Schmidt, M., Michalewicz, M., & Chiriac, C. (2005). A decision-support system based on computational intelligence: A case study. *IEEE Intelligent Systems*, 44–49.
Michalewicz, Z., Schmidt, M., Michalewicz, M., & Chiriac, C. (2006). *Adaptive business intelligence*. New-York: Springer Verlag.
Miller, J. F., & Smith, S. L. (2006). Redundancy and computational efficiency in cartesian genetic programming. *IEEE Transactions on Evolutionary Computation*, 167–174).
Nordin, P. (1997). *Evolutionary program induction of binary machine code and its applications*. Krehl Verlag.
Obayashi, S. (1997). Pareto genetic algorithm for aerodynamic design using the Navier–Stokes equations. In D. Quadraglia., J. Périaux., C. Poloni, & G. Winter (Eds.), *Genetic algorithms and evolution strategies in engineering and computer sciences* (pp. 245–266). John Wiley.
Paechter, B., Rankin, R., Cumming, A., & Fogarty, T. C. (1998). Timetabling the classes of an entire university with an evolutionary algorithm. In T. Bäck., A. Eiben., M. Schoenauer, & H. P. Schwefel (Eds.), *Proceedings of the 5th conference on parallel problems solving from nature*. Springer Verlag.
Price, K. V., Storn, R. M., & Lampien, J. A. (2005). *Differential evolution*. Springer Verlag.
Radcliffe, N. J. (1991). Equivalence class analysis of genetic algorithms. *Complex Systems*, 183–220.
Rechenberg, I. (1972). *Evolutionstrategie: Optimierung technisher systeme nach prinzipien des biologischen evolution*. Stuttgart: Fromman-Hozlboog Verlag.
Rosenman, M. (1999). Evolutionary case-based design. In *Artificial Evolution'99* (pp. 53–72). Springer Verlag, LNCS 1829.
Rudolph, G. (1997). *Convergence properties of evolutionary algorithms*. Hamburg: Kovac.
Saravanan, N., Fogel, D. B., & Nelson, K. M. (1995). A comparison of methods for self-adaptation in evolutionary algorithms. *Biosystems*, 157–166.
Schwefel, H. P. (1981/1995). *Numerical Optimization of Computer Models* (2*nd* edition). New-York: John Wiley & Sons.
Semet, Y., & Schoenauer, M. (2006). On the benefits of inoculation, an example in train scheduling. In *Proceedings of the GECCO'06* (pp. 1761–1768). ACM Press.
Spector, L., & Robinson, A. (2002). Genetic programming and autoconstructive evolution with the push programming language. *Genetic Programming and Evolvable Machines, 1*, 7–40.
Syswerda, G. (1989). Uniform crossover in genetic algorithms. In J. D. Schaffer (Ed.), *Proceedings of the 3rd international conference on genetic algorithms* (pp. 2–9). Morgan Kaufmann.
Syswerda, G. (1991). A study of reproduction in generational and steady state genetic algorithm. In G. J. E. Rawlins (Ed.), *Foundations of genetic algorithms* (pp. 94–101). Morgan Kaufmann.
Yu, T., Davis, L., Baydar, C., & Roy, R. (Eds.). (2008). *Evolutionary computation in practice*. Number 88 in *Studies in computational intelligence*. Springer Verlag.

EVOP

Evolutionary operation.

See

► Quality Control

Ex Ante Forecasts

Forecasts that are made without any knowledge of the period to be forecast.

See

► Forecasting

Exclusive-or Node

In a network, an event (node) that will be realized if one and only one of the arcs leading to it is realized.

See

► Network Planning

Expected Utility Theory

- ▶ Decision Analysis
- ▶ Preference Theory
- ▶ Utility Theory

Expert Systems

Clyde W. Holsapple[1] and Andrew B. Whinston[2]
[1]University of Kentucky, Lexington, KY, USA
[2]The University of Texas at Austin, Austin, TX, USA

Introduction

Devising computer-based systems that can solve problems by reasoning about facts and assertions has been a central, ongoing quest in the artificial intelligence field. By the early 1970s, research into these reasoning systems had begun to focus on systems that could solve difficult problems in narrow problem domains such as diagnosing diseases, assessing chemical structures of unknown molecules, determining ore deposits in geological sites, and solving applied mathematical problems. These systems came to be known as expert systems, because they solve problems that would otherwise require services of experts in their respective problem areas. Perhaps the best known of these early expert systems is MYCIN, whose approach to diagnosing blood infections is documented in Buchanan and Shortliffe (1984). Descriptions of other pioneering expert systems such as DENDRAL, MACSYMA, and PROSPECTOR can be found in Barr and Feigenbaum (1982).

By the early 1980s, the focus in expert system (ES) research had shifted from demonstrating the feasibility and efficacy of such systems to the identification of tools and methods that could facilitate their development. Each of the pioneering expert systems was custom-built, requiring considerable expense and years of development by specialists in artificial intelligence. If expert systems were to come into widespread use, it was clear that faster and less costly means for creating them had to be found. This search has been largely successful, spawning a host of commercially available, computer-based tools for ES development and leading to the creation of specific methods for guiding the process of ES development. These tools and methods have been instrumental in the growing number of expert systems used in such application areas as engineering, manufacturing, finance, and business administration (Blanning 1984; Mockler 1989; Tyran and George 1993; Liebowitz 1998; Liao 2005). When the result of expert system execution is used in decision making, which is very often the case, the expert system functions as an artificially intelligent decision support system (Holsapple and Whinston 1986, 1996).

Specific examples of expert system applications range widely from agricultural loan evaluation (Bryant 2001), to prioritizing sewer inspections (Hahn 2002), to production system advising (Wagner et al. 2003), to architectural design modification (Bachman 2004), to analysis of disturbances in the quality of power systems (Reaz et al. 2007), to machine vibration analysis (Ebersbach and Peng 2008), to interpreting ECG readings (Mahesh et al. 2009), and so forth. Expert systems have come to be very widely used on the Web (Duan et al. 2005). Being routinely embedded into Web-based applications, they are rarely thought about as being expert systems. To the extent they are even named at all, terms such as recommenders or advisors are much more common.

Two prerequisites for assessing ES possibilities are an understanding of the nature of expert systems and an appreciation of how they can be developed. The general nature of expert systems is described first, including characterizations of ES functions, architecture, and operation. Then, ES development is examined in terms of methodological issues and classes of available tools.

General Nature of an Expert System

An expert system functions as a readily available substitute for some source of expertise that cannot always be consulted in a facile, timely, and affordable manner. For instance, consider the case of a person who is an expert in some problem domain, such as financial planning. This human source of

expertise about financial planning is able to accept requests for advice about specific problems in the domain, reason with the expertise to produce recommendations, communicate the resultant advice, and explain the rationale underlying that advice. A computer-based system that can perform these same functions, giving recommendations and explanations comparable to those of the expert, is called an expert system. Not all ESs substitute for an individual human expert. An ES can also be a surrogate for a group of experts, multiple individual experts, the expertise embodied in a set of historical data, or expertise revealed in the behavior of some non-human system.

Expert systems offer many potential advantages over relying on the original source of expertise for advice (Holsapple and Whinston 1986, 1996). Unlike a human expert, an ES does not sleep, become ill, take vacations, have a bad day, forget, require compensation, become tied up with more important matters, or retire. From an organization's viewpoint, the exercise of building an ES results in a formalization and preservation of expertise. It yields an advice giver that can be readily replicated for simultaneous use at geographically diverse sites, ensuring consistency of recommendations. Holsapple and Whinston (1990) argue that ESs can be instrumental in implementing competitive strategies.

The functioning of an ES is based on three major components: a user interface (having two parts: a language system and a presentation system), an inference engine, and a body of stored knowledge that forms the basis for reasoning about problems in some domain of interest. The user interface is that part of an ES that its user (e.g., a person seeking advice) directly experiences. It accepts a user's characterization of a specific problem. It asks for clarifications of that characterization, as needed. It presents the ES's advice to the user about treating the problem. The user interface also accepts user requests for justifying advice and presents those justifications to a user. Expert systems can vary widely in terms of the style and sophistication of their user interfaces, even when the other two architectural components are fixed.

A second component of ES architecture is the knowledge store it possesses. In ES parlance, this is often called a knowledge base . It typically holds two distinct kinds of knowledge: descriptive and reasoning. Descriptive knowledge is concerned with describing states of the world (e.g., that revenue was $10 million last year or is expected to be $15 million next year). Reasoning knowledge is concerned with specifying what conclusion is valid when a particular situation is known to exist (e.g., that an unemployment rate of over 8% warrants a certain reduction in revenue expectations). Additional types of knowledge can be found in the knowledge bases of some ESs (Holsapple and Whinston 1996).

There is more than one way to represent each type of knowledge stored in an ES. Descriptive knowledge may be simply represented as values of state variables, often called attribute-value pairs (e.g., the revenue attribute or variable has a value of $10 million). Or, such pieces of descriptive knowledge may be structured into database records, frames, semantic nets, arrays, spreadsheet cells, and other computer-based organizations. Similarly, pieces of reasoning knowledge are subject to multiple representation modes. One commonly used approach involves the use of rules. Each rule has a premise, characterizing some situation, and a conclusion, indicating what actions can be taken (e.g., what changes can be made to state variable values) if the situation is determined to exist. A variety of rule representation languages exist. They differ in terms of style, flexibility, and power of representation. Some such differences are surveyed in Mockler (1989).

Inference Engine

At the heart of general ES architecture is an inference engine. This is a software component that reasons with the stored knowledge of an ES to derive advice corresponding a user's problem statement. It also tracks the flow of reasoning about the problem, as a basis for justifications presented via the user interface. Clearly, an ES's inference engine must be compatible with a) the particular representation language used to specify stored descriptive and reasoning knowledge, b) the user interface's interpretations of user requests, and c) the user interface's ability to package inference engine results. From one ES to another, inference engines vary not only to ensure compatibility with different user

interface and knowledge representation conventions, but also with respect to how they reason.

Two prominent kinds of reasoning approaches are forward chaining and backward chaining. In either case, the inference engine uses rules to establish values for variables whose states are unknown at the outset of a consultation. These values constitute the raw form of the advice that is ultimately packaged for presentation to a user. The main difference between the two kinds of reasoning is the progression of processing whereby unknown variables become known. In either case, if an ES holds insufficient knowledge to solve a problem, the inference engine will fail in its attempt to establish values for unknown variables.

In the forward chaining case, an inference engine examines the premise of each rule. If the premise is true, then the actions specified in the rule's conclusion are performed (i.e., the rule is fired). Thus, firing a rule can result in changes to values of variables, including the assignment of values to previously unknown variables. After every rule has been examined in this way, the inference engine makes a second pass through the rules. If additional rules are fired in this pass, the process continues with a third pass, and so forth. Processing stops when no further rules are fired in a pass or when some other terminal condition is satisfied (e.g., a value has been established for some designated unknown variable).

In contrast, backward chaining is a more goal-directed approach to reasoning. It considers a rule's conclusion before trying to evaluate the premise. Establishing a value for a specific unknown variable is the inference engine's overall goal. In its effort to meet that goal, the inference engine identifies the subset of rules whose conclusions could affect the goal variable's value. These are called candidate rules. In considering a candidate rule, the inference engine attempts to evaluate the premise. If this evaluation is impossible because the premise involves unknown variables, then each of those variables successively becomes the new current goal. The inference engine performs backward chaining for the current goal variable, identifying its candidate rules and attempting to evaluate their premises. When a rule's premise is found to be true, the rule is fired. When a premise is found to be false, the inference engine proceeds to another candidate rule. This basic processing pattern continues recursively until a value is established for the consultation's overall goal variable or until that variable's candidate rules are exhausted (possibly without reaching a solution).

There are many variations to each of these two reasoning approaches, affecting both the speed and the results of inference engine operation. One kind of variation involves the degree of reasoning rigor. These are variations in how exhaustive inference engines are in making passes through a rule set or in considering candidate rules. Another variation concerns rule selection order. That is, in what sequence does an inference engine process rules within a pass or within a candidate rule subset? There is also considerable room for variation in the strategies inference engines use for evaluating a premise (e.g., the order for considering conditions in a compound premise). Inference engines can vary greatly in their treatments of uncertainties about variable values and rule efficacy. Some ignore the possibility of uncertainties, while others use specific algebras to combine certainty factors in an effort to qualify the resultant advice. Holsapple and Whinston (1986, 1996) present an extensive discussion of such variations.

Expert System Development

Tools for building expert systems fall into three major categories: programming languages, shells, and integrated environments. In using the former, an ES developer (often called a knowledge engineer) designs and programs the inference engine and the user interface. Also, storage structures for holding reasoning and descriptive knowledge must be designed so their contents are accessible to the inference engine. Appropriate knowledge must then be stored in such structures. A shell removes much of this work from an ES developer, but also reduces the developer's flexibility. With a shell, the developer has a ready-made inference engine, user interface, and knowledge storage structure. Thus, the main development task consists of putting the appropriate knowledge into that structure. Shells also commonly give developers some facilities for customizing the user interface inputs and outputs. Some give developers a modicum of control over the inference engine's reasoning behavior (e.g., over the degree of rigor, selection order, treatment of uncertainty). Shell inference engines can often interface to other pieces of software (e.g., use spreadsheet data).

An integrated environment for ES development has all the facilities of a shell, plus other computing

capabilities normally found in separate software tools. That is, the inference engine is enhanced to accomplish kinds of processing other than reasoning: database management, spreadsheet processing, model management, forms handling, graph generation, and so forth. Such capabilities can be exercised in the course of rule processing. Conversely, consultation can occur in the context of one of these other kinds of processing. Such tools allow the creation of ESs that more closely approximate human experts, who are not limited to reasoning about a problem, but can also do data retrieval, extensive calculations, fancy presentation, and so on. A checklist for tool selection, plus an in-depth examination of an integrated environment, can be found in Holsapple and Whinston (1986). Mockler (1989) provides a comparative feature survey of representative commercial expert system development tools.

Regardless of the tool used, a developer faces the task of managing the ES development project. The books of Buchanan and Shortliffe (1984), Hayes-Roth, et al. (1983), and Liebowitz (1998) contain valuable insights into the methodology of ES development. The ES development cycle introduced by Holsapple and Whinston (1986, 1996) is typical of several that appear in the literature. One aspect of development that has received considerable attention is the phenomenon of knowledge acquisition (KA). This is concerned with the activity of eliciting reasoning knowledge from a source (e.g., human expert), perhaps structuring/analyzing it, and representing it in a form that can be directly stored in the knowledge base of an ES for later use in making inferences about specific problems posed by a user.

A representative overview of KA issues, methods, and tools was provided by Kidd (1987). The methods include such techniques as structured interviewing and protocol analysis. The KA tools are primarily induction mechanisms that attempt to acquire general domain knowledge from sets of specific examples of expert behavior. Dhaliwahl and Benbasat (1990) introduced a variable-oriented framework for guiding empirical research that evaluates performance of such methods and tools. Holsapple et al. (1993) provide a summary of basic theoretical and empirical KA developments.

Expert System Extensions

Over the years, various extensions to the basic idea of rule-based expert systems have arisen. Some of these involve alternative approaches to representing and processing reasoning knowledge, such as the use of case-based reasoning (Waston 1995; Bergmann et al. 2003). Other extensions imbue reasoning systems with learning capabilities, by using such techniques as skill refinement (Deng et al. 1990), neural networks (Gallant 1993; Im and Park 2007), or genetic algorithms (Holland 1992; Dehuri and Mall 2006). Beyond certainty factors, other methods to deal with uncertainties in the course of reasoning include Dempster-Shafer techniques of evidence (Yager 2008). In a somewhat related vein, expert systems can be developed to accommodate representation and reasoning for non-discrete descriptive knowledge in the form of fuzzy variables (Siler and Buckley 2005). Another extension is concerned with the coordination of multiple expert systems, each having reasoning knowledge that could be relevant to solving a problem at hand. Holsapple et al. (1997) have advanced a market-based mechanism to do this coordination in a way that causes the community of expert systems to learn (i.e., improve its performance) over time.

Further Reading

Aside from the growing number of books dealing with expert systems, there are several scholarly journals emphasizing ES topics. These include *Expert Systems*; *Expert Systems with Applications*; *IEEE Intelligent Systems*; *and Intelligent Systems in Accounting, Finance, and Management*. ES topics are also covered in more general journals of artificial intelligence (e.g., *Artificial Intelligence*), as well as journals concerned with computer systems for business (e.g., *Decision Support Systems*) and engineering (e.g., *IEEE Transactions on Data and Knowledge Engineering*). Links to numerous World Wide Web sites dealing with expert systems and artificial intelligence can be found.

See

► Artificial Intelligence
► Decision Support Systems (DSS)

References

Bachmann, F., Bass, L., Klein, M., & Shelton, C. (2004). Experience using an expert system to assist an architect in designing for modifiability. *Proceedings of the IEEE/IFIP Conference on Software Architecture*, Pittsburgh, PA, pp. 281–284

Barr, A., & Feigenbaum, E. A. (Eds.). (1982). *The handbook of artificial intelligence*. Los Altos, CA: William Kaufmann.

Bergmann, R., Althoff, K.-D., Breen, S., Göker, M., Manago, M., Traphöner, R., & Wess, S. (2003). *Developing industrial case-based reasoning applications. The INRECA methodology*. Berlin: Springer.

Blanning, R. W. (1984). Management applications of expert systems. *Information and Management, 6*, 311–316.

Bryant, K. (2001). ALEES: An agricultural loan evaluation expert system. *Expert Systems with Applications, 21*(2), 75–85.

Buchanan, B. G., & Shortliffe, E. H. (1984). *Rule-based expert systems: The MYCIN experiments of the Stanford heuristic programming project*. Reading, MA: Addison-Wesley.

Dehuri, S., & Mall, R. (2006). Predictive and comprehensible rule discovery using a multi-objective genetic algorithm. *Knowledge-Based Systems, 19*(6), 413–421.

Deng, P., Holsapple, C. W., & Whinston, A. B. (1990). A skill refinement learning model for rule-based expert systems. *IEEE Expert, 5*(2), 15–28. 53179 Abstract.

Dhaliwal, J. S., & Benbasat, I. (1990). A framework for the comparative evaluation of knowledge acquisition tools and techniques. *Knowledge Acquisition, 2*, 145–166.

Duan, Y., Edwards, J. S., & Xu, M. X. (2005). Web-based expert systems: Benefits and challenges. *Information & Management, 42*(6), 799–811.

Ebersbach, S., & Peng, Z. (2008). Expert system development for vibration analysis in machine condition monitoring. *Expert Systems with Applications, 34*(1), 291–299.

Gallant, S. I. (1993). *Neural network learning and expert systems*. Cambridge, MA: MIT Press.

Hahn, M. A., Palmer, R. A., Merrill, S., & Lukas, A. B. (2002). Expert system for prioritizing the inspection of sewers: Knowledge base formulation and evaluation. *Journal of Water Resource Planning and Management, 128*(2), 121–129.

Hayes-Roth, F., Lenat, D. B., & Waterman, D. A. (Eds.). (1983). *Building expert systems*. Reading, MA: Addison-Wesley.

Holland, J. F. (1992). Genetic algorithms. *Scientific American*, July.

Holsapple, C. W., Lee-Post, A., & Otto, J. (1997). A machine learning method for multi-expert decision support. *Annals of Operations Research, 75*, 171–188.

Holsapple, C. W., Raj, V., & Wagner, W. (1993). Knowledge acquisition: Recent theoretic and empirical developments. In C. Holsapple & A. Whinston (Eds.), *Recent developments in decision support systems* (pp. 295–312). Berlin: Springer.

Holsapple, C. W., & Whinston, A. B. (1986). *Manager's guide to expert systems*. Homewood, IL: Dow Jones-Irwin.

Holsapple, C. W., & Whinston, A. B. (1990, January). Business expert systems — gaining a competitive edge. *Proceedings of Hawaiian International Conference on Systems Sciences*, Kona, Hawaii.

Holsapple, C. W., & Whinston, A. B. (1996). *Decision support systems: A knowledge-based approach*. St. Paul, MN: West Publishing.

Im, K. H., & Park, S. C. (2007). Case-based reasoning and neural network based expert system for personalization. *Expert Systems with Applications, 32*(1), 77–85.

Kidd, A. (Ed.). (1987). *Knowledge elicitation for expert systems: A practical handbook*. New York: Plenum Press.

Liao, S. (2005). Expert system methodologies and applications—a decade review from 1995 to 2004. *Expert Systems with Applications, 28*(1), 93–103.

Liebowitz, J. (Ed.). (1998). *The handbook of applied expert systems*. Boca Raton, FL: CRC Press.

Mahesh, V., Kandaswamy, A., & Venkatesan, R. (2009). A rule-based expert system for ECG analysis. *International Journal of Engineering and Technology, 1*(3), 194–200.

Mockler, R. J. (1989). *Knowledge-based systems for management decisions*. Englewood Cliffs, NJ: Prentice-Hall.

Reaz, M. B. I., Choong, F., Sulaiman, M. S., Mohd-Yasin, F., & Kamada, M. (2007). Expert system for power quality disturbance classifier. *IEEE Transactions on Power Delivery, 22*(3), 1979–1988.

Siler, W. A., & Buckley, J. J. (2005). *Fuzzy expert systems and fuzzy reasoning*. Hoboken, NJ: Wiley.

Tyran, C. K., & George, J. F. (1993). The implementation of expert systems: A survey of successful implementations. *DATABASE, 24*, 5–15.

Wagner, W. P., Chung, Q. B., & Najdawi, M. K. (2003). The impact of problem domains and knowledge acquisition techniques: A content analysis of P/OM expert system case studies. *Expert Systems with Applications, 24*, 79–86.

Waston, I. (Ed.). (1995). *Progress in case-based reasoning*. Berlin: Springer.

Yager, R. R. (2008). *Decision making under Dempster–Shafer uncertainties*. Berlin: Springer.

Exploratory Modeling and Analysis

Steve Bankes[1], Warren E. Walker[2] and Jan H. Kwakkel[2]

[1]BAE Systems, Arlington, VA, USA

[2]Delft University of Technology, Delft, The Netherlands

Introduction

Exploratory Modeling and Analysis (EMA) is a research methodology that uses computational experiments to analyze complex and uncertain systems (Bankes 1993, 1994). EMA can be understood as searching or sampling over an ensemble of models that are plausible given a priori knowledge, or are otherwise of interest. This ensemble

may often be large or infinite in size. Consequently, the central challenge of exploratory modeling is the design of search or sampling strategies that support valid conclusions or reliable insights based on a limited number of computational experiments.

EMA can be contrasted with the use of models to predict system behavior, where models are built by consolidating known facts into a single package (Hodges 1991). When experimentally validated, this single model can be used for analysis as a surrogate for the actual system. Examples of this approach include the engineering models that are used in computer-aided design systems. Where applicable, this consolidative methodology is a powerful technique for understanding the behavior of complex systems. Unfortunately, for many systems of interest, the construction of models that may be validly used as surrogates is simply not a possibility. This may be due to a variety of factors, including the infeasibility of critical experiments, impossibility of accurate measurements or observations, immaturity of theory, openness of the system to unpredictable outside perturbations, or nonlinearity of system behavior, but is fundamentally a matter of not knowing enough to make predictions (Campbell et al. 1985; Hodges and Dewar 1992; Pilkey and Pilkey-Jarvis 2007). For such systems, a methodology based on consolidating all known information into a single model and using it to make best estimate predictions can be highly misleading.

EMA can be useful when relevant information exists that can be exploited by building models, but where this information is insufficient to specify a single model that accurately describes system behavior. In this circumstance, models can be constructed that are consistent with the available information, but such models are not unique. Rather than specifying a single model and falsely treating it as a reliable image of the target system, the available information is consistent with a set of models, whose implications for potential decisions may be quite diverse. A single model run drawn from this potentially infinite set of plausible models is not a prediction; rather, it provides a computational experiment that reveals how the world would behave if the various guesses any particular model makes about the various unresolvable uncertainties were correct. EMA is the explicit representation of the set of plausible models, the process of exploiting the information contained in such a set through a large number of computational experiments, and the analysis of the results of these experiments.

A set, universe, or ensemble of models that are plausible or interesting in the context of the research or analysis being conducted is generated by the uncertainties associated with the problem of interest, and is constrained by available data and knowledge. EMA can be viewed as a means for inference from the constraint information that specifies this set or ensemble. Selecting a particular model out of an ensemble of plausible ones requires making suppositions about factors that are uncertain or unknown. One such computational experiment is typically not that informative (beyond suggesting the plausibility of its outcomes). Instead, EMA supports reasoning about general conclusions through the examination of the results of numerous such experiments. Thus, EMA can be understood as search or sampling over the ensemble of models that are plausible given a priori knowledge.

Central Problems and Solutions

Inferring global properties of a large or infinite set from a finite sample is not a deductive process but requires some combination of inductive and abductive inference along with effective data mining and visualization tools. Consequently, EMA is computationally a more difficult problem than any specific question of deductive inference, and produces results that are more contextual and provisional. How to cleverly select the finite sample of models and cases to examine from the large or infinite set of possibilities is one of the major issues to be addressed in any EMA application. A wide range of research strategies are possible, including structured case generation by Monte Carlo, Latin Hypercube, or factorial experimental design methods, search for extremal points of cost functions, sampling methods that search for regions of "model space" with qualitatively different behavior, or combining human insight and reasoning with formal sampling mechanisms. Computational experiments can be used to examine ranges of possible outcomes, to suggest hypotheses to explain puzzling data, to discover significant phases, classes, or thresholds among the ensemble of plausible models, or to support reasoning based upon an analysis of risks, opportunities, or

scenarios. Exploration can be over both real-valued parameters and non-parametric uncertainty, such as that involving different graph structures, functions, problem formulations, or model formulations.

In making policy decisions about complex and uncertain problems, EMA can provide new knowledge, even where validated models cannot be constructed. A simple example is the use of models as existence proofs or hypothesis generators. Demonstrating a single plausible model/case with counterintuitive properties can beneficially change the nature of a policy discussion. Another example is the use of multiple models that capture different framings of the same policy problem. Instead of debating which is the right model, the policy debate can shift to the identification of policies that produce satisfying results across the different models. Another simple example of potentially credible inductive inference from model exploration is provided by situations where risk aversion is prudent. Here, an exploration that develops an assortment of plausible worst case failure modes can be very useful for designing hedging strategies. This is true even if models are not validated and sensitivities are unknown. Other examples of useful research strategies include the search for special cases where small investments could (plausibly) produce large dividends, or extremal cases (either best or worst) where the uncertainties are all one sided and a fortiori arguments can be used. All these examples depend on the fact that partial information can inform policy even when prediction and optimization are not possible. The space of models and associated computational experiments can be searched for examples with characteristics that are useful in choosing among alternative policies. The search for information of use in answering policy questions can often be served by the discovery of thresholds, boundaries, or envelopes in a space of models that decompose the entire space into sub-spaces with different properties. For example, EMA could seek to discover which models or initial states have stable or chaotic dynamics, or the search could have the goal of discovering which regions in model space favor either of two alternative policies.

EMA will typically result in a very large number of model runs. The resulting outputs must be analyzed, and displays need to be made to communicate the results to analysts and decision makers. The EMA practitioner is not interested in finding a single best policy given a validated predictive system model, but wants to display the pattern of policy performance over the entire uncertainty space of possible system models. Successfully applied algorithms in the context of EMA include the Patient Rule Induction Method (PRIM) and Classification and Regression Trees (CART) (Breiman et al. 1984; Friedman and Fisher 1999; Lempert et al. 2008; Agusdinata 2008). Advances in machine learning and data mining have generated many more algorithms that can support EMA, such as Self Organizing Maps (Kohonen 2001), (t-distributed) Stochastic Nearest Neighbor Embedding (van der Maaten and Hinton 2008), and Support Vector Machines (Vapnik 1995). Increasingly, such algorithms are available in standard statistical data analysis software packages (e.g. SPSS). The Evolving Logic company produced a software environment called the Computer Assisted Reasoning system (CARs), which supports the generation of the EMA cases to be run and the manipulation and display of the results of the runs.

EMA has proven to be a very powerful approach to the discovery of robust decisions and the development of adaptive policies. Adaptive policies are based on explicit recognition that accurate prediction is impossible in light of the many uncertainties that are present. The goal of adaptive policies is to allow implementation to begin prior to the resolution of all major uncertainties, with the policy being adapted over time based on new knowledge. Adaptive policies combine actions that are time urgent with those that make important commitments to shape the future, preserve needed flexibility for the future, and protect the policy from failure (Walker et al. 2001). EMA can support the development of adaptive policies by using the set of plausible models or plausible futures as a challenge set. Through searching for conditions under which given policies fail, these policies can iteratively be improved, resulting in adaptive policies that are robust against the full range of foreseeable future situations (Lempert et al. 2003; Agusdinata 2008). In such applications, EMA provides an important alternative to specifying policies through optimization. There is a close relationship between these computational approaches and emerging adaptive business practices, such as discovery driven planning (McGrath and MacMillan 1995; McGrath and MacMillan 2009) and real options (Amram and Kulatilaka 1999).

EMA has also been used successfully for scenario discovery. Scenario discovery is a model driven approach that builds on the intuitive logic school in scenario planning (Bryant and Lempert 2010). The aim of scenario discovery is to analyze the results from a series of computational experiments in order to reveal which combinations of hypotheses and guesses were responsible for generating the results of interest. Results of interest can be identified based on the performance of candidate policies, but other criteria can also be used. One common use of scenario discovery is to identify combinations of external events that would lead to the failure of the policy being investigated. For discovering the combinations of hypotheses and guesses responsible for the results of interest, both CART and PRIM can be used (Lempert et al. 2008). Scenario discovery has been used in the context of water resource management in California (Groves and Lempert 2007) for evaluating alternative policies considered by the U.S. Congress while debating reauthorization of the Terrorism Risk Insurance Act (Dixon et al. 2007), and for assessing the impact of a renewable energy requirement in the U.S. (Groves and Lempert 2007).

Applications of EMA

Exploratory modeling can be driven by data, a question or decision, or by the needs of model development. Data-driven exploration can be used to support model specification — exploring alternative model structures that might be used to explain a dataset. Or, it can provide an alternative to maximal likelihood or maximal entropy approaches to model estimation by supporting, for example, the visualization of level sets in likelihood surfaces. Question driven exploration begins with a question to answer (e.g., what policy should the government pursue regarding global warming?) and addresses this question by searching over an ensemble of models and cases believed to be plausible in order to inform the answer. Question driven exploration provides an alternative to supporting decision making through forecasting or prediction. EMA also provides a strong alternative approach to model development in allowing guesses and disagreements about uncertain modeling details to be avoided during the process of programming and delayed until the process of model use where these guesses can be motivated by the actual strategy of model based problem solving.

Although the practice of EMA is under continuing, it has been used on variety of decision-making problems. Lempert et al. (2003) applied it to climate change problems in an effort to identify policy options that are, on the one hand, acceptable to a wide variety of countries, depending on their state of development and their belief about climate change, and, on the other hand, are robust across a wide variety of different plausible future climate change developments. In this study, a system dynamics model was used to generate a wide variety of futures. Each future arose out of a set of beliefs about the different factors and their relative magnitudes that contribute to climate change. Next, a variety of policy options was identified, and the performance of these policies in the different futures was calculated. Using the decision-theoretic minimax criterion, the robustness of the different policies can be assessed, and improved, in order to come to a policy that is robust across as many futures as possible. Or, as is often the case when different uncertainties and risks are being considered, no policy may be dominant. In that case, a priori assignments of values can be avoided by presenting decision makers with sets of non-dominated policies and providing analytic support for understanding how choices among them trade off among risk categories.

Another EMA application is from the field of energy generation. Agusdinata (2008) studied how CO2 emissions could be reduced in the Dutch household sector. The Dutch national government wants to reduce carbon emissions from Dutch households by 80% in 2050 compared to 1990. The extent to which that goal can be realized depends on a wide range of factors and actors. For example, the energy generation depends on the layout of the electricity and gas networks. Depending on how these networks evolve, certain options become available or are excluded. Agusdinata (2008) used a 'system of systems' approach to model these complex interdependencies. He made 100,000 model runs across a variety of uncertainties and analyzed them using a CART Tree. This allowed decision makers to see the different routes by which they could reach their goal, depending on how, for example, population would evolve.

A third area in which EMA has been applied is transport planning. Agusdinata et al. (2009) report a case study related to intelligent speed limiters, which is similar in approach to Agusdinata's (2008) household energy case. The field of airport strategic planning has

been investigated by Kwakkel et al. (2010). In this study, several tools from the U.S. Federal Aviation Administration for calculating airport performance were integrated into a single fast model that calculates airport performance in terms of noise, emissions, external safety, and capacity. This component is complemented with a variety components that model different exogenous developments, such as demand, technological development, and demographics. For each type of exogenous development, a set of representations are available. Depending on how these are combined, 48 structurally different models can be specified. Kwakkel, et al. explored the performance of adaptive strategies versus static Master Plans across these 48 structures and their associated parametric uncertainties. It was shown that dynamic adaptive strategies outperformed static Master Plans in practically all plausible situations.

Other EMA applications are reported in Bankes and Margoliash (1993), Bankes (1994), Park and Lempert (1998), Brooks et al. (1999), Lempert et al. (2003), Pruyt and Hamarat (2010a, b), and Hamarat and Pruyt (2011a, b).

See

- ▶ Deep Uncertainty
- ▶ Model Accreditation
- ▶ Practice of Operations Research and Management Science
- ▶ Public Policy Analysis
- ▶ Soft Systems Methodology
- ▶ Verification, Validation, and Testing of Models

References

Agusdinata, D. B. (2008). *Exploratory modeling and analysis: A promising method to deal with deep uncertainty*. Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands.

Agusdinata, D. B., van der Pas, J. W. G. M., Marchau, V. A. W. J., & Walker, W. E. (2009). Multi-criteria analysis for evaluating the impacts of intelligent speed adaptation. *Journal of Advanced Transportation, 43*(4), 413–454.

Amram, M., & Kulatilaka, N. (1999). *Real options: Managing strategic investment in an uncertain world*. Boston, MA: Harvard Business School Press.

Bankes, S. (1993). Exploratory modeling for policy analysis. *Operations Research, 41*, 435–449.

Bankes, S. (1994). Exploring the foundations of artificial societies: Experiments in evolving solutions to N-player prisoner's dilemma. In R. Brooks & P. Maes (Eds.), *Artificial life IV*. Cambridge, MA: MIT Press.

Bankes, S., & Margoliash, D. (1993). Parametric modeling of the temporal dynamics of neuronal responses using connectionist architectures. *Journal of Neurophysiology, 69*, 980–991.

Breiman, L., Friedman, J. H., Olshen, C. J., & Stone, C. J. (1984). *Classification and regression trees*. Monterey, CA: Wadsworth.

Brooks, A., Bennett, B., & Bankes, S. (1999). An application of exploratory analysis: The weapon mix problem. *Military Operations Research, 4*(1), 67–80.

Bryant, B. P., & Lempert, R. (2010). Thinking inside the box: A participatory computer assisted approach to scenario discovery. *Technological Forecasting and Social Change, 77*, 34–49.

Campbell, D., Crutchfield, J., Farmer, D., & Jen, E. (1985). Experimental mathematics: The role of computation in nonlinear science. *Communications of the ACM, 28*, 374–384.

Dixon, L., Lempert, R. J., LaTourrette, T., & Reville, R. T. (2007). *The federal role in terrorism insurance: Evaluating alternatives in an uncertain world, MG-679-CTRMP*. Santa Monica, CA: RAND.

Friedman, J. H., & Fisher, N. I. (1999). Bump hunting in high-dimensional data. *Statistics and Computing, 9*, 123–143.

Groves, D. G., & Lempert, R. (2007). A new analytic method for finding policy-relevant scenarios. *Global Environmental Change, 17*, 73–85.

Hamarat, C., & Pruyt, E. (2011a). Energy transitions: Adaptive policy making under deep uncertainty. *Proceedings of The 4th International Seville Conference on Future-Oriented Technology Analysis (FTA)*, Seville, Spain.

Hamarat, C., & Pruyt, E. (2011b). Exploring the future of wind-powered energy. *Proceedings of The 29th International Conference of the System Dynamics Society*, Washington, DC

Hodges, J. S. (1991). Six (or so) things you can do with a bad model. *Operations Research, 39*, 355–365.

Hodges, J. S., & Dewar, J. A. (1992). *Is it you or your model talking? R-4114*. Santa Monica, CA: RAND.

Kohonen, T. (2001). *Self-organizing maps* (3rd ed.). London: Springer.

Kwakkel, J. H., Walker, W. E., & Marchau, V. A. W. J. (2010). *Assessing the efficacy of adaptive airport strategic planning: Results from computational experiments, world conference on transport research* (pp. 11–15). Porto, Portugal, July 2010.

Lempert, R. J., Bryant, B. P., & Bankes, S. C. (2008). *Comparing algorithms for scenario discovery, WR-557-NSF*. Santa Monica, CA: RAND.

Lempert, R. J., Popper, S. W., & Bankes, S. C. (2003). *Shaping the next one hundred years: New methods for quantitative long-term strategy analysis, MR-1626-RPC*. Santa Monica, CA: RAND.

McGrath, R. G., & MacMillan, I. C. (1995). Discovery driven planning. *Harvard Business Review, 73*(4), 44–54.

McGrath, R. G., & MacMillan, I. C. (2009). *Discovery driven growth: A breakthrough process to reduce risk and seize opportunity*. Boston, MA: Harvard Business.

Park, G., & Lempert, R. (1998). *The class of 2014: Preserving access to California higher education, MR-971*. Santa Monica, CA: RAND.

Pilkey, O. H., & Pilkey-Jarvis, L. (2007). *Useless arithmetic: Why environmental scientists can't predict the future*. New York: Columbia University Press.
Pruyt, E.. & Hamarat C. (2010a). The concerted run on the DSB bank: An exploratory system dynamics approach, In *Proceedings of the 28th International Conference of the System Dynamics Society*. Seoul, Korea.
Pruyt, E., & Hamarat, C. (2010b). The influenza A(H1N1)v pandemic: An exploratory system dynamics approach. *Proceedings of the 28th International Conference of the System Dynamics Society*. Seoul, Korea.
van der Maaten, L. J. P., & Hinton, G. E. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*, 2579–2605.
Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer.
Walker, W. E., Rahman, S. A., & Cave, J. (2001). Adaptive policies, policy analysis, and policymaking. *European Journal of Operational Research, 128*(2), 282–289.

Exponential Arrivals

When customers interarrival times to a queueing system are defined by a sequence of independent and identically distributed exponential random variables. If a system has exponential interarrival times with distribution function $A(t) = 1 - \exp(-\lambda t)$, then the number of arrivals to the system in any period of time t has the Poisson distribution with probability function $p_n(t) = \exp(-\lambda t)(\lambda t)^n/n!$.

See

- ▶ Poisson Arrivals
- ▶ Queueing Theory

Exponential Smoothing

Robert G. Brown
Materials Management Systems, Thetford Center, VT, USA

Introduction

Exponential smoothing is a technique for revising an estimate of the average of a time series to extrapolate as a forecast. It was first formalized by R.G. Brown (about 1944) with continuous variables in the analysis of a ball-disc integrator used in a naval fire control device. It was later applied, also by R.G. Brown (1959), with discrete observations in the early 1950s.

Exponential smoothing with discrete (usually monthly) observations of demand had considerable appeal for inventory control because it was possible to revise the forecasts for all products in an inventory in less than 30 days with unit record (punched card) equipment. The formula for revising the estimate of the average is

$$\begin{aligned} new\,forecast &= old\,forecast + \alpha \times (error) \\ &= old\,forecast + \alpha \times (latest\,observation \\ &\quad - old\,forecast) \\ &= (1-\alpha) \times old\,forecast + \alpha \\ &\quad \times (latest\,observation). \end{aligned}$$

The smoothing constant α was originally set to 0.1, not from any theoretical considerations, but because one can multiply on unit-record equipment merely by moving the wires on the plug board one place to the left and adding. But, clearly, better approaches for estimating a good value to use for α would come over time.

Later, Weiner (1949) showed that an estimate of the average of a time series gave the minimum squared error when one uses some optimum set of weights. Box and Jenkins (1970) demonstrated a rigorous procedure for finding that optimum set of weights. The optimum weights often decline more or less geometrically with age. Indeed, for short series the wholly empirical technique of exponential smoothing was usually indistinguishable from the optimum weights.

In the late 1950s, Brown (1963) extended the model to include a secular trend. The first model used single smoothing and double smoothing. Brown (1967) showed that this was equivalent to smoothing the values of the coefficients in a polynomial of any degree. This led to the generalization to include complex polynomials that could be interpreted as Fourier series to approximate repeatable seasonal variation.

Winters (1962) developed an elaborate simulation to find the best value of three smoothing constants, for level, trend, and seasonal profile respectively. Simulations with different models and different smoothing constants are usually misleading because the sampling error with short series is larger than the effect being sought. Since exponential smoothing learns, several implementers let the coefficients in the

model start from arbitrary values. There is a problem in that the rate of learning takes much longer than the normal span of patience of people who need a good forecast now.

The initial values of the coefficients should be estimated by regression on available history. (For new products there will be other products in the inventory that serve the same market which can be used as analogs). Since the Fourier series is an orthogonal basis, one can fit all the terms up to the Nyquist frequency (at least two observations per cycle for the highest frequency) and reject harmonics that are not significant under a chi-square test with two degrees of freedom.

Decisions based on a forecast need information about the probability distribution of forecast errors. The form of the distribution may be Gaussian, but there are not infrequent cases where the errors are bounded below, with a long upper tail. Therefore it is advisable to check the distribution form that is appropriate to the actual data. Usually one parameter, the variance, is sufficient to develop the model of probabilities.

Brown (1959) proposed the use of the Mean Absolute Deviation (MAD) as a measure of dispersion. On unit-record equipment, it is simple to measure the absolute deviation — leave out the wire that carries sign. If the form of the distribution is exactly normal the standard deviation is 1.25 times the MAD. However in actual data the ratio has been observed to be anywhere from 1 to 1.7. Thus it is prudent to measure the Mean Squared Error (MSE). The MSE can be revised with each new observation by applying exponential smoothing to the square of the error in the most recent forecast.

A theoretical model of a time series often is seriously different from actual data. Sales are distorted by promotions, federal regulations, competition, weather, and errors in recording. During the process of revising the forecast it is advisable to produce exception reports. The demand filter reports data that are more than K standard deviations from the most recent forecast. The tracking signal reports significant bias in the forecast.

The head of forecasting should take these exception reports seriously. First find the assignable cause for the exception, and then take appropriate action. Do not wait until the exception is reported to start thinking about the assignable cause. Be aware of events in the operating environment that could cause exceptions and use the reports to confirm or deny hypotheses about the problems actually occurring. Look for patterns, where several series seem to show the same anomalies.

Several techniques have been proposed for the tracking signal. Brown (1959) originally used the cumulative sum of forecast errors. However, the expected value of that sum in the future is the current sum. A large error can bias the signal so that a very small error later could cause an exception.

The next technique was the smoothed error tracking signal (SETS) which applied exponential smoothing to the error with sign. This technique is slow to react to a real change in the underlying process which generates the data. Trigg (1966) proposed using the SETS to modify the smoothing constant(s) — make the forecasts more responsive when they are wrong and more stable when they are close to the data. They failed to show that the feedback system is critically damped in all regions where it may be applied.

Gardner (1985) has done a study of the comparative effectiveness of a variety of tracking signal techniques. Barnard (1959) proposed the V-mask based on Wald's sequential analysis for quality control (Wald 1947). Brown (1971) used a parabolic mask as the envelope of these V-masks with a range of likelihood ratios. The technique has been extended to monitor the MSE as well as the forecast, based on analogy with Shewhart's (1931) X-bar and R charts for statistical quality control.

During the course of fitting the initial model outliers are the analogs of demand filter exceptions. Brown (1990) used the term "significant event" to refer to history where there is evidence that not all the history came from a process that can be described by the same model. Significant serial correlation with a lag of one observation may be caused by such a significant event (there are other causes). Brown (1990) has evolved a method with cumulative sums for estimating the time when that event occurred, so that the model can be fitted only to observations since that time.

The whole idea of forecasting from a description of history is on the way out. It will be common to pass point-of-sale data quickly and accurately to each operation in the logistics chain, rather than to forecast what one enterprise will order from another enterprise.

See

► Forecasting
► Marketing

- ▶ Quality Control
- ▶ R Chart
- ▶ Regression Analysis
- ▶ Retailing
- ▶ Time Series Analysis
- ▶ $\bar{X}$ Chart

References

Barnard, G. (1959). Control charts and stochastic processes. *Journal of the Royal Statistical Society Series B, 21*, 239–271.

Box, G., & Jenkins, G. (1970). *Time series analysis*. San Francisco: Holden-Day.

Brown, R. G. (1959). *Statistical forecasting for inventory control*. New York: McGraw Hill.

Brown, R. G. (1963). *Smoothing, forecasting and prediction*. Englewood Cliffs: Prentice Hall.

Brown, R. G. (1967). *Decision rules for inventory management*. New York: Holt, Rinehart & Winston.

Brown, R. G. (1971). Detection of turning points. *Decision Science, 2*, 383–403.

Brown, R. G. (1990). Significant events. In *Proceedings of ISF*, New York.

Gardner, E. (1985). Exponential smoothing: The state of the art. *Journal of Forecasting, 4*, 1–28.

Hyndman, R., Koehler, A. B., Ord, J. K., & Snyder, R. D. (2008). *Forecasting with exponential smoothing*. New York: Springer.

Shewhart, W. A. (1931). *Economic control of quality*. New York: Van Nostrand.

Trigg, D. W. (1966). Monitoring a forecasting system. *Journal of Operational Research Society, 15*, 211–274.

Wald, A. (1947). *Sequential analysis*. New York: Wiley.

Weiner, N. (1949). *Extrapolation, interpolation and smoothing of stationary time series*. New York: Wiley.

Winters, P. R. (1962). Constrained rules for production smoothing. *Management Science, 8*, 470–481.

Exponential Tilting/Twisting

In stochastic or Monte Carlo simulation, an exponentially weighted change of measure for importance sampling in estimating rare events.

See

- ▶ Rare Event Simulation
- ▶ Variance Reduction Techniques in Monte Carlo Methods

Exponential-Bounded (–Time) Algorithm

An algorithm for which it can be shown that the number of steps required to find a solution to a problem is an exponential function of the problem's data. The simplex algorithm is an exponential-bounded algorithm, although its use in practice belies that designation.

See

- ▶ Polynomially Bounded (–Time) Algorithm (Polynomial Algorithm)
- ▶ Simplex Method (Algorithm)

Extremal

Maximum or minimum.

Extremal Column

In the Dantzig-Wolfe Decomposition Algorithm, the extremal columns are the columns of the extremal (master) problem.

See

- ▶ Dantzig-Wolfe Decomposition Algorithm

Extremal Problem

In the Dantzig-Wolfe Decomposition Algorithm, the extremal column is the original linear-programming problem expressed in terms of its extreme point solutions.

See

- ▶ Dantzig-Wolfe Decomposition Algorithm

Extreme Direction

A point in a convex set that cannot be expressed as a convex combination of two other directions of the set.

Extreme Point

A point in a convex set that cannot be expressed as a convex combination of two other distinct points in the set. Extreme points are also known as corner points or vertices. The extreme points of a rectangle are its four vertices, while the extreme points of a circular disc are the points on its circumference. For a linear programming problem, the extreme points of its convex set of solutions correspond to basic feasible solutions, and it can be shown that, if the problem has a finite optimal solution, then one of the extreme points is optimal.

Extreme Point Solution

A solution to a linear-programming problem that is an extreme-point of its convex set of solutions. Such solutions correspond to basic feasible solutions.

Extreme Ray

A ray in a convex set whose direction is an extreme solution.

F

Face Validity

► Verification, Validation, and Testing of Models

Facilities Layout

Bharat K. Kaku
Georgetown University, Washington, DC, USA

Introduction

In both manufacturing and service operations, the relative location of facilities is a critical decision affecting costs and efficiency of operations. The facility layout problem (FLP) deals with the design of layouts wherein a given number of discrete entities are to be located in a given space. The definitions of entities and spaces can vary considerably, making solution techniques applicable in a wide variety of settings, as can be seen from the examples given below.

Entities	Space
Departments	Office building
Departments	Factory floor
Departments	Hospital
Interdependent plants	Geographical market
Indicators and controls	Control panel
Components	Electronic boards
Keys	Typewriter keyboard

In what follows, approaches used to model the FLP are discussed first, followed by optimal algorithms and heuristic approaches for solving these problems, and ending with some remarks concerning directions for future research.

The Quadratic Assignment Formulation

The FLP is most often treated in the OR/MS literature as the Quadratic Assignment Problem (QAP), which is a special case requiring identical area and shape requirements for the locations of all facilities. This allows pre-definition of the locations and calculation of the distances between them (typically center-to-center, either rectilinear or Euclidean). Suppose there are N facilities to be assigned to N locations. Define four $N \times N$ matrices whose elements are, respectively:

$c_{ij} =$	fixed cost of assigning facility i to location j
$F_{ij} =$	level of interaction between facilities i and j
$d_{ij} =$	cost of one unit of interaction (e.g., the distance) between locations i and j
$x_{ij} =$	1 if facility i is assigned to location j, and 0 otherwise.

Then the QAP is to

$$\text{Minimize} \quad \sum_{i,j} c_{ij}x_{ij} + \sum_{i,p}\sum_{j,q} f_{ip}\, d_{jq}x_{ij}x_{pq} \qquad (1)$$

$$\text{subject to} \quad \sum_{j} x_{ij} = 1 \quad \forall\, i \qquad (2)$$

$$\sum_{i} x_{ij} = 1 \quad \forall\, j$$
$$x_{ij} \in \{0,\ 1\} \quad \forall\, i,j \qquad (3)$$

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Alternatively, $\rho(i)$ can be defined as the location to which facility i is assigned, leading to an equivalent but more compact statement of the problem. The QAP is then to find a mapping of the set of facilities into the set of locations so as to

$$\text{Minimize} \quad \sum_i c_{i,\rho(i)} + \sum_{i,p} d_{\rho(i),\rho(p)}. \qquad (4)$$

Objective
Minimize cost of interactions
Minimize cost of material handling
Minimize movement of patients and medical staff
Maximize profit
Minimize eye/hand movement
Minimize cost of connections
Minimize typing time

The quadratic assignment problem was first formulated by Koopmans and Beckmann (1957) in the context of the location of interdependent plants. The c_{ij} elements represent the expected revenue of operating plant i at location j independent of other plant locations, the f_{ij} elements represent the required commodity flows from plant i to plant j, and the d_{ij} elements represent the transportation costs per unit between location i and location j. The objective function maximizes the net revenue, that is, the excess of expected revenue over the transportation costs.

It is the interdependence of facilities due to interactions between them that leads to the quadratic term in the objective function and makes the problem a difficult one. If the departments are independent of each other (i.e., all $f_{ij} = 0$), the QAP reduces to the familiar linear assignment problem, for which efficient solution techniques exist. Further, the traveling salesman problem is a special case of the QAP. To see this, consider the interaction matrix to be a cyclic permutation matrix with the following interpretation: A flow of one unit (the salesman) travels from the first city in the tour to the second city in the tour to the third city, and so on, finally returning to the first city in the tour. The distance matrix is simply the matrix of distances between cities, and the fixed costs are zero. A solution to this QAP can be interpreted as follows. If $x_{ij} = 1$, then city i is in the jth location in the tour. This shows that the QAP belongs to the class of NP-hard problems.

The Adjacency Requirements Formulation

This approach is based on adjacency requirements and closeness ratings. The former stipulate the set of pairs of facilities that must be adjacent, or must not be adjacent, in any feasible solution, whereas the latter are measures of the desirability of locating a pair of facilities in adjacent locations, generally based on the interaction between them. The adjacency requirements must permit at least one feasible solution. If there is more than one feasible solution, then the closeness ratings are used to choose the optimal solution. In evaluating a solution, the closeness ratings are added only for facility pairs that are adjacent.

The QAP has received most of the attention in the literature for the following two reasons. First, it considers interaction costs for all pairs of facilities, whereas the adjacency requirements formulation maximizes the sum of closeness ratings for adjacent facilities only, while satisfying the adjacency requirements. Second, the adjacency requirements formulation does not consider fixed costs which can be important, especially when a layout is being redesigned, which is more common than design of a brand-new facility. In the rest of this article, the discussion is restricted the QAP. Further information on the adjacency requirements approach can be found in Foulds (1983) and a more extensive list of references for the QAP can be found in the review by Kusiak and Heragu (1987). The book by Heragu (2008) offers a comprehensive look at applications of the QAP model in different settings.

Optimal Algorithms for the QAP

Algorithms for obtaining exact solutions to the QAP can be classified under the categories of linearization and implicit enumeration.

Linearization — Several linearizations have been proposed for the QAP. The first one was by Lawler (1963) who linearized the problem by defining variables $y_{ijpq} = x_{ij}\, x_{pq}$. Compared to the original QAP, the resulting integer programming problem has N^4 additional binary variables and $N^4 + 1$ additional constraints. A linearization proposed by Kaufman and Broeckx (1978) is the most compact,

adding only N^2 new continuous variables and N^2 new constraints. Bazaraa and Sherali (1980) suggested another linearization to which they applied Benders decomposition. None of these approaches has proved to be computationally effective. More details can be found in the survey paper by Kusiak and Heragu (1987).

Implicit Enumeration — Branch-and-bound algorithms have been the most successful in solving the QAP to optimality; problems with as many as 15 or 16 facilities can be solved in reasonable time. Some earlier implicit enumeration methods were pair-assignment algorithms where a node in the branch-and-bound tree corresponds to the assignment of a pair of facilities to a pair of locations (Land 1963; Gavett and Plyter 1966). These did not prove to be competitive with single-assignment algorithms which assign one facility to one location at each node.

Gilmore (1962) and Lawler (1963) independently developed a lower bound for use in a single-assignment branch-and-bound procedure. This lower bound forms the basis for the most successful implicit enumeration algorithms published (Bazaraa and Kirca 1983; Burkard and Derigs 1980). How the Gilmore-Lawler lower bound is calculated is described next.

Suppose $\mathcal{F}$ is the set of facilities (possibly empty) that have already been assigned, and $\mathcal{L}$ is the set of locations to which these facilities have been assigned. Using the alternate formulation (4), a lower bound on completions of this partial assignment is given by

$$\begin{aligned}\text{Min}\quad & \sum_{i\in\mathcal{F}} c_{i,\rho(i)} + \sum_{i\in\mathcal{F}}\sum_{p\in\mathcal{F}} f_{ip}\, d_{\rho(i),\rho(p)} \\ & + \sum_{i\in\mathcal{F}}\sum_{p\notin\mathcal{F}} [f_{ip}\, d_{\rho(i),\rho(p)} + f_{pi} d_{\rho(p),\rho(i)}] \\ & + \sum_{i\notin\mathcal{F}} c_{i,\rho(i)} + \sum_{i\notin\mathcal{F}}\sum_{p\notin\mathcal{F}} f_{ip}\, d_{\rho(i),\rho(p)}. \end{aligned} \tag{5}$$

The first two terms in the expression (5) are the known fixed and interaction costs of assignments already made; the third term captures the interaction costs between assigned facilities and those yet to be assigned; and the last two terms represent the fixed and interaction costs of assignments not yet made. A minimum can be calculated for the last three terms as follows. Consider the assignment of any unassigned facility $i \notin \mathcal{F}$ to a free location $j \notin \mathcal{L}$. The incremental cost due to this assignment is

$$\sum_{p\in\mathcal{F}} [f_{ip}\, d_{j,\rho(p)} + f_{pi} d_{\rho(p),j}] + c_{ij} + \sum_{p\notin\mathcal{F}} f_{ip} d_{j,\rho(p)}. \tag{6}$$

Now the first two terms of (6) are known, and the third term needs to be minimized. Form a vector of flows consisting of the ith row of the flow matrix minus the diagonal element minus the elements corresponding to assigned facilities ($i \in \mathcal{F}$). Arrange the elements of this vector in decreasing order. Form a similar vector of distances consisting of the jth row of the distance matrix minus the diagonal element minus the elements corresponding to filled locations ($j \in \mathcal{L}$), and arrange it in increasing order. The scalar product of these vectors provides the necessary minimum cost. Essentially, the largest interaction with i incurs the lowest per unit cost, the second largest interaction incurs the second lowest cost, and so on. Repeat for all pairs (i, j) such that $i \notin \mathcal{F}$ and $j \notin \mathcal{L}$. A solution to the linear assignment problem (LAP) with these incremental costs as cost coefficients provides a lower bound on the three unknown terms of (5).

Let the value of this solution be z^*. The Gilmore-Lawler lower bound is then obtained as

$$LB = \sum_{i\in\mathcal{F}} c_{i,\rho(i)} + \sum_{i\in\mathcal{F}}\sum_{p\in\mathcal{F}} f_{ip} d_{\rho(i),\rho(p)} + z^*. \tag{7}$$

Any node at which the lower bound is greater-than-or-equal-to the upper bound can be fathomed in the usual way. However, an attractive feature of this lower bound is the fact that additional information is available to be used in the search process. Consider the solution to the LAP solved to obtain the lower bound, with facility $i \notin \mathcal{F}$ assigned to location $\rho(i)$. The dual variables of the optimal solution can be used to reduce the cost matrix so that every $(i, \rho(i))$ element is zero. Adding together the next smallest element in that row and in that column gives the regret or minimum additional cost if assignment $(i, \rho(i))$ is not made. The lower bound plus regret gives the alternate cost of this assignment. Using this cost as a branching rule, the next assignment is chosen with the maximum alternate cost. Further, while backtracking, if the

alternate cost at a node is greater than the upper bound, no more nodes need to be evaluated at that level on the present branch.

Heuristic Solution Methods for the QAP

Given the limited size of problems that can be solved to optimality (smaller than most practical problems), there has been considerable interest in developing heuristic procedures for the QAP. Heuristics for the QAP can be classified as limited enumeration, construction methods, improvement methods, and hybrid methods.

Limited Enumeration — It has often been observed that an optimal solution is found fairly early in a branch-and-bound procedure, with the majority of the solution time then being spent in proving optimality. A heuristic based on limited enumeration takes advantage of this feature by setting a cut-off time to truncate the search process. The search can either be shortened or allowed to cover more of the search space in a fixed amount of time by fathoming a node at which the gap between the lower and upper bounds is sufficiently small. This gap can be set based on empirical evidence about the behavior of bounds, for example, in the QAP the lower bound rises rapidly at higher levels of the branch-and-bound tree and then more gradually. Thus a dynamic gap could be used, larger at higher levels and decreasing at lower levels of the tree.

Construction Methods — A constructive procedure starts with an empty assignment and adds assignments one at a time until a complete solution is obtained. The rule used to choose the next assignment can be a simple one such as assign the facility with the maximum interaction with a facility already assigned and place it as close as possible to that facility. Alternatively, a rule may be employed that takes into account assignments already made, as well as future assignments to be made, which is likely to lead to better solutions. Examples of the latter type of rule would be the use of alternate costs obtained in the process of calculating lower bounds (see above) or the use of an evaluation function such as that devised by Graves and Whinston (1970). The Graves-Whinston method uses statistical properties to compute an expected value for the completion of any partial assignment using only basic arithmetic operations. The computation time is very reasonable, thus making it a good choice as a constructive heuristic. In addition to the notation defined earlier, define $\mathcal{L}$ as the set of locations that have been assigned facilities. Suppose also that k assignments have already been made. The expected value of a complete assignment is given by expression (8) whose terms are analogous to those in (5):

$$EV = \sum_{i\in\mathcal{F}} c_{i,\rho(i)} + \sum_{i\in\mathcal{F}}\sum_{p\in\mathcal{F}} f_{ip} d_{\rho(i),\rho(p)} + \frac{\sum_{i\in\mathcal{F}}\sum_{p\notin\mathcal{F}}\sum_{j\notin\mathcal{L}} [f_{ip} d_{\rho(i),j} + f_{pi} d_{j,\rho(i)}]}{n-k} + \frac{\sum_{i\notin\mathcal{F}}\sum_{j\notin\mathcal{L}} c_{ij}}{n-k} + \frac{\sum_{i,p\notin\mathcal{F}} f_{ip}\left(\sum_{j,q\notin\mathcal{L}} d_{jq}\right)}{(n-k)(n-k-1)}. \tag{8}$$

Improvement Methods — Improvement procedures start with some sub-optimal solution and attempt to improve it through partial changes in the assignments. The design of an improvement routine requires decisions concerning the following: type of exchange — pairwise, triple, or some higher order; number of exchanges to consider — should all possible exchanges be considered or a limited set; choice of exchange actually made — first improvement or best improvement; order of evaluation — random or predetermined. An effective strategy is to use pair-wise exchanges in fixed order of decreasing interactions, considering all possible exchanges, and accepting the first improvement. Higher order exchanges are best used sparingly. Other improvement techniques, such as simulated annealing (Connolly 1990) and tabu search (Skorin-Kapov 1990), that avoid the trap of local optima have been applied with success to the QAP.

Hybrid Methods — Some of the most successful heuristic solution methods for the QAP can be termed hybrid methods because they combine the power of improvement methods with some method for obtaining solutions to be improved, for example, construction methods (Ligett 1981), limited enumeration (Bazaraa and Kirca 1983), or cutting planes (Burkard and Bonniger 1983). Kaku et al. (1991) successfully combined constructed solutions with exchange improvement by systematically

constructing solutions that were different from each other. This forces different areas of the search space to be examined.

Concluding Remarks

The QAP formulation suffers from two drawbacks. First, it assumes identical area and shape requirements for all facilities. Unequal areas could be dealt with by dividing all facilities into equal-area modules which could be kept together in a solution by introducing very high artificial flows between them. However, this increases the size of the problem. Improvement methods can employ such a strategy since the number of facilities is less of a concern, however, exchanges are then limited to either equal-sized facilities or to adjacent facilities. Work by Bozer et al. (1994) incorporating the use of space-filling curves in facility layout overcomes this handicap for improvement methods. Second, the QAP deals exclusively with interaction costs, generally material handling costs. This is not likely to be the only concern in facility layout. For this reason, the general practice is to allow a human designer to evaluate and fine tune a solution before implementation. For example, Fu and Kaku (1997) examined the effect of layout design on work-in-process (WIP) levels in a factory, an issue of great interest when considering lean manufacturing. They found that good QAP solutions generally reduce the levels of WIP, but there are exceptions which the QAP approach cannot discern. The following are some of the features that are desirable in a heuristic solution procedure for the FLP: The ability to handle different area requirements; the ability to produce good solutions with reasonable computational requirements; and the ability to either consider multiple criteria or present the decision maker with good layout alternatives to choose from.

See

- ▶ Branch and Bound
- ▶ Facility Location
- ▶ Heuristics
- ▶ Location Analysis
- ▶ Quadratic Assignment Problem
- ▶ Tabu Search

References

Bazaraa, M. S., & Kirca, O. (1983). A branch-and-bound-based heuristic for solving the quadratic assignment problem. *Naval Research Logistics Quarterly, 30*, 287–304.

Bazaraa, M. S., & Sherali, H. D. (1980). Bender's partitioning scheme applied to a new formulation of the quadratic assignment problem. *Naval Research Logistics Quarterly, 27*, 29–41.

Bozer, Y. A., Meller, R. D., & Erlebacher, S. J. (1994). An improvement-type layout algorithm for single and multiple-floor facilities. *Management Science, 40*, 918–932.

Burkard, R. E., & Bonniger, T. (1983). A heuristic for quadratic boolean programs with applications to quadratic assignment problems. *European Journal of Operational Research, 13*, 374–386.

Burkard, R. E., & Derigs, U. (1980). *Assignment and matching problems: Solution methods with fortran programs. Vol. 184 of lecture notes in economics and mathematical systems*. Berlin: Springer.

Connolly, D. T. (1990). An improved annealing scheme for the QAP. *European Journal of Operational Research, 46*, 93–100.

Foulds, L. R. (1983). Techniques for facilities layout: Deciding which pairs of activities should be adjacent. *Management Science, 29*, 1414–1426.

Fu, M., & Kaku, B. K. (1997). Minimizing work-in-process and material handling in the facilities layout problem. *IIE Transactions, 29*, 29–36.

Gavett, J. W., & Plyter, N. V. (1966). The optimal assignment of facilities to locations by branch and bound. *Operations Research, 14*, 210–232.

Gilmore, P. C. (1962). Optimal and suboptimal algorithms for the quadratic assignment problem. *Journal of the Society for Industrial and Applied Mathematics, 10*, 305–313.

Graves, G. W., & Whinston, A. B. (1970). An algorithm for the quadratic assignment problem. *Management Science, 17*, 453–471.

Heragu, S. S. (2008). Facilities Design, 3rd ed. *CRC Press*.

Kaku, B. K., Thompson, G. L., & Morton, T. E. (1991). A hybrid heuristic for the facilities layout problem. *Computers and Operations Research, 18*, 241–253.

Kaufman, L., & Broeckx, F. (1978). An algorithm for the quadratic assignment problems using Bender's decomposition. *European Journal of Operational Research, 2*, 204–211.

Koopmans, T. C., & Beckmann, M. (1957). Assignment problems and the location of economic activities. *Econometrica, 25*, 53–76.

Kusiak, A., & Heragu, S. S. (1987). The facility lay-out problem. *European Journal of Operational Research, 29*, 229–251.

Land, A. H. (1963). A problem of assignment with inter-related costs. *Operational Research Quarterly, 14*, 185–199.

Lawler, E. L. (1963). The quadratic assignment problem. *Management Science, 9*, 586–599.

Ligett, R. S. (1981). The quadratic assignment problem: An experimental evaluation solution strategies. *Management Science, 27*, 442–458.

Skorin-Kapov, J. (1990). Tabu search applied to the quadratic assignment problem. *ORSA Journal on Computing, 2*, 33–45.

Facility Location

Dilip Chhajed[1], Richard L. Francis[2] and Timothy J. Lowe[3]
[1]University of Illinois at Urbana-Champaign, Champaign, IL, USA
[2]University of Florida, Gainesville, FL, USA
[3]University of Iowa, Iowa City, IA, USA

Introduction

Location problems that can be quantified as optimization problems are natural candidates for operations research approaches, and many such problems have been studied using mathematical-programming methodology. This article gives an overview of some of this activity. Models of these location problems are classified as planar, network, and mixed integer-programming models, and methodology for solving such types of models is outlined. There is little doubt that the contributions to facility location consist principally of algorithms — well-defined computational procedures for solving quantifiable problems.

A location problem must be quantifiable for there to be any hope of solving it with an algorithm: there must be a well-defined objective to be optimized, for example, cost to be minimized, or profit to be maximized. Likewise there are usually well-defined constraints, for example, budget constraints, which limit the scope of the optimization. Location problems which are highly subjective or political in nature are thus usually not very good candidates for operations research approaches, although even for such problems there may be results which can help to reduce the scope of the problem under consideration, or identify basic tradeoffs of interest.

In what follows, some basic location models are considered. For further reading, see the texts by Handler and Mirchandani (1979); Love et al. (1988) and Francis et al. (1992).

Selected Models

This section describes some of the basic but popular and useful models of location theory, along with brief discussions of their solution approaches. Note that demand points are referred to as existing facilities and the facilities to be located as new facilities. In developing the models, the following notation is used:

p: number of new facilities. The value of p may be a decision variable or may be fixed;
m: number of existing facilities;
w_i : weight associated with existing facility i;
X: location of a single new facility;
$\boldsymbol{X} = (X_1, \ldots, X_p)$: locations of p new facilities; and
$D_i(X)$: the distance between existing facility i and the *nearest* new facility.

Model 1, P-Center Problem

The objective in this model is to locate p new facilities to minimize the maximum distance to an existing facility. Let $g(\boldsymbol{X}) = \max_{i = 1,\ldots,m} \{w_i D_i(\boldsymbol{X})\}$ represent the maximum (weighted) distance any person has to travel; then the problem can be posed as

$$\underset{x}{\text{Minimize}} \; g(X).$$

This problem is known as the p-center problem and, besides other applications, has been used to model locations of emergency medical facilities such as the location of a helicopter to minimize the maximum time to respond to an emergency, and the location of a transmitter to maximize the lowest signal level received.

Model 2, Covering Problem

In this problem, the number of facilities to be located is not fixed a priori. Each existing facility should be within a specified weighted distance from at least one new facility. The objective is to find the number of new facilities, p, and their locations, $\boldsymbol{X}$, to minimize the cost of the new facilities.

The version of the covering problem with a finite set of candidate facility locations can be modeled as a set-covering problem. With S as the set of n candidate sites for facilities, denote the sites by $j = 1, \ldots, n$. Define variables y_j which take on a value 1 if a new facility is opened at site j and 0 otherwise. Let f_j represent the cost of locating (opening) a facility at j. Customers (existing facilities) are indexed

by $i =; 1,\ldots, m$. Let $a_{ij} = 1$ if a new facility located at site j can cover existing facility i, 0 otherwise, $i = 1,\ldots, m$ and $j = 1,\ldots, n$. Note that the a_{ij} values are constants and are determined prior to the formulation. An integer-programming formulation of the set-covering problem can be written as:

$$\min \sum_{j \in S} f_i y_i$$

$$\text{subject to :} \quad \sum_{j \in S} a_{ij} y_i \geq 1, \qquad \forall i = 1, ..., m$$

$$yi \in \{0, 1\} \qquad \forall i = 1, ..., n.$$

The objective function sums up the fixed costs of locating the new facilities. When each $f_j = 1$, the objective function minimizes the number of new facilities to be located. The first constraint ensures that each existing facility is covered while the second constraint restricts the variables to be binary.

Model 3, Simple Plant Location Problem

In the simple plant location problem (SPLP), a number of new facilities need to be opened (the actual number is a decision variable) to serve a given set of customers. There is a fixed cost of opening each facility. The objective is to minimize the sum of the fixed and variable costs of serving the demand points, and to determine the optimal allocation pattern for all customers.

The SPLP can be formulated as a mixed-integer program as follows. In addition to S, f_j and y_j defined earlier, let $c_{ij} = 0$, $i \not\in 1,\ldots, m$ and $j = 1,\ldots, n$, be the unit cost of servicing customer i from a new facility located at j. Letting x_{ij} denote the fraction of customer i's service provided by site j, a formulation of the SPLP is:

$$\min \quad \sum_{j \in S} f_i y_i + \sum_{i=1}^{m} \sum_{j \in S} c_{ij} x_{ij} \qquad (1)$$

$$\text{subject to :} \quad \sum_{j \in S} x_{ij} = 1, \quad i = 1, ..., m \qquad (2)$$

$$y_j \in \{0, 1\}, \quad j = 1, ..., n \qquad (3)$$

$$x_{ij} \geq 0, \quad i = 1, ..., m, \quad j = 1, ..., n \qquad (4)$$

and

$$x_{ij} \leq y_j, \quad i = 1, ..., m, \quad j = 1, ..., n \qquad (5)$$

Expression (1) totals site costs and service costs. The requirement that each customer be completely served is assured by (2). Expression (3) prevents a fractional opening of a site, and (4) assures nonnegative service. The condition that service cannot be provided from an unopened facility is guaranteed by (5).

If there are no facility related cost terms in the objective function (i.e., $f_j = 0$ for all j) and the number of facilities is restricted to exactly p, then the resulting problem is known as the p-median problem.

Solving the Models

In order to describe selected contributions that operations research has made towards providing solution procedures for the above problems, the problem space is divided into three classes: planar models, network models, and discrete models. Many of the above models can be posed on any one of the three spaces with some slight modifications, and almost all models can be put into one of the three classes.

The main difference in these three classes of models is the manner in which the distance between two points is defined. In planar models, the distance function $d(\cdot)$ is a norm, often Euclidean, rectilinear, or some other norm, and the number of possible locations for new facilities is infinite. This renders the corresponding problems as continuous. If (a_i, b_i) are the coordinates of a point i, then the Euclidean distance between points i and j is given by $\sqrt{(a_i - a_j)^2 + (b_i - b_j)^2}$, while the rectilinear distance is given by $|a_i + a_j| - |b_i - b_j|$, where $|\cdot|$ is the absolute value function.

Many of the best known OR algorithms for solving location problems involve choosing best locations from a finite collection of possible sites. Since a site either is chosen or is not, such problems are intrinsically discrete in nature and are candidates for being solved as integer-programming problems. For example, a banking corporation might be uncertain as to how many branch banks there should be. The corporation would realize that the more banks it

locates, the more convenient the branches would be to its customers in terms of travel time or travel cost. On the other hand, the more branches there are, the higher would be the operating expenses and fixed costs. Thus there is a trade-off between convenience and operating costs, which it would be important to analyze. Such trade-offs often occur in solving location problems.

In discrete models, the number of existing facilities and the number of potential sites for new facilities is finite. Distances may be derived from planar or network distances, or some more general type of transport cost which is proportional to distance. Discrete problems, modeled as mixed integer programs, are often more difficult to solve. On the other hand, many realistic assumptions can be incorporated in discrete models which cannot be included in planar or network models.

When a location problem has substantial transport costs, and the fixed site costs are relatively independent of location, there are several other approaches to modeling it. Often it is assumed that transport costs are directly proportional to transport distances. When these distances are incurred on a transport network, such as a road network, the result is often a network model. Such models usually employ shortest path algorithms to compute travel distances. The focus of network model research has been principally upon two topics: 1) algorithms to solve the problems, and 2) localization results, such as vertex-optimality results, which reduce to a finite collection the set of locations which must be considered to obtain an optimal solution (Hooker et al. 1991) Once such a finite set is obtained, the resulting remaining problem may well be modeled as an integer- or mixed- integer programming problem.

Distances are often more accurately represented in network models than in planar models, but the need for data is also higher in network models since the length of each segment is needed. For many models it becomes advantageous to work directly with the network, exploiting its properties in developing a solution procedure. The existing facilities are located on the network, and the new facilities are to be located at points on the network. An additional advantage of network models is that they make problem visualization easier. Thus, even if the problem is not solved as a network problem, a solution presented in network form may assist the decision maker in understanding the problem and the issues involved.

It is possible, for a network location problem, that it may be prohibitively expensive to obtain, or to work with, the necessary network data. In such cases, network distances may well be approximated using planar distances, for example, Euclidean or rectilinear distances. This results in what is called a planar model. The related problems are often easier to analyze, and can be helpful for providing insight. Often results from nonlinear programming can be employed to help solve such problems.

Real-world location problems which occur in urban contexts may well have millions of demand points when each private residence is a demand point. It is common to use some means of aggregating demand points. The aggregation reduces the problem size, but introduces error into the model. There is evidence that aggregation errors decrease with the number of aggregate demand points at a decreasing rate. For a small number of aggregate demand points, the error may be quite high; for a larger number it may be quite low, and increasing the number even more will give little additional decrease in the error, see Francis et al. (1999).

As concerns algorithms/solution approaches to the three classes of problems, planar problems are usually solved using linear or nonlinear programming methods; network problems are usually solved using network and graph-theoretic methods; discrete problems are usually solved using integer-programming methods. More information on these methods can be found in the references. In addition, many artificial intelligence based heuristic approaches, such as genetic algorithms, tabu search, and simulated annealing, can be used to solve location problems.

See

- ▶ Artificial Intelligence
- ▶ Facilities Layout
- ▶ Genetic Algorithms
- ▶ Integer and Combinatorial Optimization
- ▶ Location Analysis
- ▶ Network
- ▶ Shortest-route Problem
- ▶ Simulated Annealing
- ▶ Stochastic Programming
- ▶ Tabu Search

References

Drezner, Z. (Ed.). (1995). *Facility location: A survey of applications and methods*. New York: Springer.

Francis, R. L., Lowe, T. J., Rushton, G., & Rayco, B. (1999). A synthesis of aggregation methods for multifacility location problems: Strategies for containing error. *Geographical Analysis, 31*, 67–87.

Francis, R., McGinnis, L. F., & White, J. A. (1992). *Facility layout and location: An analytical approach*. Englewood Cliffs, NJ: Prentice Hall.

Handler, G. Y., & Mirchandani, P. B. (1979). *Location on networks: Theory and algorithms*. Cambridge, MA: MIT Press.

Hooker, J. N., Garfinkel, R. S., & Chen, C. K. (1991). Finite dominating sets for network location problems. *Operations Research, 39*, 100–118.

Love, R. F., Morris, J. G., & Wesolowsky, G. O. (1988). *Facilities location: Models and methods*. Amsterdam: North-Holland.

Factorable Programming

Richard H. F. Jackson
National Institute of Standards and Technology, Gaithersburg, MA, USA

Factorable programming problems are mathematical programming problems of the form

$$\underset{x \in R^n}{\text{minimize}}\, f(x),$$
$$\text{subject to } g_i(x) > 0,$$

for $i = 1,\ldots, m$, in which all the functions involved are factorable. Loosely, a factorable function is a multivariate function that can be written as the last of a finite sequence of functions, in which the first n functions in the sequence are just the coordinate variables, and each function beyond the nth is a sum, a product, or a single-variable transformation of previous functions in the sequence. More rigorously, let $[f_1(x), f_2(x),\ldots, f_L(x)]$ be a finite sequence of functions such that $f_i : R^n \to R$ where each $f_i(x)$ is defined according to one of the following rules:

Rule 1: For $i = 1,\ldots, n, f_i(x)$ is defined to be the ith Euclidean coordinate, or $f_i(x) = x_i$.

Rule 2: For $i = n + 1,\ldots, L, f_1(x)$ is formed using *one* of the following compositions:

a. $f_i(x) = f_{j(i)}(x) + f_{k(i)}(x)$; or
b. $f_i(x) = f_{j(i)}(x) \cdot f_{k(i)}(x)$; or
c. $f_i(x) = T_i[f_{j(i)}(x)]$;

where $j(i) < i$, $k(i) < i$, and T_i is a function of a single variable. Then $f(x) = f_L(x)$ is a factorable function and $[f_1(x), f_2(x),\ldots, f_L(x)]$ is a factored sequence. Thus a function, $f(x)$, will be called factorable if it can be formed according to Rules 1 and 2, and the resulting sequence of functions will be called a factored sequence, or at times the function written in factored form.

Although it is not always immediately grasped, the concept of a factorable function is actually a very natural one. In fact it is just a formalization of the natural procedure one follows in evaluating a complicated function. Consider for example the function

$$f(\boldsymbol{x}) = [\boldsymbol{a}^T\boldsymbol{x}]\ \sin[\boldsymbol{b}^T\boldsymbol{x}]\ \exp[\boldsymbol{c}^T\boldsymbol{x}],$$

where $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, and $\boldsymbol{x}$ are (2×1) vectors. The natural approach to evaluating this function for specified values $\boldsymbol{x}_1^0$ and $\boldsymbol{x}_2^0$ is first to compute the quantities within the brackets then to apply the sine and exponential functions, and finally to multiply the three resulting quantities. This might be done in stages as follows:

$$\begin{aligned}
f_1 &= \boldsymbol{x}_1^0 & f_9 &= \boldsymbol{c}_1 f_1\\
f_2 &= \boldsymbol{x}_2^0 & f_{10} &= \boldsymbol{c}_2 f_2\\
f_3 &= \boldsymbol{a}_1 f_1 & f_{11} &= f_9 + f_{10}\\
f_4 &= \boldsymbol{a}_2 f_2 & f_{12} &= \sin(f_8)\\
f_5 &= f_3 + f_4 & f_{13} &= \exp(f_{11})\\
f_6 &= \boldsymbol{b}_1 f_1 & f_{14} &= f_5 \cdot f_{12}\\
f_7 &= \boldsymbol{b}_2 f_2 & f_{15} &= f_{13} \cdot f_{14}\\
f_8 &= f_6 + f_7
\end{aligned}$$

This is one possible factored sequence for $f(\boldsymbol{x})$.

To understand what follows, the concept of an outer product matrix must be introduced. An $(m \times n)$ matrix $\boldsymbol{A}$ is called an outer product matrix if there exists a scalar α, an $(m \times 1)$ vector $\boldsymbol{a}$, and an $(n \times 1)$ vector $\boldsymbol{b}$ such that

$$\boldsymbol{A} = \boldsymbol{a}\,\alpha\boldsymbol{b}^T.$$

The expression $\boldsymbol{a}\,\alpha\boldsymbol{b}^T$ is called an outer product or a dyad. Note that a dyad is conformable since the

dimensions of the product are $(m \times 1)(1 \times 1)(1 \times n)$, which yields the $(m \times n)$ outer product matrix $\boldsymbol{A}$ as desired. A useful property of outer product matrices is that, if kept as dyads, matrix multiplication is simplified to inner products alone, saving the computations required to form the matrices involved. For example,

$$\begin{aligned} \boldsymbol{Ac} &= \boldsymbol{a}\ \alpha[\boldsymbol{b}^T\boldsymbol{c}], \\ \boldsymbol{d}^T\boldsymbol{A} &= [\boldsymbol{d}^T\boldsymbol{a}]\ \alpha\boldsymbol{b}^T, \quad \text{and} \\ \boldsymbol{AF} &= \boldsymbol{a}\ \alpha[\boldsymbol{b}^T F], \end{aligned}$$

where $\boldsymbol{c}$ is $(n \times 1)$, $\boldsymbol{d}$ is $(m \times 1)$ and F is $(n \times m)$.

It is well-known (McCormick 1983) that factorable functions possess two very special properties that can be exploited to produce efficient (fast and accurate) algorithms: i) once written in factorable form, their gradients and Hessians may be computed exactly, automatically, and efficiently; and ii) their Hessians occur naturally as sums of dyads whose vector factors are gradients of terms in the factored sequence. The first of these properties eases the task of providing the derivatives of a nonlinear programming problem to a computer software solution routine, and has the potential eventually to trivialize it. The second, as noted above, changes the way matrix multiplication is interpreted, which in many cases results in less computational effort.

There are factorable problems whose structure is such that the factorable approach results in more work: small, dense problems, for example. For these problems, the factorable approach can still be used for easy input, but some of the matrix techniques would be replaced by classical approaches.

Software packages have been written that perform the factoring automatically from natural language input. See Jackson and McCormick (1988) for a history of such efforts, as well as Jackson, McCormick, and Sofer (1989). The latter paper describes a system that allows user input for nonlinear functions in a format similar to FORTRAN, without any requirement on the user to understand the details of factorable functions.

As mentioned above, one fundamental value of factorable functions lies in the simple and computationally efficient forms that result for their Hessians. In fact, factorable programming is based on the existence of, and the simplified operations that result from, these simple forms. The seminal result is that the Hessian of a factorable function can be written as the sum of dyads, or outer products, of gradients of functions in the factored sequence (Fiacco and McCormick 1968, pp. 184–188). This basic result was generalized in Jackson and McCormick (1986). Before explaining the generalization, it is necessary to generalize the concepts of Hessian and dyad.

Let $\boldsymbol{A} \in R(^{n1\mathrm{x}\ldots\mathrm{x}nN})$, and let $\boldsymbol{A}_{i1,\ldots,i\,N}$ denote the $(i_1,\ldots,i_N)$th element of this array. For the purposes of this article, $\boldsymbol{A}$ is called the *Nth-order tensor* of a multivariable function $f(\boldsymbol{x})$ if

$$\boldsymbol{A}_{i_1,\ldots,i_N} = \partial^N f(\boldsymbol{x})/\partial \boldsymbol{x}_{iN} \ldots \partial \boldsymbol{x}_{i_1}.$$

Note that gradients and Hessians are tensors of order 1 and 2 respectively.

An N-dimensional array $\boldsymbol{A}$ is called a generalized outer product matrix if there exists a scalar α, and an ordered set of vectors $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_N$ (where each a_k is $(n_k \times 1)$) such that each element of $\boldsymbol{A}$ is generated by the product of the scalar α and certain specific elements of the vectors $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_N$ as follows:

$$\boldsymbol{A}_{i_1,\ldots,i_N} = \alpha * \boldsymbol{a}_{1,i_1} * \cdots * \boldsymbol{a}_{N,i_N}$$

for $i_1 = 1,\ldots, n_1;\ldots; i_N = 1,\ldots, n_N$, where $\boldsymbol{a}_{k,i\ k}$ represents the (i_k)th element of the $(n_k \times 1)$ vector $\boldsymbol{a}_k$.

The scalar and set of vectors which generate a generalized outer product matrix taken together are called a polyad and are written

$$(\alpha : \boldsymbol{a}_1 \cdots \boldsymbol{a}_N), \tag{1}$$

where order is important, that is, the vector in position j is associated with the jth dimension. A polyad containing N vector factors is an *N-ad*. Also, an expression containing a sum of polyads is a polyadic, and an expression containing a sum of N-ads is an *N-adic*. (The actual addition here is performed as a sum of the associated generalized outer product matrices.) When vector factors in a polyad are repeated, exponential notation is used, as in the case of the symmetric N-ad, $(\alpha{:}[\boldsymbol{a}]^N)$. Note that the representation of a generalized outer product matrix by a polyad is not unique.

For example, $(\alpha \mid \gamma{:}[\boldsymbol{a}_1\ \gamma]\ldots\boldsymbol{a}_N)$ generates the same N-dimensional array of numbers as does (1) for any nonzero scalar γ. Finally, a 2-*ad* of the form $(\alpha{:}\boldsymbol{ab})$ is equivalent to the more familiar dyad of the form $\boldsymbol{a}\,\alpha\boldsymbol{b}^T$, and the two will be used interchangeably.

The generalization mentioned above is that all tensors (that exist) of factorable functions possess a natural polyadic structure. Furthermore, the vector factors that comprise the monads of the gradient are the same vector factors which comprise the dyads of the Hessian, the triads of the third order tensor, and so on. This has important computational implications in mathematical programming. It means that once the gradient of a factorable function is computed, a major portion of the work involved in computing higher-order derivatives is already calculated. Consequently, high-order minimization techniques, previously considered computationally intractable, are once again worthy of consideration (Jackson and McCormick 1986).

It should be noted that, by their very nature, the tensors of factorable functions are ideally suited for computation on parallel processing and array processing computers. Few other such ideal applications in numerical optimization are known. Also, it has been shown (McCormick 1985) that all factorable programming problems have an equivalent separable programming representation, and that efficient algorithms (Falk and Soland 1969; Falk 1973; Hoffman 1975; McCormick 1976; Leaver 1984) exist for finding global solutions to these problems. Thus there exists the potential of finding global solutions to factorable programming problems fast and accurately.

The discovery and development of factorable functions and their uses in mathematical programming is credited to McCormick (1974). Since the discovery of these functions, the theory of Factorable Programming has been further developed and refined. Ghaemi and McCormick (1979) developed a computer code (FACSUMT), which processes the functions in a factorable program and provides the interface to the SUMT nonlinear programming code (see Mylander et al. 1971). A preliminary version of this code is described in Pugh (1972).

Further extensions of factorable programming theory were provided by Shayan (1978), who developed an automatic method for computing the mth order of a solution technique that can be evaluated when the functions are factorable by counting basic operations and basic functions, a more accurate measure of efficiency than the popular technique of counting the number of equivalent function evaluation (Miele and Gonzalez 1978).

The natural dyadic structure of the Hessian of a factorable function was exploited by Emami (1978) to develop a matrix factorization scheme for obtaining a generalized inverse of the Hessian of a factorable function. Ghotb (1980) also capitalized on this structure and provided formulae for computing a generalized inverse of a reduced Hessian when it is given in dyadic form. Sofer (1983) extended this last concept further by utilizing the dyadic structure to obtain computationally efficient techniques for constructing a generalized inverse of reduced Hessian and updating it from iteration to iteration.

Another direction was pursued by DeSilva and McCormick (1978), who developed the formulae and methodology to utilize the input to general nonlinear programs in factorable form to perform first-order sensitivity analysis on the solution vector. This was generalized in Jacksonand McCormick (1988), where second order sensitivity analysis methods were developed, with formulae involving third order tensors used to compute second derivatives of components of a local solution with respect to problem parameters.

It is important to understand that the derivative calculations performed in factorable programming are not estimations, but mathematically exact calculations. Furthermore they are also compact, since factored sequences mimic hand calculations. Thus, this technique is different from symbolic manipulation techniques for differentiation, which tend to produce large amounts of code. The techniques used in factorable programming are efficient exploitations of the special structure inherent in factorable functions and their partial derivative arrays. Moreover, while it is true that some symbolic differentiaters also can recognize functions which can be described similarly as a sequence of rules, each of which can be differentiated, the similarity ends there. Such symbolic differentiaters continue to differentiate the rules, without exploiting the polyadic structure of the result (Kedem 1980; Rall 1980; Wengert 1964; Reiter and Gray 1967; and Warner 1975). It is this latter effort which provides

the real value of factorable functions and which therefore separates the two techniques.

See

- ▶ Mathematical Programming
- ▶ Nonlinear Programming

References

DeSilva, A., & McCormick, G. P. (1978). *Sensitivity analysis in nonlinear programming using factorable symbolic input* (Technical Report T-365). The George Washington University, Institute for Management Science and Engineering, Washington, DC.

Emami, G. (1978). Evaluating strategies for Newton's method using a numerically stable generalized inverse algorithm. *Dissertation*, Department of Operations Research, George Washington University, Washington, DC.

Falk, J. E. (1973). *Global solutions of signomial problems* (Technical report T-274). George Washington University, Department of Operations Research, Washington, DC.

Falk, J. E., & Soland, R. M. (1969). An algorithm for separable nonconvex programming problems. *Management Science, 15*, 550–569.

Fiacco, A. V., & McCormick, G. P. (1968). *Nonlinear programming: Sequential unconstrained minimization techniques*. New York: John Wiley.

Ghaemi, A., & McCormick, G. P. (1979). *Factorable symbolic SUMT: What is it? How is it used?* (Technical Report No. T-402). Institute for Management Science and Engineering, George Washington University, Washington, DC.

Ghotb, F. (1980). Evaluating strategies for Newton's method for linearly constrained optimization problems. *Dissertation*, Department of Operations Research, George Washington University, Washington, DC.

Hoffman, K. L. (1975). *NUGLOBAL-User guide* (Technical Report TM-64866). Department of Operations Research, George Washington University, Washington, DC.

Jackson, R. H. F., & McCormick, G. P. (1986). The polyadic structure of factorable function tensors with applications to high-order minimization techniques. *Journal of Optimization Theory and Applications, 51*, 63–94.

Jackson, R. H. F., & McCormick, G. P. (1988). Second-order sensitivity analysis in factorable programming: Theory and applications. *Mathematical Programming, 41*, 1–27.

Jackson, R. H. F., McCormick, G. P., & Sofer, A. (1989). *FACTUNC, A user-friendly system for optimization* (Technical Report NISTIR 89–4159). National Institute of Standards and Technology, Gaithersburg, Maryland.

Kedem, G. (1980). Automatic differentiation of computer programs. *ACM Transactions on Mathematical Software, 6*, 150–165.

Leaver, S. G. (1984). Computing global maximum likelihood parameter estimates for product models for frequency tables involving indirect observation. *Dissertation*, The George Washington University, Department of Operations Research, Washington, DC.

McCormick, G. P. (1974). *A minimanual for use of the SUMT computer program and the factorable programming language* (Technical Report SOL 74–15). Department of Operations Research, Stanford University, Stanford, California.

McCormick, G. P. (1976). Computability of global solutions to factorable nonconvex programs: Part I–Convex underestimating problems. *Mathematical Programming, 10*, 147–175.

McCormick, G. P. (1983). *Nonlinear programming: Theory, algorithms and applications*. New York: John Wiley.

McCormick, G. P. (1985). *Global solutions to factorable nonlinear optimization problems using separable programming techniques* (Technical Report NBSIR 85–3206). National Bureau of Standards, Gaithersburg, Maryland.

Miele, A., & Gonzalez, S. (1978). On the comparative evaluation of algorithms for mathematical programming problems. In O. L. Mangasarian et al. (Eds.), *Nonlinear programming* (3rd ed., pp. 337–359). New York: Academic Press.

Mylander, W. C., Holmes, R., & McCormick, G. P. (1971). *A guide to sumt-version 4: the computer program implementing the sequential unconstrained minimization technique for nonlinear programming* (Technical Report RAC-P-63). Research Analysis Corporation, McLean, Virginia.

Pugh, R. E. (1972). A language for nonlinear programming problems. *Mathematical Programming, 2*, 176–206.

Rall, L. B. (1980). Applications of software for automatic differentiation in numerical computations. *Computing, Supplement, 2*, 141–156.

Reiter, A., & Gray, J. H. (1967). Compiler for differentiable expressions (CODEX) for the CDC 3600 (MRC Technical Report No. 791). University of Wisconsin, Madison, Wisconsin.

Shayan, M. E. (1978). A methodology for comparing algorithms and a method for computing mth order directional derivatives based on factorable programming. *Dissertation*, Department of Operations Research, George Washington University, Washington, DC.

Sofer, A. (1983). Computationally efficient techniques for generalized inversion. *Dissertation*, Department of Operations Research, The George Washington University, Washington, DC.

Warner, D. D. (1975). *A partial derivative generator, computing science* (Technical Report No. 28). Bell Telephone Laboratories, Murray Hill, New Jersey.

Wengert, R. E. (1964). A simple automatic derivative evaluation program. *Communications of the ACM, 7*, 463–464.

Failure-Rate Function

The failure rate at time t of a unit with lifetime density $f(t)$ and lifetime CDF $F(t)$ is defined by the (approximate) probability $h(t)\Delta t$ that a random lifetime ends in a small interval of time Δt, given that

it has survived to the beginning of the interval. For the continuous case, this is formerly written as

$$h(t) = \lim_{\Delta t \to 0} \left[\frac{\Pr\{\text{failure in}(t, t+\Delta t) | \text{survival up to } t\}}{\Delta t} \right] = \frac{f(t)}{1 - F(t)}.$$

The function $h(t)$ is often also called the hazard rate function, the force of mortality, or the intensity rate function.

See

- ▶ Distribution Selection for Stochastic Modeling
- ▶ Reliability Function
- ▶ Reliability of Stochastic Systems

Farkas' Lemma

Given a matrix $\boldsymbol{A}$ and a column vector $\boldsymbol{b}$, one and only one of the following two alternatives holds. Either: (1) there exists a column vector $\boldsymbol{x} \geq \boldsymbol{0}$ with $\boldsymbol{Ax} = \boldsymbol{b}$, or (2) there exists an unrestricted row vector $\boldsymbol{y}$ for which $\boldsymbol{yA} \geq \boldsymbol{0}$ and $\boldsymbol{yb} < \boldsymbol{0}$. This lemma can be proved by defining appropriate primal and dual linear-programming problems and applying the duality theorem.

See

- ▶ Gordan's Theorem
- ▶ Strong Duality Theorem
- ▶ Theorem of Alternatives

Farrell Measure

- ▶ Data Envelopment Analysis

Fast Fourier Transform

An efficient algorithm for computing the discrete Fourier transform and its inverse.

Fathom

To analyze a computational path in enough detail to logically conclude that the analysis of the path has provided as much information possible and/or required.

See

- ▶ Branch and Bound

Fat-Tailed Distribution

- ▶ Heavy-Tailed Distribution

FCFS

The First-Come, First-Served queueing discipline in which customers are selected for service in the precision order in which they arrive to the queue.

See

- ▶ Queueing Theory

Feasible Basis

A basis to a linear-programming problem that yields a solution that satisfies all the constraints of the problem.

See

- ▶ Linear Programming
- ▶ Simplex Method (Algorithm)

Feasible Region

The set of points that satisfy prescribed restrictions (constraints) on a solution.

Feasible Solution

A solution to an optimization problem that satisfies its constraints. In linear programming, these are the conditions $\boldsymbol{Ax} = \boldsymbol{b}$ and $\boldsymbol{x} \geq 0$.

See

- ▶ Infeasible Solution
- ▶ Linear Programming

FEBA

Forward edge of a battle area.

See

- ▶ Battle Modeling

Feedback Queue

A system where customers may return upon completion of service. In many real problems, there is a nonzero probability that a customer just completing service returns to the end of the queue and is serviced again.

See

- ▶ Networks of Queues
- ▶ Queueing Theory

Field Analysis

Howard W. Kreiner
Center for Naval Analyses, Alexandria, VA, USA

Introduction

Field analysis is the practice of operations research usually at the place where the operations occur. It uses observations and data from those operations as they are carried out by the people who normally conduct them. Its purpose may be the immediate modification of an unsatisfactory process, or at longer range, the elucidation of the critical steps in the process for further analysis of options and changes—its vital importance to systems analysis lies in this latter alternative.

As the operations under study are real and current, they involve the use of equipment or machines already in place, and the practices of operators who have been trained in their use. Projections of possible future capabilities and alternative training methods are not a major part of the basic data on which the analysis must be based.

Problems that are visible are those that obviously interfere with the smooth functioning of the system. Their solution must make major differences to be considered useful. There is a need to seek a solution that can improve matters by hemibels (about a factor of 3). Anything less may be lost in the noise of the system. In investigating the causes of problems, many of the potential variables will not lie within the control of the operators or the analyst as attempts are made to identify them. The analyst must think in heuristic terms, rather than those of full scientific rigor.

Case studies often are interesting for their problem-solving methodologies, but as the problems differ in detail, such case studies do not fall into clear groupings that can be characterized as the essence of field analysis. The mathematical content of many, if not most field analyses, is simple, usually not beyond the level of undergraduate mathematics. Therefore, it is methods of behavior, thought, and exposition rather than mathematics that truly constitute the methodology of field analysis.

Historical Origins

The term field analysis arose in the earliest activities of operations analysts in the U.S. Navy. The Navy set up the nation's first operations research organization in 1942. It did so because it was engaged with the Germans submarine forces in a battle that was not going well. Forces, doctrine, and tactics that had grown from experience were proving insufficient to defeat the enemy. The people it brought into the operations research organization, however, were civilian scientists with little or no direct experience of

naval operations. There was only the hope that a fresh scientific view of the situation might produce new methods and the means of victory. The hope was based upon successes of operations research in the British effort against the German bombing campaign.

The scientists at first worked with the statistics of combat derived from action reports. They soon found, though, that full understanding of the action reports required that they have closer contact with the operating forces that engaged the enemy and wrote the reports.

To obtain this contact, they sent scientists to the naval commands deployed against the German submarines. Their purpose was to talk directly with the naval officers and men who performed the combat activities, and to the degree possible, to observe at first hand the circumstances of warfare. Initially, their efforts were intended to insure that the scientists at the home office made proper interpretations of the reports and the statistics they derived. As they became more experienced, and as the aims of the operations research progressed toward model development and predictions of effectiveness, the analysts' purposes and roles with the operating forces also broadened. This pattern of visits and later, longer assignments, became known as the field program, and the analysis done by the analysts at the deployed commands, as the field analyses.

At the end of the war, Morse and Kimball (1946) characterized the program's purposes as: (a) direct help to the service units, (b) securing difficult-to-obtain information for the headquarters organization, and (c) providing to the individual analyst the practical education indispensable in avoiding the pitfalls to which the pure theorist may be subject. They also commented on the administrative factors that made for successful field work. They stressed the need for the command receiving the analyst to invite the assignment and to approve the individual. The analyst should be attached to the highest level of the field activity, take assignments from the commanding officer, and make reports at the same level. There should be regular rotation of analysts at the field command, to bring back to the central staff the experience gained in the field.

Morse and Kimball typified the nature of the scientific work of field analysts in six categories:

1. Analytical
2. Statistical
3. Liaison
4. Experimental
5. Educational
6. Publication

At any field assignment station, however, work would not be limited to just one of these categories; the analysts might do all these types of work in some proportion.

The system of field assignments worked well under the pressure and circumstances of the war. Problem areas were of vital urgency. Help from this promising source was generally welcome at the field commands (though there were instances where it took dramatic analytic successes to establish acceptance for the analyst), and in the senior levels of the service. One instance of initial reluctance overcome by analytic success was Steinhardt's development of barriers against South Atlantic blockade runners (Tidman 1984).

At the end of World War II, the postwar successor to the U.S. Navy's Operations Research Groups, the Operations Evaluation Group continued the practice of field assignments and field analyses. Because of a reduction in the size of the group, and because the activities of the Navy's deployed forces also were greatly curtailed, the field assignments were limited to units of the Navy's test and evaluation forces. The start of the Korean War caused an increase in the size of the parent group, and a revival of the assignment of analysts to fleet staffs and combat operational units.

The activities of operations analysts in the field were not limited to the U.S. Navy. The U.S. Army, both ground and air forces (and later, the U.S. Air Force), also formed analytic groups that sent representatives to field forces. Not all followed precisely the same administrative procedures, but the principal purposes of the assignments were paralleled in each case. Postwar, also, these organizations continued the practice, and expanded their field activities as war and other circumstances required.

The first paper published by the *Journal of the Operations Research Society of America* to report on field analysis in a non-military subject was Thornthwaite (1953). Among operations research practitioners used to the military version of the profession, it was a great relief and cause for elation. It showed that there truly was a possibility that the kind of operations research, field analysis, with which they were most familiar in the military services also could

be applied successfully in the non-military world. More than forty years later, the methods it describes are in use in unchanged form at the site where they were developed. Kreiner (1994) revisits Thornthwaite's paper to fill and clarify gaps in its exposition and make explicit the qualities it exhibits as a fine example of field analysis.

Since then, many other examples of good field analyses have appeared in various formats and publications. As remarked above, they generally have been identified primarily by the subject matter of the problem, rather than as examples of field analysis viewed as a separately defined branch of operations research.

Field Analysis in an Era of Systems Analysis

During World War II, there was a very close tie between the work of the military field analyst and that of the headquarters staff. The initial motivation for creating a field program was exactly that close tie; headquarters staff interest was directed almost exclusively to the day-to-day problems and success of the deployed forces. The guiding principles of operations research formulated in the postwar summaries were identical for the field and the headquarters analyses. In the years since the U.S. creation of formal operations research organizations, this has continued to be true when war dominates the activities of the military services from combat forces to the highest command levels. It also was true to a large extent in peacetime immediately after World War II. The field analysts' assignments were to operational test and evaluation commands concerned with individual combat systems whose procurement decisions depended upon those test results.

The tie has become less close in the military services with the trend to high-level systems analysis at headquarters operations research groups. The increasing complexity and interaction of systems, their cost, and the very long development time for newer combat systems made headquarters command levels more concerned with future systems. It elevated the procurement process to the strategic level, and concentrated the attention of the central military staff on long-term budgetary matters. Headquarters operations research groups necessarily altered their point of view as well. Field commands, however, retained their concern with training their forces to operate and integrate systems already in use. The problem of divergence of interest between the field analyst and the headquarters group did not disappear entirely even during the Vietnam War and the Gulf War. Those wars were limited in character, and the Cold War and the larger threat of nuclear war still tended to dominate budgetary and strategic interests.

For the field analyst, however, the main subjects for analysis continue to be the operations involving the use of equipment already designed, developed, produced and distributed. If the equipment is not quite at this stage, it is at least far enough along in the process to justify operational testing and tactical development. The field analyst's concerns in commercial, nonmilitary government or military activities, are with the practices for employing, and with the training of people to make the equipment and its use as effective and efficient as possible. To the extent that a central group, commercial or military, arranges to provide field analysts and uses the field assignments both for training analysts and to insure realism in describing current and possible future operations, it shares the same objectives. It can be difficult for a central group, however, to balance priorities for attention to field analysis with those of important future systems studies.

A Continuing Role for Field Analysis

Compared with the circumstances at the time operations research first was introduced, there now are greatly improved methods for data collection, and greatly enhanced ability of computers to model interactions of equipment and people. There have been theoretical developments in operations analysis and problem-solving techniques that are reported worldwide in the journals of numerous operations research societies. Yet, if operations research retains its focus on problem solving in operations, there continues to be an important mission for analysis to be done at the point, and in direct observation of the operations under study. This is particularly important when large-scale modeling of operations is a major means of analysis in the headquarters groups. Both during the building of models and afterwards, it can be extremely difficult to review and test all the assumptions and possible omissions of critical factors.

Morse and Kimball made the point that it was a strength of the operations analyst to think in hemibels, to seek improvements in operations that multiply effectiveness by a factor of three or more. This differs qualitatively from the notion of improvements in small increments. The field analyst is in a unique position to see opportunities for hemibel improvements; the analyst she can observe at first hand the factors that control the operation. If there are differences between what has been assumed about those factors and what actually is occurring, the analyst can document them, measure them, and propose changes to exploit the differences in favor of improved understanding, and ultimately, improved operations.

Kreiner (1992) noted an example in radar detection of small targets, in which the field analyst identified assumptions about the statistical character of radar returns. Current data at the field site proved the assumptions to be faulty. Ultimately, the original theory had to be abandoned, and alternate methods devised. Kreiner (1992) also reports on an analytical look at an operational plan that revealed unstated, unconsidered, and erroneous assumptions. The plan assumed static, fixed naval forces, assured of long warning times of possible attack, and manned by pilots expected to fly missions they considered suicidal, although safer, equally effective alternatives were available. When the analysis made these assumptions explicit, the entire plan had to be rewritten.

The field analyst also is in a better position to examine the choice of measures for evaluating operational effectiveness than any counterpart located at headquarters. Larson (1988), though not assigned as a field analyst, nevertheless functioned as one as a customer of a queueing system when he attempted to buy a bicycle for his daughter. In his job as an analyst of such systems, he had accepted the standard measure of average customer waiting time, and the goal of minimizing this measure. As an actual customer, however, he discovered a major deficiency in the measure, a lack of perceived fairness to the individual customer. His article explores the ways to make the queues and the measures of their performance more responsive to the broader interpretation of effectiveness.

The field analyst has another important requirement, the need to develop results in terms that serve primarily the purposes of the customer. Scientific journals, including those of the operations research societies, require presentations that are concise, rigorous, and of enough generality to be of interest to a spectrum of fellow professionals. The field analyst has another audience entirely, that is, the analyst must initially establish a role as a participant in the operations analyzed, lest the activities lack credibility. As an outsider, the analyst may not gain access to the intimate, seemingly tiny details that make up the operation. The field analyst similarly must make reports in operational terms. The audience will classify problems in their own terms, rather than by the methodologies used to solve them. The report must make the same close connection with the immediate problem. It may be important to the analyst to innovate in methodology. The client wants only assurance that the methodology addresses the correct aspects of the problem, that the analyst is competent to apply it, and that the results enable them to improve their operations.

See

- ▶ Air Force Operations Analysis
- ▶ Air Force Operations Research
- ▶ Center for Naval Analyses
- ▶ Implementation
- ▶ Military Operations Research
- ▶ Operations Research Office and Research Analysis Corporation
- ▶ Practice of Operations Research and Management Science
- ▶ RAND Corporation

References

Kreiner, H. W. (1992). *Fields of operations research*. Baltimore: Operations Research Society of America.

Kreiner, H. W. (1994). Operations research in agriculture: Thornthwaite's classic revisited. *Operations Research, 42*, 987–997.

Larson, R. C. (1988). There's more to a line than its wait. *Technology Review, 91–5*, 60–67.

Morse, P. M., & Kimball, G. E. (1946). *Methods of operations research, OEG report 54*, Office of the Chief of Naval Operations, U.S. Navy Department, Washington, DC. (Reprinted as: Morse, P. M., & Kimball, G. E. (2003). *Methods of operations research*. Mineola/New York: Dover Publications).

Thornthwaite, C. W. (1953). Operations research in agriculture. *Journal of Operations Research Society of America, 1*, 33–38.

Tidman, K. R. (1984). *The operations evaluation group*. Annapolis, MD: Naval Institute Press.

FIFO

The First-In, First-Out queue discipline in which customers are taken out of the line for service in the exact order in which they arrived (meant to be equivalent to the first-come, first-served scheme).

See

- ▶ FCFS
- ▶ Queueing Theory

Financial Derivative

- ▶ Financial Engineering

Financial Engineering

John R. Birge
The University of Chicago, Chicago, IL, USA

Introduction

Financial engineering refers to the application of scientific methods to the design, analysis, and implementation of financial products and services. The applications within this domain extend from trading at the nano–second timescale to informing investment decisions that can stretch across centuries. Much of financial engineering can be aligned with classical methods in operations research, including the following functions: (i) asset–pricing, particularly for derivatives, relying heavily on stochastic modeling and simulation; (ii) portfolio optimization and asset–liability management, using various methods from optimization; (iii) trading and hedging, often involving dynamic programming and control techniques; and (iv) risk management, involving many of the principles of reliability theory. The following sections describe basic applications within each of these areas and the associated relevant operations research methods.

Asset–Pricing and Derivative Evaluation

Much of the interest in financial engineering began with the widespread use of the Black–Scholes–Merton formula for option pricing following the publication in the academic literature of their seminal papers describing this pricing method (Black and Scholes 1973; Merton 1973a), although the concepts appeared earlier starting with Bachelier (1900). The formula applied directly to the basic (European) call and put options, which provide the buyer the right (but the obligation) to purchase (for a call) or sell (for a put) an asset (e.g., a share of a stock) at a fixed price (called the strike or exercise price) at a future expiration (or maturity) date. Extensions of this methodology have been applied to many other types of derivative securities that derive their value from the price of a set of intrinsic (or underlying) assets over some specified period of time. They are also referred to as contingent claims since their payoff depends on some unknown outcome.

The basis for this and other asset–pricing models is an application of linear programming duality theory, known as the Fundamental Theorem of Asset Prices (Harrison and Pliska 1981), which states that either the securities market admits an arbitrage opportunity to earn a non–negative risk–free payoff in all future states (and a positive payoff in some state) without any initial investment or there exists a price or weighting on each future state that in expectation yields the current market price for any security in the market. Most pricing formulae can then be derived from assuming that the market does not allow arbitrage (at least for any extended period of time) and that the prices of securities not currently in the market (or whose market prices might be stale or suspect) can be evaluated from the prices of other securities. If markets are complete in representing all possible future cash flows, then prices should be unique in this framework, but even if the market is incomplete, an assumed absence of arbitrage can imply bounds on prices that are consistent with the market, e.g., settings with physical assets such as energy and commodities (Staum 2008).

As an example of the asset–pricing theorem, suppose an asset has future payoff $\mathbf{S}$ distributed as S_i with probability p_i at time $T = 1$ for $i = 1, \ldots, N$ for N future states of the market. (Continuous future cash flow distributions can be modeled in the same way with a more general linear programming model).

Absence of arbitrage implies that no one in the market can purchase or sell an equivalent cash flow to **S** in such a way that always produces non–negative and, with some positive probability, positive surpluses. This result means, in particular, that a buyer cannot sell or purchase at unit price s an equivalent cash flow to a share x of **S** with y in other assets (with current prices f and future prices F_i in state i) and risk–free investing (B at a risk–free rate r that yields $e^r B$ in the future with certainty) and obtain positive future cash flows, i.e., the maximum expected value of a future position for buying a share x with no losses is zero:

$$0 = \max_{x,y,B} \sum_i p_i (S_i x - F_i^T y - e^r B) \tag{1}$$

$$\text{subject to} - sx + f^T y + B = 0, \tag{2}$$

$$- S_i x + F_i^T y + e^r B \leq 0 \, \forall i, \tag{3}$$

$$0 \leq x \leq 1, \tag{4}$$

where the superscript "T" denotes the transpose operator. The feasibility of the dual problem when the value of (1)–(4) is bounded above (no arbitrage) then provides the Fundamental Theorem of Asset Pricing. The dual is:

$$0 = \min_{\lambda \leq 0, \pi \geq 0, \rho} \rho \tag{5}$$

$$\text{subject to} \sum_i S_i p_i + \lambda s + \sum_i \pi_i S_i - \rho \leq 0, \tag{6}$$

$$- \sum_i F_i p_i - \lambda f - \sum_i \pi_i F_i \geq 0, \tag{7}$$

$$- e^r - \lambda - \sum_i \pi_i e^r = 0. \tag{8}$$

For an optimal dual solution, $(\lambda^*, \pi^*, \rho^*)$, $\lambda^* < 0$ (where the strict inequality follows from (8)), $\pi^* \geq 0$, and $\rho^* = 0$;

$$s \leq \sum_i S_i (p_i + \pi_i^*)/(-\lambda^*); \tag{9}$$

$$f = \sum_i (p_i + \pi_i^*) F_i / (-\lambda^*); \tag{10}$$

$$- \lambda^* = e^r \left(1 + \sum_i \pi_i^* \right). \tag{11}$$

If a strictly positive $x^* > 0$ solves (1)–(4), then the inequality (9) must be tight by complementarity. This condition would follow if the cash flow in S_i can be reproduced perfectly by $F_i^T y$ and $e^r B$ with $f^T y + B = sx$, which holds if the market is complete. This then implies

$$\begin{aligned} s &= e^{-r} \sum_i s_i \frac{p_i + \pi_i^*}{1 + \sum_i \pi_i^*} \\ &= e^{-r} \sum_i s_i \frac{p_i + \pi_i^*}{\sum_i (p_i + \pi_i^*)} = e^{-r} \sum_i s_i q_i^*, \end{aligned} \tag{12}$$

where $q_i^* \geq 0$ and $\sum_i q_i^* = 1$, i.e., $\{q_i^*\}$ define a probability distribution (called a risk–neutral or equivalent martingale measure) on S_i for the price s, which is also consistent with the other assets so that $f = e^{-r} \sum_i F_i q_i^*$. In a complete market, everything could be priced this way with the same probabilities; or state prices as in the general equilibrium of Arrow and Debreu (1954). If the market is not complete, then a range of prices $s \in [s_L, s_U]$ would be consistent with a cash–flow that is not completely represented in the market.

The basic principles of replicating a cash flow from existing market instruments with an assumption of no arbitrage or of using a consistent set of prices underly most asset–pricing applications. The following section on portfolio optimization discusses asset pricing theories — the Capital Asset Pricing Model (CAPM) and Arbitrage Pricing Theory (APT) — based on market agents' preferences, but the basic consequences of consistent prices with no arbitrage opportunities account for much of actual pricing practice in financial engineering; see Derman and Taleb (2005) for a discussion of such practical approaches to pricing.

To see the implications of consistent prices, consider the pricing of a simple European call option to purchase a share of a non–dividend–paying stock at time T at a price K. In a complete market with no arbitrage, the Fundamental Theorem of Asset Pricing implies that the value of the call (or premium) C_0 at time $t = 0$ is the present value of the expected future

cash flows using the consistent (risk–neutral or equivalent martingale) probability density q_T on future states at time T that exists from the theorem. Since the payoff of the call option is $(S_T - K)^+$ when the price of a share is S_T at T, the theorem implies

$$C_0 = e^{-rT} \int_0^\infty (S_T - K)^+ q_T(S_T) dS_T. \tag{13}$$

The next step then is to determine the appropriate distribution represented by q_T.

For the distribution of S_T given by q_T, if the risk–free rate r is constant, the no–arbitrage property implies that from any time t when the share has price S_t, the conditional expectation of S_T under q_T discounted back to t must be S_t, i.e.,

$$S_t = \int_0^\infty e^{-r(T-t)} S_T q_T(S_T) dS_T, \tag{14}$$

which is the martingale property of the process $Y_\tau = e^{-r\tau} S_\tau$, such that $Y_s = E_Q(Y_\tau | Y_s)$ for any $\tau \geq s$, where Q represents the probability distribution implied by the no–arbitrage condition at each time τ. This means that, under Q, the prices of all non–dividend paying assets increase at the same exponential rate r. Now, if price changes are the result of random arrivals of new information that push prices up or down in a way that does not change in terms of relative increases or decreases in price over time, i.e., the price changes have the features of a random walk, then, as the arrival rate of information increases, the prices should follow geometric Brownian motion (GBM). Assuming no dividends, a constant risk–free rate, and no changes in the information arrival rate or its effects (which keeps the volatility in the process constant), prices under the distribution given by Q would obey the following stochastic differential equation:

$$dS_t = rS_t dt + \sigma S_t dW_t, \tag{15}$$

where σ is the volatility of the price process and W_t is a standard Brownian motion. If S_t follows the dynamics of (15), then $\log S_T - \log S_0$ is normally distributed with mean $(r - \sigma^2/2)T$ and variance σT, i.e., S_T is log–normally distributed under Q with mean $e^{rT} S_0$ and variance $e^{2rT}(e^{\sigma^2 T} - 1) S_0^2$. This result leads to the Black–Scholes–Merton (BSM) option pricing formula (Black and Scholes 1973; Merton 1973a) by evaluating the integral in (13) with q_T as the log–normal density:

$$C_0 = S_0 \Phi\left(\frac{\log(S_0/K) + (r + \sigma^2/2)T}{\sigma\sqrt{T}}\right) - e^{-rT} K \Phi\left(\frac{\log(S_0/K) + (r - \sigma^2/2)T}{\sigma\sqrt{T}}\right), \tag{16}$$

where Φ is the standard normal cumulative distribution function.

The formula in (16) can be derived in many ways. One approach is to start with the physical model of the dynamics of the price S_t and of the call option value C_t and to note that C_t can be dynamically replicated by continuously adjusting a fraction of S_t and a risk–free bond (which assumes that trading is frictionless or has no transaction cost). An equivalent approach is to model a random walk in prices directly, to solve a dynamic program recursion that gives the fraction to hold of S_t and the risk–free bond at each point to replicate the option, and then to take the limit of this value as the discretization interval reduces to zero. Each approach yields the same formula and the observation that the price does not depend on investors' risk attitudes.

The significance of the BSM formula is that the option price can be found without making an assumption on risk preferences. The actual physical distribution of S_T would be different from that given by q_T if the market includes a non–zero risk premium, but the evaluation can be done under q_T without deciding what that premium should be. The formula does, however, rely on several assumptions, such as a constant risk–free rate, constant volatility, the log–normal density at any point in time, and the completeness of the market to ensure the existence of the equivalent risk–neutral distribution. While much of the work in financial engineering considers deviations from these assumptions, the basic framework provides an efficient description of prices and a methodology for checking price consistency.

From the formula in (16), European put option prices (i.e., the option to sell at a given price) can be found again with a no–arbitrage principle of put–call parity on non–dividend paying assets that, since the payoff of a call minus a puton the same asset at the

same exercise price K and maturity T is the same as the payoff of the asset minus a risk–free bond paying K at T, the present values must be the same as well, i.e.,

$$C_t - P_t = S_t - e^{-r(T-t)}K, \tag{17}$$

where r is the risk–free discount rate (from t to T). Other relationships can be found to bound prices of options at different exercise prices and maturities.

A main emphasis of financial engineering is to determine prices of different types of derivatives and to relax some of the assumptions involved in (16), e.g., see Hull (2011). In general, analytical formulae for payoffs at a fixed point in time can be found under the basic assumptions plus a variety of extensions (some of which require additional assumptions or information on risk preferences to obtain the formulas) including dividends; stochastic risk–free rate; stochastic volatility; different price dynamics, including time–varying drift, non–linear volatility, processes with jumps, and general Lévy processes (Carr et al. 2003; Cont and Tankov 2004; Wu 2008); and different payoff structures, including different linear payoff structures, payoffs on multiple assets (including different currencies and interest rates), 0–1 (digital options), and other non–linear payoff structures, including bonds and convertible bonds.

In addition to derivatives that depend on asset values at a fixed point in time, other derivatives can depend on entire sample paths of prices. Basic examples of such path–dependent options are American options that are identical to the European call and put options described here except that they can be exercised at any time until maturity. They effectively then have two components, the European option value plus a premium for early exercise. In some cases, such as an American call option on a non–dividend paying under the assumptions for the basic BSM model, the early exercise premium is zero, but, in other cases, including the simple American put option, the early exercise premium is positive and can be difficult to compute. These options do not have simple analytical formulas for evaluation and require an approximation or numerical method.

Other path–dependent options include Asian options that depend on the average asset price until maturity, barrier options that depend on whether the asset price ever crosses a threshold (or thresholds), and lookback options that depend on the extreme (minimum or maximum) prices achieved by the option. In these cases, Laplace transforms can yield analytical expressions for these derivatives that can then be inverted (Craddock et al. 2000; Kou 2008a, b). Other analytical approaches include spectral methods that, for example, yield formulas for Asian option values (Linetsky 2004, 2008), and fast Hilbert transform methods (Feng and Linetsky 2008).

In more general and complex settings, pricing can require either numerical solutions of systems of partial differential equations (PDE) or methods based on Monte Carlo simulation. For PDE methods, basic methods in a financial context appear in Tavella and Randall (2000) and approaches appropriate for more general conditions are given in Feng et al. (2008). For Monte Carlo methods, Glasserman (2004) provides a comprehensive review of this methodology's use in all areas of financial engineering.

Portfolio Optimization and Asset–Liability Management

The discussion of the Fundamental Theorem of Asset Pricing in the previous section on pricing applications relied on the absence of arbitrage, i.e., the inability to construct a portfolio of market securities to obtain risk–less profits relative to any given market security. In practice, portfolios include varying levels of investment risk in exchange for a premium on the return of the portfolio. A goal in portfolio construction is then to find combinations to form an efficient portfolio that obtains the lowest possible level of risk for a given return.

Markowitz (1952) formulated this problem of finding an efficient portfolio as a quadratic program to minimize the variance of portfolio returns over a fixed time period subject to meeting a constraint on the expected return over that period. If the proportion of wealth invested in asset i is x_i for $i = 1, \ldots, n$, the expected return of asset i is r_i with $\mathbf{r} = (r_1, \ldots, r_n)^T$, and the covariance between returns on assets i and j is σ_{ij}, where $\sigma_{ii} = \sigma_i^2$, the variance of the return on asset i, with $\mathbf{\Sigma} = [\sigma_{ij}]$, then the Markowitz mean–variance model with return target level r_0 isto find $\mathbf{x} = (x_1, \ldots, x_n)^T$ and portfolio variance $v(r_0) =$

$$\min \mathbf{x}^T \mathbf{\Sigma} x \tag{18}$$

$$\text{subject to } \mathbf{r}^T\mathbf{x} \geq r_0, \tag{19}$$

$$\mathbf{e}^T\mathbf{x} = 1, \tag{20}$$

where $\mathbf{e}$ is a column vector containing all 1's. The set of solutions $(r_0, v(r_0))$ defines the mean–variance efficient frontier of portfolios. Much effort in financial engineering involves finding efficient portfolios using variations on the model in (18)–(20). The basic extensions of the mean–variance model include variations in the utility (which is quadratic here) or in the assumptions on the distribution of returns. The objective represents a wide–range of risk preferences or utilities if returns are normally distributed (since the return distribution of the portfolio is normal as well and is determined by its mean and variance), but if the distribution of returns is not normal, then the quadratic form of preferences appears more restrictive (although it could still be approximately correct). Any implementation of (18)–(20) also requires estimates of Σ and $\mathbf{r}$, which can present some difficulties. Some of the work on portfolio optimization in financial engineering includes compensation for estimation errors, as discussed in Kan and Zhou (2007).

Other approaches consider different utility functions, which are particularly relevant for non–normal distributions. One approach assumes that $\mathbf{\Sigma}$ and $\mathbf{r}$ are only known to be within some range (uncertainty set). This approach is called robust portfolio optimization (Fabozzi et al. 2007; Cornuéjols and Tütüncü 2007). Other alternatives consider different utility forms and non–Gaussian return distributions.

An implication of the mean–variance model is that, if everyone in the market believes the assumptions leading to (18) and chooses an efficient portfolio, then, in equilibrium, the price of each asset in the market simply reflects that asset's contribution to the portfolio's risk. The result is the Capital Asset Pricing Model (CAPM), pioneered by Sharpe (1964) and Lintner (1965), which states that, in an equilibrium, if the market corresponds to an efficient portfolio with expected return r_m and variance on return σ_m^2, then the expected return of any asset i is given by:

$$r_i = r_f + \beta_i(r_m - r_f), \tag{21}$$

where r_f is the risk–free rate and $\beta_i = \sigma_{im}/\sigma_m^2$.

Many of the pricing models in financial engineering use the CAPM as a basis for analysis (although not necessarily as a direct pricing method). Empirical tests of the CAPM suggest that prices may not completely reflect (21) using standard proxies for the market return such as a stock index, but it is not clear how to obtain or measure r_m, the return on the entire market (Fama and MacBeth 1973; Ferson 2003). The market portfolio may, however, be explained by multiple risk factors such as an index and the differences in portfolios of firms based on their size and leverage (Fama and French 1993). This version would suggest that

$$r_i = r_f + \mathbf{b_i}^T\lambda, \tag{22}$$

where λ is the expected excess (i.e., above r_f) returns on the relevant set of risk factors $\mathbf{b_i}$. This conclusion on prices also arises from the Arbitrage Pricing Theory (APT) of Ross (1976) that does not require all investors in the market to follow a mean–variance strategy (but that relies on the availability of many assets). In any case, the relationship in (22) can provide a pricing tool for financial engineering, and also an alternative for constructing optimal portfolios, since the large estimation requirements in determining Σ and $\mathbf{r}$ are replaced with a reduced set for finding the $\mathbf{b_i}$ coefficients. In addition, a financial engineer may have a particular view about a certain asset such that $r_i - r_f = \alpha_i + \mathbf{b_i}^T\lambda$ for some non–zero α_i. In that case, these views can be combined with the data estimates in $\mathbf{b_i}$. A consistent methodology for this is the approach in Black and Litterman (1992), which imposes views on top of a prior based on the CAPM to obtain a posterior distribution on returns.

Another difficulty in the form of (18) is that this model is purely static, while portfolios change dynamically over time. A dynamic version of the mean–variance portfolio considers the utility of a portfolio that is continuously adjusted over time. If the prices of the assets all follow a process like (15) and the drift is r_i for asset i and Brownian motion W_i given with covariances Σ, then the value of a portfolio beginning at time 0, which maintains a fraction x_i in asset i has a value at time T given by

$$w_T = e^{(\mathbf{r}^T\mathbf{x} - \mathbf{x}^T\Sigma\mathbf{x}/2)T + \sqrt{(\mathbf{x}^T\Sigma x)T}Z}, \tag{23}$$

where Z is a standard normal random variable. For power constant–relative risk aversion, the utility

function has the form w^γ/γ for some $\gamma > 0$ (or $log(w_T)$ if $\gamma \to 0$), and the objective function becomes: $E[w_T^\gamma/\gamma] = e^{\gamma(\mathbf{r}^T\mathbf{x} - \frac{1-\gamma}{2}\mathbf{x}^T\Sigma\mathbf{x})}/\gamma$. The utility–maximizing solution in this case is then again a mean–variance efficient portfolio. If an $(n+1)$st risk–free asset is included and the returns given as excess above the risk–free rate, with $x_{n+1} = 1 - \sum_{i=1}^n x_i$, effectively removing the constraint for $\sum_{i=1}^n x_i = 1$, the solution is given by $\mathbf{x}^* = (x_1^*, \ldots, x_n^*)^T = \frac{\Sigma^{-1}\mathbf{r}}{1-\gamma}$ with $1 - \sum_{i=1}^n x_i$ invested in the risk–free asset. This dynamic portfolio can then yield an inter temporal CAPM; see Fama (1970); Merton (1973b); Constantinides (1982); and Duffie (2003) for justification of CAPM pricing in fairly general settings.

In practice, portfolio optimization requires consideration of additional issues such as transaction costs and delays, limited liquidity, and trading restrictions, such as lock–in and vesting periods. These practical considerations complicate models beyond the simple form in (18) and require dynamic considerations over time. In those cases, financial engineering may provide control rules (typically tested with Monte Carlo simulation) or more general stochastic optimization formulations to find $x_t, t = 1, \ldots,$ where $x_t \in X_t$ (some feasible set that may depend on state realizations) to maximize (given x_0)

$$\sum_{t=1}^{\infty} E[f_t(x_{t-1}, x_t, s_t)], \tag{24}$$

where f_t is the utility function, s_t represents a state vector that is revealed over time and the decisions x_t can represent investment, consumption, and trading activity, such as allocations in each asset or asset class, purchases and sales of assets, consumption, and commitments for future sales or purchases. A variety of methods can then be used to solve these problems that take advantage of structural properties of the models; see the summary in Birge (2008).

Trading and Hedging

The dynamic model in (24) represents the general form of models that can also be used for trading assets and hedging their values over time. When the frequency of trading increases, approximations are generally required either in terms of the model parameters (such as the objective) or the range of possible controls. Examples described in this section focus on shorter time horizons than those for the longer–period asset allocation decisions in the previous section. These situations include identifying and exploiting arbitrage opportunities (or market inefficiencies) across markets and securities, minimizing slippage (excess transaction costs) to execute an order, and maintaining a risk–neutral position for a derivative security exposure.

The problem of identifying an exploitable arbitrage requires the rapid consideration of what is often a large number of prices and quick execution of the proposed actions. A simple form of these trading examples might include the comparison of the prices of constituent prices in an index with a security representing the index. Program trading often refers to the practice of automatically identifying discrepancies between these two quantities and executing trades to take advantage of the difference.) As a more computationally–focused application, consider a set of European options (e.g., on an index) on the same underlying, all with the same maturity. Suppose the call options have a highest bid price b_i^c with v_i^c contracts bid at that price and a lowest offered or ask price a_i^c with u_i^c contracts at that price, and suppose corresponding put options with bid prices b_i^p with v_i^p contracts and ask prices a_i^p with u_i^p contracts, all at strike prices, $K_i, i = 1, \ldots, n$. The bid and offered prices should reflect put–call parity and any other relationship implied by the absence of arbitrage, but occasionally trading in each option can lead to inefficiencies that can be exploited (at least to the depth of the orders at bid and offered prices). A linear program can be used to discover these pricing anomalies. Let $x_i^{c\pm}$ and $x_i^{p\pm}$ represent the number of call and put option contracts, respectively, with exercise K_i to trade, where the superscript $+$ indicates a purchase and $-$ indicates a sale. Net borrowing is B at rate r (which can be positive or negative here but could also correspond to two variables with different rates for borrowing and lending). The objective is to maximize payoffs at maturity subject to constraints ensuring no losses:

$$\max \sum_{i=1}^{n+1} \pi_i \tag{25}$$

subject to

$$\sum_{i=1}^{n} (b_i^c x_i^{c-} + b_i^p x_i^{p-} - a_i^c x_i^{c+} - a_i^p x_i^{p+}) - B = 0; \tag{26}$$

$$\sum_{i=1}^{n} (-x_i^{p-} + x_i^{p+}) \geq 0; \tag{27}$$

$$\sum_{i=1}^{n} (-x_i^{c-} + x_i^{c+}) \geq 0; \tag{28}$$

$$\sum_{i=j}^{n} (-(K_i - K_j)x_i^{p-} + (K_i - K_j)x_i^{p+}) +$$

$$\sum_{i=1}^{j} (-(K_j - K_i)x_i^{c-} + (K_j - K_i)x_i^{c+}) + Be^{rT} \geq \pi_i;$$

$$j = 1, \ldots, n; \tag{29}$$

$$0 \leq \pi_i; 0 \leq x^{c-} \leq u_i^c; 0 \leq x^{c+} \leq v_i^c; \tag{30}$$

$$0 \leq x^{p-} \leq u_i^p; 0 \leq x^{p+} \leq v_i^p. \tag{31}$$

Constraints (26) balance cash at origination of the trade. Constraints (27) and (28) ensure no losses outside the range of the exercise prices of options in the market. Constraints (29) give the net proceeds at each exercise price. If a positive objective (25) can be attained, then trading at the current market conditions produces a risk–less surplus at maturity (although the trade is more likely to be un–wound before maturity when the bid and ask orders return to equilibrium).

Pure arbitrages, as when (25) has a positive value, are rare in liquid markets and require rapid execution before additional orders enter the market or previous orders are withdrawn. More common arbitrages are statistical in nature and depend on identifying when current prices depart from an historically valid statistical relationship among prices. Identification of these opportunities may also involve optimization to recognize the strength of the relationship and the departure from the historical conditions and to design a trading strategy that maximizes an objective that may involve risk adjustment.

The identification of patterns or information not yet absorbed in terms of price movements by the market over longer time scales leads to longer–term trading requirements for purchases and sales of varying numbers of assets. When these numbers are large relative to the market size, placing a market order for the full number of shares can have a significant effect on the price and add to the transaction or execution cost (a phenomenon known as slippage). Optimization methods can also be used to minimize slippage by breaking an order for a large number of shares into smaller orders that are submitted sequentially.

Models for optimally choosing these trade sizes balance the price impact of execution with the risk over price changes in delays in completing the execution; examples are given in Bertsimas and Lo (1998) and Almgren and Chriss (2000). In general, these models use a dynamic program, with some approximation. For example, suppose the goal is to buy a total of $s_0 = \sum_{t=1}^{T} s_t$ shares with purchases $s_t \geq 0$ at times $t = 1, \ldots, T$, where the purchase price $p_t(s_t, p_t^-)$ is a function of the number sold s_t and the (ask) price p_t^- immediately before the tth transaction, and the objective is an additive (convex for risk–averse minimization) function, with

$$V_1(p_1^-, s_0) = \min_{s_1, \ldots, s_T \geq 0 : s_0 = \sum_{t=1}^{T} s_t} E\left[\sum_{t=1}^{T} u_t(p_t s_t)\right].$$

The Bellman equation for this dynamic program is

$$V_t(p_t^-, s) = \min_{s_t} \{u_t(p_t(s_t, p_t^-)s_t) + E[V_{t+1}(p_{t+1}^-, s - s_t)|p_t, s_t]\} \tag{32}$$

with boundary condition, $V_T(p_T^-, s) = u_T(p_T(s, p_T^-)s)$. Note that the price dynamics and value function in the future may include some effect of the order size at time t as well as the last price at t. The state space could also be enlarged to include more order book information. With a model of the effect on prices of differently sized orders at different points in time and a model of price dynamics, the dynamic program can be solved to obtain an optimal (with respect to the model) policy.

Another trading function of financial engineers is to maintain a hedging position for a trader with a net position in a given derivative, e.g., either because the trader is a market maker in an exchange–traded

product (and is, therefore, obligated to accept a fraction of the trades within some market responsibility) or has sold an option over–the–counter to a buyer. Much of this trading involves balancing the changes in values from long and short positions using the derivatives of the option prices with respect to the underlying (called delta hedging), but the general optimization framework in (25)–(31) can also be used to minimize the cost in maintaining hedged positions so that net positions do not vary as market conditions change.

Risk Management

Risk management is another broad category of financial engineering practice that generally refers to identifying, analyzing, and controlling the impact of uncertain conditions on financial performance. The types of risks that can affect this performance include:

- Market risk, which generally refers to changes in the prices of assets traded in the market;
- Credit risk, which refers to the likelihood of an economic agent's not fulfilling a financial obligation, such as repaying a loan;
- Liquidity risk, which is sometimes paired with market risk and refers to uncertainty in the depth (or accessibility) of a market for an asset;
- Operational risk, which refers to the risks in executing a process, such as a trade, as intended;
- Behavioral and environmental risk, which is used here to refer to risks not captured in the other areas, such as the prepayment risk of a mortgage loan due to customer choice, changes in mortality affecting the liabilities of a pension fund, changes in the demand for electricity due to weather conditions, and the risk of a regulatory change affecting the tax treatment of an asset;
- Model risk, which is used as another broad category to refer to the risk in basing actions on a model that does not adequately reflect reality.

Comprehensive treatments of the financial engineering applications in these areas appear in McNeil et al. (2005) and, particularly for credit risk, in Duffie and Singleton (2003) and Bielecki and Rutkowski (2004).

The general use of derivatives and optimal asset allocations is often for the purpose of managing risks, such as a firm's exposure to interest rate movements or the exposure of a pension fund to changes in the value of assets and liabilities. The general approaches described in the previous sections can then be used in those circumstances. Other risk exposures focus on the reliability of other market participants. An example of these different types of exposures is the exposure of a bank to credit risk in the form of defaults on loans.

The bank must maintain reserve funds as economic capital to cover losses in the event of defaults. Consider a simple example with n equally–sized loans such that upon default each results in a loss (or loss given default) of L. The bank wishes to reserve sufficient economic capital C to cover all losses until time t with some confidence levelα. Alternatively, the bank requires that no loss greater than C occurs with probability greater than $1-\alpha$ (or C is the α–level Value–at–Risk (VaR) for this portfolio of loans).

The presentation here frames this requirement in terms of reliability theory; see Barlow and Proschan (1975) for fundamentals of reliability theory and D'Amico et al. (2005) for an additional view of loan portfolios from this perspective. In this framework, the bank's loan system fails if C/L or more of the n loans default. This defines a k–of–n system (with $k = \lceil C/L \rceil$). If the state of the system is given by the structure function, $\phi(x, \lceil C/L \rceil, n)$, where $x = (x_1, \ldots, x_n)$ are indicators for the functioning of each loan, then

$$\phi(x,\ \lceil C/L \rceil, n) = 1\left\{\sum_{i=1}^{n} x_i \geq \frac{C}{L}\right\}, \qquad (33)$$

where $1\{\cdot\}$ denotes the indicator function. Now, let $X_i(t)$ be the random variable defined by $X_i(t) = 1$ if the default time T_i of loan i satisfies $T_i > t$, and $X_i(t) = 0$ otherwise. The capital decision is then to find

$$C = \min_{C} \{C | E[\phi(X,\ \lceil C/L \rceil, n)] \geq \alpha.\} \qquad (34)$$

Calculating the reliability depends critically on the correlations between components of X. Vašiček (1987) gave a method for computing $\mathrm{E}[\phi(X, k, n)]$ that is consistent with a CAPM model. This formula assumes that each loan is the debt obligation of an asset with future values consistent with a GBM from a homogeneous set with a uniform correlation ρ between the asset values of any pair of assets.

If each loan in this setup has an equal probability p of defaulting by time t, then

$$\mathrm{E}[\phi(X,k,n)] = \sum_{l=k}^{n} \binom{n}{l} \int_{-\infty}^{\infty} \Phi\left(\frac{\Phi^{-1}(p) - \sqrt{\rho}u}{\sqrt{1-\rho}}\right)^{l} \left[1 - \Phi\left(\frac{\Phi^{-1}(p) - \sqrt{\rho}u}{\sqrt{1-\rho}}\right)\right]^{n-l} d\Phi(u). \tag{35}$$

The limiting distribution on $k = n\theta$ for some fraction θ of all loans as $n \to \infty$ is then

$$\lim_{n\to\infty} \mathrm{E}[\phi(X, n\theta, n)] = \Phi\left(\frac{\sqrt{1-\rho}\Phi^{-1}(\theta) - \Phi^{-1}(\theta)}{\sqrt{\rho}}\right). \tag{36}$$

The formulae in (35) and (36) are useful in extending results for k–of–n systems from standard formulas for independent failures to those with a specific form of correlation, but they require assumptions that are met by few (if any) loan portfolios. They also can be used (with appropriate change of the probability distributions to reflect a risk premium) in computing the value of loan portfolio derivatives, such as credit default obligations. Practical financial engineering implementations in these contexts should, however, involve more detailed analyses to include differences from the simplifying assumptions for (35) and (36). These extensions can, for example, employ Monte Carlo simulation to capture the characteristics of each loan and their relationships.

In this model, the idiosyncratic components and default correspond to a first passage time of the asset below some level, as in the model of default of Merton (1974), but perhaps not exactly equal to the sum of obligations. Extensions of this model form the basis for the methodologies used by Moody's KMV and RiskMetrics Group in their commercial applications.

Concluding Remarks

Methods from operations research are widely applied in financial engineering, with the main areas overviewed here. The topics described illustrate how different techniques from stochastic modeling, optimization, dynamic systems, control, and reliability all contribute to financial engineering practice.

See

- Approximate Dynamic Programming
- Bellman Optimality Equation
- Conditional Value-at-Risk (CVaR)
- Dynamic Programming
- Financial Markets
- Markov Decision Processes
- Portfolio Theory: Mean-Variance Model
- Risk Management for Software Engineering
- Simulation of Stochastic Discrete-Event Systems

References

Almgren, R., & Chriss, N. (2000). Optimal execution of portfolio transactions. *Journal of Risk, 3*, 5–39.

Arrow, K. J., & Debreu, G. (1954). Existence of an equilibrium for a competitive economy. *Econometrica, 22*, 265–290.

Bachelier, L. (1900). Théorie de la spéculation. *Annales Scientifiques de l'École Normale Supérieure, 3*(17), 21–86.

Barlow, R., & Proschan, F. (1975). *Statistical theory of reliability and life–testing*. New York: Holt, Rinehart, and Winston.

Bertsimas, D., & Lo, A. W. (1998). Optimal control of execution costs. *Journal of Financial Markets, 1*, 1–50.

Bielecki, T., & Rutkowski, M. (2004). *Credit risk: Modeling, valuation, and hedging*. Berlin/Heidelberg: Springer–Verlag.

Birge, J.R. (2008). Optimization methods in dynamic portfolio management, Chapter 20. In J. R Birge, & V. Linetsky (Eds.). *Handbook in operations research and management science. Vol. 15., Financial engineering*. Amsterdam: Elsevier.

Black, F., & Litterman, R. (1992). Global portfolio optimization. *Financial Analysts Journal*. September 1992. 28–43.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy, 81*, 637–654.

Carr, P., Geman, H., Madan, D., & Yor, M. (2003). Stochastic volatility for lévy processes. *Mathematical Finance, 13*, 345–382.

Constantinides, G. M. (1982). Intertemporal asset pricing with heterogeneous consumers and without demand aggregation. *Journal of Business, 55*, 253–267.

Cont, R., & Tankov, P. (2004). *Financial modelling with jump processes*. London: Chapman & Hall/CRC Press.

Cornuéjols, G., & Tütüncü, R. (2007). *Optimization methods in finance*. Cambridge University Press.

Craddock, M., Heath, D., & Platen, E. (2000). Numerical inversion of Laplace transforms: A survey of techniques with applications to derivative pricing. *Journal of Computational Finance, 4*, 57–81.

D'Amico, G., Janssen, J., & Manca, R. (2005). Homogeneous semi–Markov reliability models for credit risk management. *Decisions in Economics and Finance, 28*, 79–93.

Derman, E., & Taleb, N. N. (2005). The illusions of dynamic replication. *Quantitative Finance, 5*, 323–326.

Duffie, D. (2003). Intertemporal asset pricing theory, Chapter 11. In G. M. Constantitinides, M. Harris, & R. M. Stulz (Eds.). *Handbook of the economics of finance. Vol. 1B., Financial markets and asset pricing*. Amsterdam: Elsevier.

Duffie, D., & Singleton, K. (2003). *Credit risk: Pricing, measurement, and management*. Princeton, NJ: Princeton University Press.

Fabozzi, F., Kolm, P., Pachamanova, D., & Focardi, S. (2007). *Robust portfolio optimization and management*. New York: Wiley.

Fama, E. F. (1970). Multiperiod consumption–investment decisions. *American Economic Review, 60*, 163–174.

Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics, 33*, 3–56.

Fama, E. F., & MacBeth, J. D. (1973). Risk, return, and equilibirium: Empirical tests. *Journal of Political Economy, 91*, 607–636.

Feng, L., Kovalov, P., Linetsky, V., & Marcozzi, M. (2008). Variational methods in derivative pricing, Chapter 7. In J. R. Birge, & V. Linetsky (Eds.). *Handbook in operations research and management science. Vol. 15., Financial engineering*. Amsterdam: Elsevier.

Feng, L., & Linetsky, V. (2008). Pricing discretely monitored barrier options and defaultable bonds in lévy process models: A fast Hilbert transform approach. *Mathematical Finance, 18*, 337–384.

Ferson, W. E. (2003). Tests of multifactor pricing models, volatility bounds and portfolio performance, 2003. Chapter 12. In G. M. Constantitinides, M. Harris, & R. M. Stulz (Eds.). *Handbook of the economics of finance. Vol 1B., Financial markets and asset pricing*. Amsterdam: Elsevier.

Glasserman, P. (2004). *Monte Carlo methods in financial engineering*. New York: Springer.

Harrison, J. M., & Pliska, S. R. (1981). Martingales and stochastic integrals in the theory of continuous trading. *Stochastic Processes and their Applications, 11*, 215–260.

Hull, J. C. (2011). *Options, futures, and other derivatives* (8th ed.). Englewood Cliffs, NJ: Prentice Hall.

Kan, R., & Zhou, G. (2007). Optimal portfolio choice with parameter uncertainty. *Journal of Financial and Quantitative Analysis, 42*, 621–656.

Kou, S. G. (2002). A jump–diffusion model for option pricing. *Management Science, 48*, 1086–1101.

Kou, S.G. (2008a). Jump–diffussion models for asset pricing in financial engineering, Chapter 2. In J. R. Birge, & V. Linetsky (Eds.). *Handbook in operations research and management science. Vol. 15., Financial engineering*. Amsterdam: Elsevier.

Kou, S.G. (2008b). Discrete barrier and lookback options, Chapter 8. In J. R. Birge, & V. Linetsky (Eds.). *Handbook in operations research and management science. Vol. 15., Financial engineering*. Amsterdam: Elsevier.

Kou, S. G., & Wang, H. (2004). Option pricing under a double exponential jump–diffusion model. *Management Science, 50*, 1178–1192.

Linetsky, V. (2004). Spectral expansions for Asian (average price) options. *Operations Research, 52*, 856–867.

Linetsky, V. (2008). Spectral methods in derivatives pricing, Chapter 6. In J. R. Birge, & V. Linetsky (Eds.). *Handbook in operations research and management science. Vol. 15., Financial engineering*. Elsevier, Amsterdam.

Lintner, J. (1965). The valuation of risk assets and the selection of risky investment in stock portfolios and capital budgets. *The Review of Economics and Statistics, 47*, 13–37.

Markowitz, H. M. (1952). Portfolio selection. *Journal of Finance, 7*, 77–91.

McNeil, A. J., Frey, R., & Embrechts, P. (2005). *Quantitative risk management*. Princeton, NJ: Princeton University Press.

Merton, R. C. (1973a). An intertemporal capital asset pricing model. *Econometrica, 41*, 867–887.

Merton, R. C. (1973b). Theory of rational option pricing. *Bell Journal of Economics and Management Science, 4*, 141–183.

Merton, R. C. (1974). On the pricing of corporate debt: The risk structure of interest rates. *Journal of Finance, 29*, 448–470.

Ross, S. A. (1976). The arbitrage theory of capital pricing. *Journal of Economic Theory, 13*, 341–360.

Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *Journal of Finance, 19*, 425–442.

Staum, J. (2008). Incomplete markets, Chapter 12. In J. R. Birge, & V. Linetsky (Eds.). *Handbook in operations research and management science. Vol. 15., Financial engineering*. Amsterdam: Elsevier.

Tavella, D., & Randall, C. (2000). *Pricing financial instruments: The finite difference method*. New York: Wiley.

Vašìček, O. (1987). *Probability of loss on loan portfolio*. San Francisco, CA: KMV Corporation.

Wu, L.W. (2008). Modeling financial security returns using Lévy processes, Chapter 3. In J. R. Birge, & V. Linetsky (Eds.). *Handbook in operations research and management science. Vol. 15., Financial engineering*. Amsterdam: Elsevier.

Financial Markets

John L. G. Board[1], Charles M. S. Sutcliffe[2] and William T. Ziemba[3,4]
[1]Henley Business School, University of Reading, Reading, UK
[2]University of Reading, Reading, UK
[3]University of British Columbia, Vancouver, British Columbia, Canada
[4]Oxford University, Oxford, UK

Introduction

Over the last half-century, a strong relationship between operations research (OR) and finance has

developed, resulting in a large and rapidly growing literature. Although most applications have been of OR techniques to finance, finance problems have also stimulated the development and refinement of OR techniques.

Finance problems, and especially those relating to financial markets, are particularly well suited to analysis using OR techniques. These problems are generally separable and well defined, have a clear objective (often to maximize profit or minimize risk), and have variables which are quantified in monetary terms. The relationships between the variables in finance models are usually stable and well defined, so that the resulting OR model is a good representation of the problem. As there are few concerns about human behavior ruling out the implementation of some solutions, the solutions produced by the analysis can usually be implemented. In addition, large amounts of data, both historic and real-time, are readily available and can be used in OR models. Some finance problems involve very large sums of money, so that even a very small improvement in the quality of the solution is profitable to implement.

This review describes the application of OR to problems in the analysis of financial markets (e.g. the markets for debt, equity and foreign exchange markets and the corresponding derivatives markets). A more extensive analysis of OR in financial markets appears in Board, Sutcliffe and Ziemba (2003). For a review of the application of OR to other areas of finance, such as: the management of the firm's finances, working capital management, capital investment, multinational taxation, and financial planning models (such as those developed for banks), see Ashford, Berry and Dyson (1988).

Portfolio Theory

A seminal application of OR techniques to finance was by Harry Markowitz (1952, 1987) when he specified the portfolio problem in terms of optimization over the assumed known means, variances and co-variances of the assets available, and proposed the solution of this problem through quadratic programming. In addition to specifying the portfolio problem in terms of OR techniques, Markowitz also developed solution algorithms for more general quadratic programming problems. This provides an example of the interaction between OR techniques and finance, with the former sometimes being adapted to meet the needs of the latter.

The most obvious application of portfolio theory is in choosing efficient equity portfolios, and empirical papers (e.g., Board and Sutcliffe 1994; Perold 1984 have used quadratic programming for this problem). The technique can also be applied more widely to selecting portfolios of currencies, bonds, or commercial loans. Multi-period portfolio problems have been specified as dynamic programming problems (Elton and Gruber 1971). Mulvey and Vladimirou (1992) used a stochastic generalized network model, and stochastic programming models have become widely used (Ziemba 2003, 2010).

OR researchers have also modified or replaced the quadratic programming approach to portfolio problems, often by explicitly specifying the relevant utility function and using stochastic linear programming with recourse to model risk in a multi-period framework. For example, Bradley and Crane (1972) proposed forming bond portfolios to maximize their expected value using stochastic linear programming to allow for interest rate risk. The scenarios included in portfolio models may be generated by Monte Carlo simulation, prior to the use of stochastic programming to maximize expected utility, e.g., (Golub et al. 1995; Zenios 1991, 1993b; Vassiadou-Zeniou and Zenios 1996; Zenios et al. 1998), who applied this approach to form portfolios of mortgage backed securities.

The investment policy of a pension fund can be formulated using asset-liability management (ALM) models that allow for the correlations between the values of the fund's assets and liabilities. While these problems can be formulated using quadratic programming (Board and Sutcliffe 2005), they have usually been solved in other ways (Ziemba and Mulvey 1998; Wallace and Ziemba 2005). For example, Mulvey (1994) assumed that the objective was to maximize the expected value of a nonlinear utility of wealth function, and specified the problem as a nonlinear network problem, with the simulation of future pension fund liabilities. Mulvey et al. (2008) used multi-period stochastic programming to determine investment policy for a defined benefit pension scheme. Similar asset-liability problems are also faced by insurance companies, for example Cariño et al. (1994, 1998a, b) formulated this

problem for a Japanese insurance company. Klaassen (1998) pointed out that the use of Monte Carlo simulation can bias the results by including arbitrage opportunities in the sampled scenarios. To avoid this, he aggregated an arbitrage-free event tree before its inclusion in a multi-stage stochastic programming model of the asset-liability problem. Surveys of implemented ALM models and their theory are in Ziemba (2003); Wallace and Ziemba (2005); Zenios and Ziemba (2006, 2007).

Another application of quadratic programming is generalized hedging in which the objective is usually to minimize the variance of a portfolio of a given set of assets and the chosen hedging instruments. If the hedging instruments include options, this introduces a nonlinearity into the hedging decision, and Murtagh (1989) devised a nonlinear programming model to hedge foreign currency exposure using a mixture of currency forward and options contracts. Similarly, quadratic programming has been used to construct index tracking portfolios, where the purpose is to select a portfolio of assets (e.g. equities or bonds) which, when combined with a matching short position in the index to be tracked, has minimum risk (Meade and Salkin 1989, 1990; Rudd 1980; Seix and Akhoury 1986). Multi-stage stochastic programming with recourse, in conjunction with Monte Carlo simulation to generate the scenarios, has been used by Vassiadou-Zeniou and Zenios (1996) and Zenios et al. (1998) to track an index of mortgage backed securities; also see Zenios and Ziemba (2007).

A related problem is that of portfolio immunization in which the objective is to construct a portfolio of interest rate dependent securities whose value is the same as some target asset (usually another interest rate dependent asset). There is also a literature on managing the assets and liabilities held by banks (which are taken to exclude equities), where the objective is usually to maximize the value (or expected value) of the portfolio over one (or many) time periods (net of penalty costs from constraint target violations), subject to restrictions of the total investment, maximum capital loss, and various bank regulations. By matching the duration of the portfolio with that of the target asset, the portfolio is immunized against small parallel shifts in the yield curve (the yield curve shows the interest rates for different maturities), see Fong and Vasicek, (1983); Kornbluth and Salkin, (1987); Nawalkha and Chambers, (1996); Alexander and Resnick, (1985). These immunization studies use a risk measure which does not involve squares or cross products of the decision variables, and so linear programming, not quadratic programming, is the solution technique.

In some applications of portfolio theory, the decision variables must be integer. Peterson and Leuthold (1987) and Shanker (1993) used quadratic-integer programming to compute hedging strategies involving futures.

Some authors have argued that formulation and solving quadratic-programming portfolio problems is too onerous, and proposed simplified solution techniques. Sharpe (1963) proposed a single index model which can be solved by the use of special purpose quadratic-programming algorithms. When each asset represents only a small proportion of the portfolio, Sharpe (1967) showed that his single index model can be treated as having a linear objective function. In 1971, Sharpe suggested using a piecewise linear approximation to the quadratic objective function, enabling the application of linear programming to solve portfolio problems. Another proposal is to minimize the mean absolute deviation (MAD), which can be solved using linear programming, rather than quadratic programming (Konno and Yamazaki 1991, 1997; Yawitz et al. 1976; Zenios and Kang 1993; Worzel et al. 1994). Another approach is to specify the problem as choosing between a range of prespecified equity portfolios using data envelopment analysis (Premachandra et al. 1998). A further approach is to reformulate the portfolio problem as a nonlinear generalized network model for which efficient solution algorithms exist (Mulvey 1987).

Portfolio problems, with the twin objectives of maximizing returns and minimizing risk, can also be viewed as goal programming problems with two goals. Additional goals can be introduced, and a number of authors have solved portfolio problems using goal programming, among them Kumar, Philippatos and Ezzell (1978), Kumar and Philippatos (1979), and Lee and Lerro (1973). The stochastic programming literature uses one objective, usually expected wealth maximization, with targets for the other objectives. The non-attainment of targets then yields convex penalties in various periods for the goals. An implemented application of this approach to the Siemens Austria pension fund is Geyer and Ziemba (2008).

Pricing Derivatives

It is very important when trading in financial markets to have a good model for valuing the asset being traded, and OR techniques have made a substantial contribution in this area. Indeed, the very rapid growth of these markets is partly due to the application of OR techniques in pricing models.

In 1977, Boyle proposed the use of Monte Carlo simulation as an alternative to the binomial model for pricing options for which a closed form solution is not readily available. Monte Carlo simulation has the advantage over the binomial model that its convergence rate is independent of the number of state variables (e.g., the number of underlying asset prices and interest rates), while that of the binomial model is exponential in the number of state variables. Simulation is used to generate paths for the price of the underlying asset until maturity. The cash flows from the option for each path, weighted by their risk neutral probabilities (i.e., the probabilities which can be inferred from prices by assuming that investors are risk neutral), are discounted back to the present using the risk free rate, allowing the average present value across all the sample paths to be computed, thus yielding the current price of the option (Boyle, Broadie and Glasserman 1997). As well as generating option prices, Monte Carlo simulation can be used to compute the sensitivities of results to misspecification of model parameters, including the hedge ratio, which are essential for many trading strategies (Broadie and Glasserman 1996).

In the past, it was thought that Monte Carlo simulation could not be used to price American style options because no closed form solutions for their price exist. This was considered a major problem, as the majority of options are American style. Progress has being made, however, in developing Monte Carlo simulation techniques for pricing American style options (Broadie and Glasserman 1997; Grant et al. 1997). Options have also been priced using finite difference approximations, and Dempster and Hutton (1996) and Dempster, Hutton and Richards (1998) have proposed using linear programming to solve the finite difference approximations to the price of American style put options. In addition, American style options can be priced using dynamic programming, Dixit and Pindyck (1994).

Provided a price history is available, a neural network can be trained to produce prices using a specified set of inputs, which can then be used for the out-of-sample pricing (Hutchinson et al. 1994; Bennell and Sutcliffe 2004) of securities.

Mortgage backed securities (MBS) are created by the securitization of a pool of mortgages. For any specific mortgage, the borrower has the right to repay the loan early (the prepayment option), or may default on the payments of capital and interest. Thus, MBS are hybrid securities, as they are variable interest rate securities with an early exercise option. Monte Carlo simulation can be used to generate interest rate paths for future years. Forecasts of the mortgage prepayment rates then permit the computation of the cash flows from each interest rate path, and these sequences of cash flows are used to value the MBS (Zenios 1993a; Ben-Dov et al. 1992; Boyle 1989). This procedure, which can be used to identify mispriced MBS in real time, is computationally demanding and parallel (and massively parallel) and distributed processing have been used in the solution of the problem. Simulation has also been used to price collateralized mortgage obligations or CMOs (Paskov 1997). Other hybrid securities, such as callable and putable bonds and convertible bonds face similar valuation problems to MBS and require similarly intensive solution methods.

There is an active secondary market in loan portfolios which may carry a significant default risk. Del Angel et al. (1998) used a Markov chain analysis with 14 loan performance states and Monte Carlo simulation to generate the probability distribution of the present value of loan portfolios.

Trading Tactics

As well as accurately pricing financial securities, traders are interested in finding imperfections in financial markets which can be exploited to make profits. One aspect of this is the search for weak form inefficiency (i.e., that an asset's past prices can be used as the basis of a profitable trading rule). Among the early attempts to find such exploitable regularities in stock prices were use of Markov chains (Dryden 1968, 1969).

Arbitrageurs seek to exploit small price discrepancies to give riskless profits, and network models have been used to find arbitrage opportunities between sets of currencies (Christofides et al. 1979; Kornbluth and Salkin 1987; Mulvey 1987;

Mulvey and Vladimirou 1992). This problem can be specified as a maximal flow network, where the aim is to maximize the flow of funds out of the network, or as a shortest path network.

OR techniques have been widely used by hedge funds to devise trading strategies. For example, Shaw, Thorp and Ziemba (1995) show how to construct and implement risk arbitrage in the Japanese warrant market, while Mulvey, Ural, and Zhang (2007) examine overlay strategies.

There has been a growing interest in using artificial intelligence based techniques (expert systems, neural networks, genetic algorithms, fuzzy logic, and inductive learning) to develop trading strategies for financial markets (Trippi and Turban 1993; Refenes 1995; Goonatilake and Treleaven 1995; Wong and Selvi 1998). Such approaches have the advantage that they can pick up nonlinear dynamics and require little prior specification of the relationships involved.

Funding Decisions

OR techniques have also been used to help firms determine the most appropriate method by which to raise capital from the financial markets. Brick, Mellon, Surkis and Mohl (1983) put forward a chance-constrained linear programming model to compute the values of the debt-equity ratio each period that maximize the value of the firm. Other studies have specified the choice between various types of funding as a linear goal programming problem (Hong 1981; Lee and Eom 1989).

A different approach to the debt problem is to assume that the firm has found its desired debt-equity ratio and is purely concerned with raising the requisite debt as cheaply as possible. In this case, debt can be treated like any other input to the productive process, and inventory models used to determine the optimal reorder times and quantities (Bierman 1966; Litzenberger and Rutenberg 1972).

The design of callable bonds has been addressed by Consiglio and Zenios (1997a, b), who used nonlinear programming, while Holmer, Yang and Zenios (1998) used a simulated annealing algorithm. Firms which have issued callable debt face the bond-scheduling problem in which they must decide when to call (repay) the existing debt and refinance it with a new issue, presumably at a lower cost. This is a dynamic programming problem and has been modeled as such by Weingartner (1967), Elton and Gruber (1971) and Kraus (1973).

Finally, the problem facing borrowers of choosing between alternative mortgage contracts (e.g., fixed rate, variable rate and adjustable rate mortgages) has been modeled using decision trees (Heian and Gale 1988; Luna and Reid 1986).

Strategic Problems

In recent years, some of the decisions facing traders and market makers in financial markets have been analyzed using game theory (O'Hara 1995; Dutta and Madhavan 1997). Traders in stock markets seek to trade at the most attractive prices, and large trades are often broken up into a sequence of smaller trades in an effort to minimize the price impact. This can be viewed as a strategic problem, and Bertsimas and Lo (1998) used stochastic dynamic programming to compute an optimal trading strategy.

Powers (1987) applied game theory to the situation where a company has two major shareholders, and a large number of very small shareholders. This can be modeled as an oceanic game, in which the two large players behave strategically while the many small shareholders (the ocean) do not. This approach can be used to derive the highest price a large shareholder will pay in the market for corporate control.

Regulatory and Legal Problems

Financial regulators have become increasingly concerned about financial markets with their very large and rapid international financial flows. OR techniques have proved useful in regulating the capital reserves held by banks and other financial institutions to cover their risk exposure. OR techniques have also been used to ensure compliance with various legal requirements by designing appropriate strategies, and to solve other legal problems relating to financial markets.

A key regulatory issue is determining the capital required by financial institutions to underpin their activities in financial markets. An increasingly popular approach to this problem is the value at risk (VaR), which involves quantification of the lower tail

of the probability distribution of outcomes from the firm's portfolio. Portfolios usually include options (or financial securities with option-like characteristics), and these have highly asymmetric payoffs. For such securities, analytical solutions to finding the probabilities in the lower tail of the pay off distribution are unreliable. Riskmetrics™ uses approximations of the probability tail behavior for options that are at or near the money (an option is at-the-money when the current price of the underlying asset is close to the price at which the option can be exercised), and Monte Carlo simulation for other options positions (Morgan and Reuters 1996). A related application of Monte Carlo simulation is stress testing, which quantifies the sensitivity of a portfolio to specified, often adverse market scenarios. Some securities are also subject to credit risk, which has a highly nonnormal distribution for all instruments. Therefore, Monte Carlo simulation is relevant to modeling the credit risk of portfolios of financial instruments (e.g., loans, letters of credit, bonds, trade credit, swaps, forwards) as in CreditMetrics™ (Morgan 1997).

Data envelopment analysis has been used to assist in bank regulation by measuring bank efficiency, which is then used to predict bank failure (Barr et al. 1993; Bauer et al. 1998).

Traders are required to put up margin when they trade options, and Rudd and Schroeder (1982) have developed a linear programming model in which the problem was modeled as a transportation problem.

An extensive set of rules governs the way in which a to-be-announced MBS can be structured, leading to a complex problem in devising a feasible solution. This can be specified as a complicated integer programming problem (with the objective of maximizing the originator's profit). Collateralized mortgage obligations (CMOs) also involve the securitization of a mortgage pool, but in this case the pool is structured into a series of bonds (or tranches), each with a different maturity and risks. Dahl, Meeraus and Zenios (1993) have proposed a complex zero–one programming model for solving this problem, with the objective of maximizing the proceeds from the issue.

Sharda (1987) proposed a linear programming formulation to establish the maximum loss that investors could have sustained from trading in a company's shares. This figure can then be used by the company's lawyers when fighting a lawsuit claiming damages from a misleading statement by the company.

In August 1982, the Kuwait Stock Market collapsed leaving $94 billion of debt to be resolved. This led to the problem of devising a fair method for distributing the assets seized from insolvent brokers among the other brokers and private investors. This problem was solved using linear programming, which reduced the total unresolved debt to $20 billion, saving an estimated $10.34 billion in lawyer's fees (Taha 1991; Elimam et al. 1996, 1997).

Economic Understanding

OR can help in trying to understand the economic forces shaping the finance sector. Using a linear programming model of a bank, Ben-Horim and Silber (1977) employed annual data to compute movements in the shadow prices of the various constraints. They suggested that a rise in the shadow price of the deposits constraint led to the financial innovation of negotiable CDs.

Arbitrage Pricing Theory (APT) seeks to identify the factors which affect asset returns. Most tests of the APT use factor analysis and have difficulty in determining the number and definition of the factors that influence asset returns. To overcome these problems, Ahmadi (1993) suggested using a neural network, which also has the advantage that the results are distribution free.

Concluding Remarks

Mathematical programming is the OR technique that has been most widely applied in financial markets. Most types of mathematical programming have been employed — linear, quadratic, nonlinear, integer, goal, chance constrained, stochastic, fractional, DEA and dynamic. Monte Carlo simulation is also widely used in financial markets — mainly to value exotic options and securities with embedded options, and to estimate the VaR for various financial institutions. In some cases the use of OR techniques has influenced the way financial markets function since they permit traders to make better decisions in less time. For example, exotic options would trade with much wider

bid-ask spreads, if they traded at all, in the absence of the accurate prices computed using Monte Carlo simulation.

Other OR techniques are less used in financial markets. Arbitrage and multi-period portfolio problems have been formulated as network models, while market efficiency has been tested using neural networks. Game theory has been applied to battles for corporate control, decision trees to analyze mortgage choice, inventory models to set the size and timing of corporate bond issues, and Markov chains to valuing loan portfolios and testing market efficiency. One important OR technique — queueing theory — has found little application in financial markets.

This review has shown that OR techniques have been usefully applied to portfolio problems and the accurate pricing of complex financial instruments. They are also used by financial regulators and financial institutions in setting capital adequacy standards.

See

- ▶ Data Envelopment Analysis
- ▶ Dynamic Programming
- ▶ Financial Engineering
- ▶ Goal Programming
- ▶ Linear Programming
- ▶ Nonlinear Programming
- ▶ Portfolio Theory: Mean-Variance Model
- ▶ Quadratic Programming
- ▶ Stochastic Programming
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Ahmadi, H. (1993). Testability of the arbitrage pricing theory by neural networks. In R. R. Trippi & E. Turban (Eds.), *Neural networks in finance and investing: Using artificial intelligence to improve real world performance* (pp. 421–432). Chicago: Probus Publishing.

Alexander, G. J., & Resnick, B. G. (1985). Using linear and goal programming to immunize bond portfolios. *Journal of Banking and Finance, 9*(1), 35–54.

Ashford, R. W., Berry, R. H., & Dyson, R. G. (1988). Operational research and financial management. *European Journal of Operations Research, 36*(2), 143–152.

Barr, R. S., Seiford, L. M., & Siems, T. F. (1993). An envelopment analysis approach to measuring the managerial efficiency of banks. *Annals of Operations Research, 45*(1–4), 1–19.

Bauer, P. W., Berger, A. N., Ferrier, G. D., & Humphrey, D. B. (1998). Consistency conditions for regulatory analysis of financial institutions: A comparison of frontier efficiency methods. *Journal of Economics and Business, 50*(2), 85–114.

Ben-Dov, Y., Hayre, L., & Pica, V. (1992). Mortgage valuation models at prudential securities. *Interfaces, 22*(1), 55–71.

Ben-Horim, M., & Silber, W. L. (1977). Financial innovation: A linear programming approach. *Journal of Banking and Finance, 1*(3), 277–296.

Bennell, J., & Sutcliffe, C. M. S. (2004). Black-scholes versus artificial neural networks in pricing FTSE 100 options. *Intelligent Systems in Accounting, Finance and Management, 12*(4), 243–260.

Bertsimas, D., & Lo, A. W. (1998). Optimal control of execution costs. *Journal of Financial Markets, 1*(1), 1–50.

Bierman, H. (1966). The bond size decision. *Journal of Financial and Quantitative Analysis, 1*(4), 1–14.

Board, J. L. G., & Sutcliffe, C. M. S. (2005). Joined-up pensions policy in the UK: An asset-liability model for simultaneously determining the asset allocation and contribution rate. In S. A. Zenios & W. T. Ziemba (Eds.), *Handbook of asset and liability management*, *Handbooks in finance* (Vol. 2, pp. 1029–1067). North Holland/Elsevier Science B.V. 2007.

Board, J. L. G., & Sutcliffe, C. M. S. (1994). Estimation methods in portfolio selection and the effectiveness of short sales restrictions: UK evidence. *Management Science, 40*, 516–534.

Board, J. L. G., Sutcliffe, C. M. S., & Ziemba, W. T. (2003). Applying operations research techniques to financial markets. *Interfaces, 32*(2), 12–34.

Boyle, P. P. (1977). Options: A Monte Carlo approach. *Journal of Financial Economics, 4*(3), 323–338.

Boyle, P. P. (1989). Valuing Canadian mortgage backed securities. *Financial Analysts Journal, 45*(3), 55–60.

Boyle, P. P., Broadie, M., & Glasserman, P. (1997). Monte Carlo methods for security pricing. *Journal of Economic Dynamics and Control, 21*, 1267–1321.

Bradley, S. P., & Crane, D. B. (1972). A dynamic model for bond portfolio management. *Management Science, 19*, 139–151.

Brick, I. E., Mellon, W. G., Surkis, J., & Mohl, M. (1983). Optimal capital structure: A multi period programming model for use in financial planning. *Journal of Banking and Finance, 7*(1), 45–67.

Broadie, M., & Glasserman, P. (1996). Estimating security price derivatives using simulation. *Management Science, 42*, 269–285.

Broadie, M., & Glasserman, P. (1997). Pricing American style securities using simulation. *Journal of Economic Dynamics and Control, 21*, 1323–1352.

Cariño, D. R., Kent, T., Myers, D. H., Stacy, C., Sylvanus, M., Turner, A. L., Watanabe, K., & Ziemba, W. T. (1994). The Russell-Yasuda Kasai model: An asset-liability model for a Japanese insurance company using multistage stochastic programming. *Interfaces, 24*(1), 29–49. Reprinted in Ziemba and Mulvey (1998).

Cariño, D. R., Myers, D., & Ziemba, W. T. (1998). Concepts, technical issues and uses of the Russell Yasuda Kasai model. *Operations Research, 46*, 450–462.

Cariño, D. R., & Ziemba, W. T. (1998). Formulation of the Russell Yasuda Kasai financial planning model. *Operations Research, 46*, 433–449.

Christofides, N., Hewins, R. D., & Salkin, G. R. (1979). Graph theoretic approaches to foreign exchange. *Journal of Financial and Quantitative Analysis, 14*, 481–500.

Consiglio, A., & Zenios, S. A. (1997a). A model for designing callable bonds and its solution using tabu search. *Journal of Economic Dynamics and Control, 21*, 1445–1470.

Consiglio, A., & Zenios, S. A. (1997b). High performance computing for the computer aided design of financial products. In L. Grandinetti, J. Kowalik, & M. Vajtersic (Eds.), *Advances in high performance computing* (NATO Advanced Science Institute Series, Vol. 30, pp. 273–302). Dordrecht: Kluwer Academic Publishers.

Dahl, H., Meeraus, A., & Zenios, S. A. (1993). Some financial optimization models: II financial engineering. In S. A. Zenios (Ed.), *Financial optimization* (pp. 37–71). Cambridge: Cambridge University Press.

Del Angel, G. F., Márquez, A., & Patiño, E. P. (1998). A discrete Markov chain model for valuing loan portfolios. The case of Mexican loan sales. *Journal of Banking and Finance, 22*, 1457–1480.

Dempster, M. A. H., & Hutton, J. P. (1996, October). *Pricing American stock options by linear programming*. Working Paper, Department of Mathematics, University of Essex, 34 p.

Dempster, M. A. H., Hutton, J. P., & Richards, D. G. (1998, September). *LP valuation of exotic American options exploiting structure*. Working Paper, Judge Institute of Management Studies, University of Cambridge, WP 27/98.

Dixit, A. K., & Pindyck, R. S. (1994). *Investment under uncertainty*. Princeton, NJ: Princeton University Press.

Dryden, M. M. (1968). Short-term forecasting of share prices: An information theory approach. *Scottish Journal of Political Economy, 15*, 227–249.

Dryden, M. M. (1969). Share price movements: A Markovian approach. *Journal of Finance, 24*, 49–60.

Dutta, P. K., & Madhavan, A. (1997). Competition and collusion in dealer markets. *Journal of Finance, 52*, 245–276.

Elimam, A. A., Girgis, M., & Kotob, S. (1996). The use of linear programming in disentangling the bankruptcies of al-Manakh stock market crash. *Operations Research, 44*, 665–676.

Elimam, A. A., Girgis, M., & Kotob, S. (1997). A solution to post crash debt entanglements in Kuwait's al-Manakh stock market. *Interfaces, 27*(1), 89–106.

Elton, E. J., & Gruber, M. J. (1971). Dynamic programming applications in finance. *Journal of Finance, 26*, 473–506.

Fong, H. G., & Vasicek, O. (1983). The tradeoff between return and risk in immunized portfolios. *Financial Analysts Journal, 39*(5), 73–78.

Geyer, A., & Ziemba, W. T. (2008). The innovest Austrian pension fund planning model INNOALM. *Operations Research, 56*(4), 797–810.

Golub, B., Holmer, M., McKendall, R., Pohlman, L., & Zenios, S. A. (1995). A stochastic programming model for money management. *European Journal of Operational Research, 85*, 282–296.

Goonatilake, S., & Treleaven, P. (Eds.). (1995). *Intelligent systems for finance and business*. New York: John Wiley.

Grant, D., Vora, G., & Weeks, D. (1997). Path dependent options: Extending the Monte Carlo simulation approach. *Management Science, 43*, 1589–1602.

Heian, B. C., & Gale, J. R. (1988). Mortgage selection using a decision tree approach: An extension. *Interfaces, 18*(4), 72–81.

Holmer, M. R., Yang, D., & Zenios, S. A. (1998). Designing callable bonds using simulated annealing. In C. Zopounidis (Ed.), *Operational tools in the management of financial risks* (pp. 177–196). Kluwer Academic Publishers.

Hong, H. K. (1981). Finance mix and capital structure. *Journal of Business Finance and Accounting, 8*, 485–491.

Hutchinson, J. M., Lo, A. W., & Poggio, T. (1994). A non-parametric approach to pricing and hedging derivative securities via learning networks. *Journal of Finance, 49*, 851–889.

Klaassen, P. (1998). Financial asset-pricing theory and stochastic programming models for asset/liability management: A synthesis. *Management Science, 44*, 31–48.

Konno, H., & Yamazaki, H. (1991). Mean absolute deviation portfolio optimization model and its applications to Tokyo stock market. *Management Science, 37*, 519–531.

Konno, H., & Yamazaki, H. (1997). An integrated stock–bond portfolio optimization model. *Journal of Economic Dynamics and Control, 21*, 1427–1444.

Kornbluth, J. S. H., & Salkin, G. R. (1987). *The management of corporate financial assets: Applications of mathematical programming models*. London: Academic Press.

Kraus, A. (1973). The bond refunding decision in an efficient market. *Journal of Financial and Quantitative Analysis, 8*, 793–806.

Kumar, P. C., & Philippatos, G. C. (1979). Conflict resolution in investment decisions: Implementation of goal programming methodology for dual purpose funds. *Decision Sciences, 10*, 562–576.

Kumar, P. C., Philippatos, G. C., & Ezzell, J. R. (1978). Goal programming and the selection of portfolios by dual purpose funds. *Journal of Finance, 33*, 303–310.

Lee, S. M., & Eom, H. B. (1989). A multi criteria approach to formulating international project financing strategies. *Journal of the Operational Research Society, 40*, 519–528.

Lee, S. M., & Lerro, A. J. (1973). Optimizing the portfolio selection for mutual funds. *Journal of Finance, 28*, 1087–1101.

Litzenberger, R. H., & Rutenberg, D. P. (1972). Size and timing of corporate bond flotations. *Journal of Financial and Quantitative Analysis, 7*, 1343–1359.

Luna, R. E., & Reid, R. A. (1986). Mortgage selection using a decision tree approach. *Interfaces, 16*(3), 73–81.

Markowitz, H. M. (1952). Portfolio selection. *Journal of Finance, 7*, 77–91.

Markowitz, H. M. (1987). *Mean-variance in portfolio choice and capital markets*. London: Blackwell.

Meade, N., & Salkin, G. R. (1989). Index funds — construction and performance measurement. *Journal of the Operational Research Society, 40*, 871–879.

Meade, N., & Salkin, G. R. (1990). Developing and maintaining an equity index fund. *Journal of the Operational Research Society, 41*, 599–607.

Morgan, J. P. (1997). CreditMetrics™ — technical document, Morgan Guarantee Trust Company of New York.

Morgan, J. P., & Reuters. (1996). RiskMetrics™ — technical document, Morgan Guarantee Trust Company of New York.

Mulvey, J. M. (1987). Nonlinear network models in finance. In K. D. Lawrence, J. B. Guerard, & G. R. Reeves (Eds.), *Advances in mathematical programming and financial planning* (Vol. 1, pp. 253–271). Greenwich, CT: JAI Press.

Mulvey, J. M. (1994). An asset liability investment system. *Interfaces, 24*(3), 22–33.

Mulvey, J. M., Simsek, K. D., Zhang, Z., Fabozzi, F. J., & Pauling, W. R. (2008). Assisting defined-benefit pension plans. *Operations Research, 56*(5), 1066–1078.

Mulvey, J. M., Ural, C., & Zhang, Z. (2007). Improving performance for long-term investors: Wide diversification, leverage and overlay strategies. *Quantitative Finance, 7*(2), 175–187.

Mulvey, J. M., & Vladimirou, H. (1992). Stochastic network programming for financial planning problems. *Management Science, 38*, 1642–1664.

Murtagh, B. A. (1989). Optimal use of currency options. *Omega, 17*, 189–192.

Nawalkha, S. K., & Chambers, D. R. (1996). An improved immunization strategy: M-absolute. *Financial Analysts Journal, 52*(5), 69–76.

O'Hara, M. (1995). *Market microstructure theory*. London: Blackwell.

Paskov, S. H. (1997). New methodologies for valuing derivatives. In M. A. H. Dempster & S. R. Pliska (Eds.), *Mathematics of derivative securities* (pp. 545–582). Cambridge: Cambridge University Press.

Perold, A. F. (1984). Large scale portfolio optimization. *Management Science, 30*, 1143–1160.

Peterson, P. E., & Leuthold, R. M. (1987). A portfolio approach to optimal hedging for a commercial cattle feedlot. *Journal of Futures Markets, 7*, 443–457.

Powers, I. Y. (1987). A game theoretic model of corporate takeovers by major stockholders. *Management Science, 33*, 467–483.

Premachandra, I., Powell, J. G., & Shi, J. (1998). Measuring the relative efficiency of fund management strategies in New Zealand using a spreadsheet-based stochastic data envelopment analysis model. *Omega, 26*, 319–331.

Refenes, A. P. (Ed.). (1995). *Neural networks in the capital markets*. New York: John Wiley.

Rudd, A. (1980). Optimal selection of passive portfolios. *Financial Management, 9*(1), 57–66.

Rudd, A., & Schroeder, M. (1982). The calculation of minimum margin. *Management Science, 28*, 1368–1379.

Seix, C., & Akhoury, R. (1986). Bond indexation: The optimal quantitative approach. *Journal of Portfolio Management, 12*(3), 50–53.

Shanker, L. (1993). Optimal hedging under indivisible choices. *Journal of Futures Markets, 13*, 237–259.

Sharda, R. (1987). A simple model to estimate bounds on total market gains and losses for a particular stock. *Interfaces, 17*(5), 43–50.

Sharpe, W. F. (1963). A simplified model for portfolio analysis. *Management Science, 9*, 277–293.

Sharpe, W. F. (1967). A linear programming algorithm for mutual fund portfolio selection. *Management Science, 13*, 499–510.

Sharpe, W. F. (1971). A linear programming approximation for the general portfolio analysis problem. *Journal of Financial and Quantitative Analysis, 6*, 1263–1275.

Shaw, J., Thorp, E. O., & Ziemba, W. T. (1995). Convergence to efficiency of the Nikkei put warrant market of 1989-90. *Applied Mathematical Finance, 2*, 243–271.

Taha, H. A. (1991). Operations research analysis of a stock market problem. *Computers and Operations Research, 18*, 597–602.

Trippi, R. R., & Turban, E. (Eds.). (1993). *Neural networks in finance and investing: Using artificial intelligence to improve real world performance*. Chicago: Probus Publishing.

Vassiadou-Zeniou, C., & Zenios, S. A. (1996). Robust optimization models for managing callable bond portfolios. *European Journal of Operational Research, 91*, 264–273.

Wallace, S. W., & Ziemba, W. T. (Eds.). (2005). *Applications of stochastic programming*, SIAM-MPS.

Weingartner, H. M. (1967). Optimal timing of bond refunding. *Management Science, 13*, 511–524.

Wong, K., & Selvi, Y. (1998). Neural network applications in finance: A review and analysis of literature (1990–1996). *Information Management, 34*(3), 129–139.

Worzel, K. J., Vassiadou-Zeniou, C., & Zenios, S. A. (1994). Integrated simulation and optimization models for tracking indices of fixed income securities. *Management Science, 42*, 223–233.

Yawitz, J. B., Hempel, G. H., & Marshall, W. J. (1976). A risk-return approach to the selection of optimal government bond portfolios. *Financial Management, 5*(3), 36–45.

Zenios, S. A. (1991). Massively parallel computations for financial planning under uncertainty. In J. P. Mesirov (Ed.), *Very large scale computation in the 21st century* (pp. 273–294). Philadelphia: Society for Industrial and Applied Mathematics.

Zenios, S. A. (1993a). Parallel Monte Carlo simulation of mortgage backed securities. In S. A. Zenios (Ed.), *Financial optimization* (pp. 325–343). Cambridge: Cambridge University Press.

Zenios, S. A. (1993b). A model for portfolio management with mortgage backed securities. *Annals of Operations Research, 43*, 337–356.

Zenios, S. A., & Ziemba, W. T. (Eds.). (2006). *Handbook of asset liability modelling*: *Vol. 1. Theory and methodology*. North Holland.

Zenios, S. A., & Ziemba, W. T. (Eds.). (2007). *Handbook of asset liability modelling: Vol. 2. Applications and case studies*. North Holland.

Zenios, S. A., Holmer, M. R., McKendall, R., & Vassiadou-Zeniou, C. (1998). Dynamic models for fixed income portfolio management under uncertainty. *Journal of Economic Dynamics and Control, 22*, 1517–1541.

Zenios, S. A., & Kang, P. (1993). Mean absolute deviation portfolio optimization for mortgage backed securities. *Annals of Operations Research, 45*, 433–450.

Ziemba, W. T. (2003). *The stochastic programming approach to asset liability and wealth management*. Charlottesville, VA: AIMR.

Ziemba, W. T. (2010). Ideas in asset liability management in the tradition of H.M. Markowitz. In J. Guerard. (Ed.), *Essays in honour of H.M. Markowitz*. Springer.

Ziemba, W. T., & Mulvey, J. M. (Eds.). (1998). *Worldwide asset and liability modelling*. Cambridge University Press.

Finite Source

When the potential number of customers who could use a queueing system is finite, as in models of machine repair.

See

► Queueing Theory

Fire Safety Modeling and Applications

John R. Hall Jr.[1] and John M. Watts Jr.[2]
[1]National Fire Protection Association, Quincy, MA, USA
[2]Fire Safety Institute, Middlebury, VT, USA

Introduction

Individuals and organizations make decisions where fire safety is an explicit or implicit objective. Much of the modeling available to support decisions where the objective is explicit is drawn from an OR/MS concept or is used within a larger OR/MS framework.

OR/MS professionals may find themselves working on problems and for decision makers where part of the relevant objective function involves a measure of fire loss, risk, or safety. When that happens, they may want to avail themselves of some of the fire safety models and calculation methods developed in the fields of fire protection engineering, fire safety engineering, and fire safety science, to predict or estimate measures of fire loss, risk or safety. Alternatively, the OR/MS professional may encounter such models and methods in the analytic work of others on their project team or competing project teams or advocates. For any of these reasons, those working on issues of fire safety will want to be familiar with fire safety models and applications.

Types of loss may involve fire damage to people or property – the most familiar and most common measures of interest – or may involve fire damage to continuity of mission, business or operations; to the environment; or to cultural heritage.

Among the tools in use, the term "fire model" is normally reserved for physics-based models of the spread of combustion products and other fire effects through a defined space. The term "fire safety science" is also normally used in this way. The OR/MS professional, like the fire protection engineer, may need other models and calculation methods to complete an analysis, including:

- Models of the effects of fire on specified targets, including the health of people (such as burn, heat stress, and toxic effects) and the value and functionality of objects (such as corrosion or degradation of structural strength and integrity).
- Models of the behavior of people and the usage of products, needed to produce timelines of locations of potential targets which will, when combined with timelines of fire conditions by location, produce a timeline of degree of fire exposure for such targets.

OR/MS professionals may be asked to frame the decision problem in terms of a fire risk assessment of each of several alternative courses of action. Some of the fire safety field's fire risk assessment methods are based on fire models and other physical models used within a scenario-based structure. Other methods are not so detailed.

This article provides the construct in which fire safety objectives are established and an overview of fire models and other physical models relevant to fire safety evaluation in terms of the threat and the potential consequences. Most of the material, however, is devoted to an overview of fire risk assessment methods.

Objective-Setting

A generic OR/MS formulation of a decision problem identifies

- a set of controllable variables,
- a set of uncontrollable variables,
- a set of outcomes with measures of their attractiveness, and
- a model that shows how controllable and uncontrollable variables combine to produce outcomes.

The analyst can then optimize the choice of controllable variables or use the model to select the best controllable variables from a limited set of available alternatives.

Some fire safety codes and standards have fairly recently added explicit statements of their goals and objectives. These include protection of people, property, mission, environment, and cultural heritage.

Life safety is the most commonly regulated fire safety objective. It is greatly focused on the occupants of a building, but may also include people who respond to fire emergencies and, in that capacity, may enter a burning building. Measures of effectiveness are typically in terms of expected mortality.

Property protection fire safety objectives may include the facility (structure), as well as its contents (processes, storage, etc.). In many cases, the property protection risk management objective will be related to limiting the spread of fire to a defined area and may be converted to an expected loss value. It is also common to first establish a monetary value for acceptable loss, and then determine the maximum tolerable fire size (i.e., the extent of fire and smoke spread and the potential suppression agent damage that would lead to the maximum tolerable loss).

Continuity of operations objectives typically reflect the maximum tolerable downtime of a process, building or facility due to fire. In many respects, objectives for continuity of operations will follow the analysis used for property loss, but the focus is on the cost associated with a return to operations. Establishing continuity of operations objectives requires the decision maker to determine to what extent the organization understands and values the process, building, facility, or concern.

Environmental protection objectives are typically related to air pollution, ground water contamination, ecosystem damage, and adverse health effects. They may address sustainability issues as well, including recycling and reuse of materials.

Cultural heritage resources represent intangible or non-economic values that may not be recoverable. At risk is the loss of resources such as architecture, artifacts, and art from fire or fire fighting, or the intrusion of fire safety systems on authenticity.

Controllable Variables, Outcomes, and Measures of Attractiveness

In a fire safety context, the controllable variables might be:

- the design characteristics of a building,
- the design characteristics of a burnable product,
- the design characteristics of an object that could provide the ignition heat for a fire,
- the design characteristics of a product that is required to contain, resist, or continue performing a function despite exposure to fire,
- the design characteristics of equipment or a system for fire detection or suppression, or
- the specifications for an educational or training program.

As for outcomes and measures of their attractiveness, it is likely that there will be a variety of interested parties affected by the choices, and they may all have a say in the specification of outcomes. The conventional OR decision maker – who might be the architect or builder of a building or the manufacturer of a new product – is often constrained by codes, standards, and regulations designed to limit the potential harm from an approved design.

In this decision-making environment, analysis is typically performed not as a multi-objective constrained optimization problem, but as an exercise in establishing equivalency; that is, demonstrating that the outcomes associated with the candidate design will be no worse than those that would have occurred with a fully code-compliant conventional design, even if the candidate design does not itself fully comply with the more prescriptive requirements of the code.

Fire Models and Formal Scenario-Based Analysis

It is almost never the case that one can identify, for a fire safety problem, a complete set of controllable and uncontrollable variables, each expressible on a quantitative scale and each linked to needed data. If one could do so, then it is even more unlikely that there would be a validated model linking all variables to each other and to the outcomes of interest. Some type of simplification is necessary.

This section discusses scenario-based approaches to analysis. The fire models described in this section are particularly relevant when the decision maker or the project team insist on the kind of detail and evidence of validity associated with fire protection engineering, which is to say, strong reliance on detailed mathematical models of physical phenomena supported by data taken primarily from laboratory testing of the properties of materials, products,

assemblies and systems. The next section discusses qualitative, semi-quantitative, and purely statistical/empirical methods.

Fire and Behavioral Scenarios

The universe of possible combinations of uncontrollable variables is, in the fire safety analysis world, the universe of possible fire and related scenarios. A fire scenario describes fire conditions as a function of spatial location and time, from the beginning of the fire to its conclusion. Related scenarios are used to provide parallel characterizations of the locations and conditions of potential targets of harm from the fire. Objects that could be subject to property damage, environmental damage, or cultural damage are likely to be fixed, which means the scenario need only describe locations and conditions at the start of the fire, knowing that those locations and conditions will not change during the fire. People who could be injured or killed by fire are a different matter, because they probably will change their locations during the fire. Hence, fire scenarios are likely to be used with behavioral scenarios to produce parallel timelines for fire conditions and conditions of potential victims.

Each scenario is a different kind of fire challenge to the design. A scenario structure identifies a finite number of scenarios, each of which represents many like scenarios and all of which collectively capture the universe. In a complete risk assessment, each representative scenario is weighted by the combined likelihood of the scenarios it represents. In a more typical engineering analysis, a small number of scenarios are considered to represent the most typical and the distinct types of most challenging scenarios. In either type of analysis, the same types of models are used to calculate the predicted consequences (the preferred term for the measure of attractiveness of the predicted outcomes).

Models of Use in Scenario Analysis

There are several comprehensive references that should be consulted for additional information on fire safety models for scenario-based analysis: SFPE Handbook, 2008; Engineering Guide: Fire Risk Assessment, 2006; SFPE Engineering Guide to Performance-Based Fire Protection, 2007; Olenick and Carpenter 2003; NFPA 551 2007.

Fire Models

Fire models predict the change in heat and mass (smoke) conditions over space and time. They do not predict fire initiation or fire growth but rather predict the spread of heat and combustion products. Such models adapt the more general models of fluid mechanics and heat transfer to the specific purpose of fire modeling. (See most of Section 1 in the SFPE Handbook).

Fire models differ greatly in granularity. The earliest fire models treated each room or compartment as a single zone, see Chapter 3–7, SFPE (2008). For several decades, popular so-called zone models used two-zone representations of each compartment, taking advantage of the fact that within a compartment, the most important and pronounced variation in fire conditions is between the upper layer, which fills with fire and smoke first, and the cleaner lower layer, with the two layers separated by a boundary that moves down over time.

Advances in computing speed and power have shifted usage to so-called computational fluid dynamics (CFD) or field models, see Chapter 3–8, SFPE (2008). CFD models represent a compartment by many small control volumes in a grid representation. The laws of conservation of mass, momentum, and energy are used to predict fire conditions in successive time intervals. When zone models were popular, CFD models were commonly limited to representations in thousands of control volumes, while today millions of control volumes are often used in analyses that cost less and take less time than the coarser-grid analyses of the past.

The more sophisticated the fire model is, the more extensive are its associated data needs, including quantities and burning properties of potential fuel sources in the fire area and the spatial dimensions and fire properties of the boundaries of compartments. While there has been extensive verification and validation work on CFD models, with associated development of rules of good practice, there has been far less work on the sensitivity of results to variations in estimated data values for the many uncontrollable variables used by the fire models that must use estimates because of the absence of well-established sources of data.

There is also a tradition of stochastic fire models – notably Markov process (or state transition) models and network model, see Chapter 3–14,

SFPE (2008). Such models do not produce the kind of detailed descriptions of fire conditions by place and time that are produced by zone models, let alone CFD models, but they can provide useful and valid predictions when the fire safety problem does not require such details. For example, it may be quite sufficient for evaluation of a building design to know which rooms and how many rooms were fire involved, without knowing the heat and smoke conditions within any room.

Network models have a notable advantage over CFD and zone models in the area of phase change events. The likelihood and time delay in burning through a barrier, such as a door or wall, is more easily modeled using a network model. The same can be true for window breaking, a random event that produces a qualitative change in ventilation for the fire, and for the deformation or collapse of a load-bearing element.

Traditional fire risk assessment models using event trees can be modified to treat the spread of combustion products as simply a set of events.

Modeling Fire Consequences for Scenario Analysis

Egress. There are a number of models available to predict the movement of people in response to fire cues, see Chapters 3-11, 12, 13 and 17, SFPE (2008). Early models simplified the process in either of two ways. Hydraulic-style models moved people through corridors as if they were ball bearings in a tube. These models tend to predict egress times far shorter than those that are observed in practice. The other approach was to construct a network model with a number of embedded behavioral rules to guide behavioral choices. Whereas the hydraulic model ignored behavioral choices entirely, and thus was primarily of value in modeling large buildings where the timeline was dominated by travel over distance, the network models were designed to concentrate primarily or exclusively on behavior and were primarily of value in modeling small buildings where travel distances are short.

Some of the more popular current egress models adapt the format of CFD fire models. These models track the positions of individuals, as the hydraulic models did not, and so they have been more easily adapted to deal with queueing delays, as well as more complex phenomena such as counter-flows. As with CFD fire models, egress models tend to be more simplified when they are required to model very large spaces and buildings.

There is very little data available to support the behavioral rules needed by egress models both during and before egress. Empirical studies have found large variations in the time spent before egress travel begins. Most fire safety engineering studies assume that people initiate egress shortly after a shared cue, such as a building-wide fire alarm.

From a validation standpoint, most fatal victims of fire are not fatally injured while trying to escape, and almost no large-life-loss fires have involved multiple deaths because egress took too long to complete.

If the fire safety analysis requires a fully developed calculation of expected fire risk, then the failure to use egress models that reflect common practice is a major concern. If the fire safety analysis requires only a focused assessment of equivalency to existing codes, then one would be justified in examining only those behavioral scenarios for which a fully code-compliant design would produce acceptable outcomes.

Toxicity and Damage. For assessments of loss of life or health, models are needed to translate fire conditions at the location of an individual with an unacceptable adverse change in health. This is generally referred to a toxicity model, although the fire effects considered are not limited to toxic effects of smoke, but also include heat effects and effects on escape behavior, such as an inability to find the way to safety through dense smoke, see Chapters 2–4, 2–6, 3–11, SFPE (2008).

A toxicity model is built on thresholds of cumulative dose or instantaneous exposure (concentration) that produce the level of effect defined as unsatisfactory. There has been considerable controversy over the selection of unsatisfactory levels and the specification of thresholds. The earliest toxicity models defined unsatisfactory effect as a lethal dose, which focused attention on carbon monoxide and other narcotic gases. The consensus has shifted to define incapacitation as the unsatisfactory level on the theory that an incapacitated person will be stuck in place in a hazardous environment and is very likely to receive a fatal dose before receiving needed assistance to escape. This has broadened attention to irritant gases, burns, heat stress, and fire effects that act indirectly, such as smoke obscuration that does not physically incapacitate the individual,

but may make attempts to complete egress ineffective. Standards developers are working on even lower thresholds, corresponding to changes in behavior and egress effectiveness, such as reduced speed.

In addition to the controversy associated with shifting the definition of unsatisfactory further and further away from the fatal end-point, there are controversies involved in the inference of human thresholds from primarily animal-based data and on the use of safety factors to reflect the variation in sensitivities across people. The safety factors for sensitivity sometimes appear to take a calculation modification that would help to ensure safety and use it inappropriately in a prediction of likely outcomes. There is some evidence that the use of these standard factors would predict far more deaths and injuries under realistic fire conditions than actually occur.

Similar thinking is required – but is not nearly so advanced – to convert physical fire conditions near targets into estimates of property damage (for example, due to corrosion or simple soiling) or environmental damage (where the targets can be plant and animal species or regions of air, water and land that will, if contaminated, lead to harm to plants and animals).

Fire safety analysis will often compensate for the lack of advanced tools for calculating fire impact by treating any movement of fire effects into the space occupied by targets as an unacceptable outcome. This will produce conservative results, but may not produce practical results unless, for example, the lower zone of a room filling with smoke is assumed to be completely uncontaminated.

Cost. While not strictly part of the calculation of fire consequences, cost will typically be part of the overall assessment of acceptability. ASTM Committee E06 has published a number of standards to systematize the calculation of costs for various building designs.

Modeling Likelihood for Scenario Analysis

Ignition. Most fire safety analyses treat ignition empirically because the data are rarely available to apply existing science about the conditions of ignition for different materials.

If a building design is being assessed, it may be enough to estimate the likelihood of different gross types of initial burning rates (such as extended initial smoldering, ordinary flaming, fast flaming, explosion) and different areas of fire origin. The most challenging areas of origin, however, may not be the most common areas of fire origin.

The typical areas of origin are the rooms that are normally occupied such as living rooms, kitchens, bedrooms, offices, and other site-specific function areas. Fires beginning in concealed spaces, structural areas, or exterior surfaces or nearby properties such as sheds or brush, all are far less common but collectively represent a fire challenge with non-trivial likelihood and possibly heightened potential to evade or defeat protective measures.

Even empirical estimates can be improved through the use of modeling. For example, fire incident data can be used to estimate the annual likelihood of a fire beginning with ignition of a piece of upholstered furniture by a lit cigarette. Laboratory test data can be used to distinguish the relative ease of ignition by cigarette of different parts of the upholstered furniture (for example, cushion, arm, crevice). By chaining together probability estimates, one can develop a final estimate of the needed likelihood specific to the item in question.

As with reliability estimates, there are a number of failure mode models that can also be used, beginning with traditional fault trees, but there is rarely enough data to obtain full value from these sophisticated methods, and they tend to under-emphasize the human errors that historically are involved in most fire ignitions.

System Reliability. For all the different elements of fire safety built into a design, the decision to include the element and the initial specifications of the element will typically be controllable variables, but the status of the element when fire begins will be an uncontrollable variable. This is a generalization of reliability. Will the element be in condition to perform at all, and if so, will it be in condition to perform as designed and intended? This would include detection and suppression equipment that is not able to respond at all or responds ineffectively because of delay or some other problem, see Chapter 5–3, SFPE (2008).

There are many models of system and equipment failure, beginning with simple fault trees, but most cases of failure for most types of protective equipment are entirely or primarily due to human error. Models of behavior are not nearly so well

developed as models of equipment failure, and the data to validate and support behavioral models is largely lacking. For these reasons, fire safety analyses may use empirically based estimates of needed reliability parameters, although there are small quantities of such data available, and the available estimates are rarely able to distinguish different types of equipment, let alone different designs, models or brands.

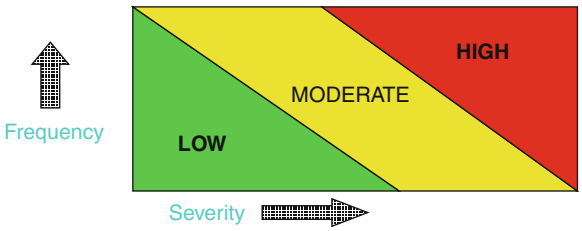
Reliability is an issue not only for active systems but also for passive features. Doors can be chocked open. Walls can be penetrated by holes. Stairways can be unenclosed and permit easy passage of fire and smoke from floor to floor. Empirical data on these failures are even more rare than data on failures of active systems.

Uncertainty. All guidance documents to the use of fire safety models emphasize the importance of uncertainty analysis, but few fire safety analyses provide depth or substantive treatment of uncertainty, see Chapter 13, SFPE (2006) and Chapter 5–4, SFPE (2008).

Research is underway to provide validation and verification for some of the major classes of models, including CFD models. Precision and bias statements are also typically lacking for standard test methods, and the nature of the major fire incident databases makes it very difficult to meaningfully characterize the uncertainty of estimates from the more detailed databases, see Chapter 5–5, SFPE (2008).

Other Risk Models

The more formal approaches described in the previous section tend to be relevant where the decision maker is a regulatory body, seeking an engineering justification for the design or adoption of a proposed requirement, or a builder, seeking an engineering case for declaring an innovative design equivalent in safety to a conventional design that is fully compliant with an existing code. Other fire-safety decision makers do not require this level of engineering detail, which is fortunate, because fire safety decisions often have to be made under conditions where the data are sparse and uncertain. The technical parameters of fire safety evaluation are complex and involve a network of interacting components, the interactions generally being nonlinear and multidimensional.



Fire Safety Modeling and Applications, Fig. 1 Fire Risk Function

Under such circumstances, are results more valid when they come from a detailed engineering analysis using extensive engineering judgment to fill in missing data or from a less-detailed approach that does not require such data but incorporates the key phenomena and interactions in a way that makes sense conceptually? Some decision makers, notably the insurance industry and managers responsible for risk management across a wide range of types of risk (not just fire), have favored the latter approach and have developed a number of different kinds of tools to support such analysis.

Four types of fire risk models are discussed here:

- qualitative fire risk matrix,
- logic diagrams, including decision trees,
- fire risk indexing, and
- stress-strength models.

In the cases of the more generic concepts of logic diagrams and probabilistic models, a specific example of application to fire safety is presented.

Qualitative Fire Risk Matrix

Fire risk is generally considered a function of the two characteristics of an unwanted event or fire scenario, that is, likelihood (frequency) and consequence (severity). Figure 1 illustrates that events with low frequency or likelihood and low severity or consequence have low risk while events with high frequency and high severity are considered high risk.

Throughout the range of likelihood and consequence, there are many levels of risk. The concept of combining these characteristics in the form of a graphical risk matrix was widely adopted in the 1960's as a systems safety technique for military systems and was documented in MIL-STD-882 (DOD 1969). It has since been incorporated in fire risk assessment guides produced by the National Fire

Fire Safety Modeling and Applications, Table 1 Probability Levels

Frequent	Likely to occur frequently ($p > 0.1$)
Probable	Likely to occur several times in system life ($p > 0.001$).
Occasional	Unlikely to occur in a given system operation ($p > 10^{-6}$)
Remote	So improbable it can be assumed this hazard will not be experienced ($p < 10^{-6}$).
Improbable	Probability of occurrence cannot be distinguished from zero ($p \sim 0.0$)

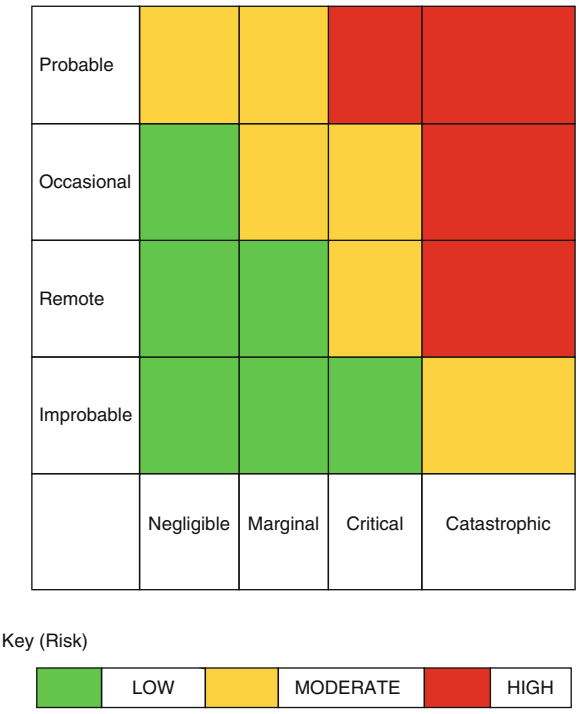
Fire Safety Modeling and Applications, Table 2 Severity Categories

Negligible	The impact of loss will be so minor that it would have no discernible effect on the facility or its operations.
Marginal	The loss will have noticeable impact on the facility. It may have to suspend some operations briefly. Some monetary investments may be necessary to restore to full operations. May cause minor personal injury.
Critical	Will cause personal injury or substantial economic damage. Loss would not be disastrous, but the facility would have to suspend at least part of its operations immediately. Reopening the facility would require significant monetary investment.
Catastrophic	Will produce death or multiple death or injuries, or the impact on operations will be disastrous, resulting in long-term or permanent closing. The facility would cease to operate immediately after the fire occurred

Protection Association (NFPA 551 2007) and the Society of Fire Protection Engineers (SFPE 2006).

The fire risk function can be quantified by applying discrete measurement scales to the axes. It then becomes a means of evaluating the relative level of fire risk for representative fire scenarios involving a building or other system. In this approach, each hazard or fire scenario is assigned a probability level and a severity category. Tables 1 and 2 below are adapted from corresponding tables in MIL-STD-882 (DOD 1969).

The probability levels and severity categories can then be used to represent the axes of a two-dimensional risk matrix such as shown in Fig. 2.



Fire Safety Modeling and Applications, Fig. 2 Risk Matrix

Each fire scenario will have a likelihood category (on the left of the matrix) and a consequence category (at the bottom of the matrix). This will locate each fire scenario within one of the 16 cells of the matrix. According to a scenario's location, the matrix indicates that improbable scenarios with negligible consequences (lower left) represent a low risk and frequently occurring scenarios with greater consequences (upper right) represent high risk levels.

Usually, fire scenarios of low risk are considered to be acceptable. Conversely, fire scenarios of high risk are unacceptable and must be eliminated or have appropriate mitigation strategies implemented. Fire scenarios of moderate risk need to be considered carefully to address whether the risk needs to be mitigated or can be considered to be acceptable. These may be the scenarios that are chosen for more detailed analysis such as that described in the previous section. In such circumstances, the matrix operates like a scenario triage tool.

Logic Diagrams, Including Decision Trees

The main tools for quantifying fire experience and other forms of failure using information on basic

events that contribute to this experience is the logic diagram. There are a number of different types of logic diagrams used in fire risk analysis, usually in the structure of a tree. Among these are:

- Decision a tree
- Fault tree
- Success tree
- Event tree
- Cause-consequence diagram

Decision trees are logic diagrams used to represent the outcomes of decisions and events made at different levels. Fault trees are logic diagrams used to represent the alternative ways in which a system can fail, resulting in a critical event, referred to as the top event in the tree. Success trees are logic diagrams used to represent alternative ways in which a certain goal can be achieved. They can be constructed as the dual of a fault tree or separately as a qualitative logic diagram such as the Fire Safety Concepts Tree, discussed in more detail below. A quantified fault or success tree applies probabilities at every branch, which permits calculation of the overall likelihood of success or failure – useful if the fire safety objectives can be expressed qualitatively – and analysis of which parts of the design provide the greatest leverage to improve the overall likelihood of success or failure.

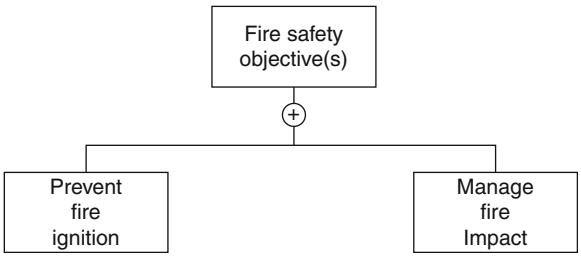
Event trees are logic diagrams used to represent the alternative ways in which a system can continue following an initial critical event, e.g., fire. Cause-consequence trees are logic diagrams that are used to represent causes (backward in time) and consequences (forward in time from a given critical event). Both replace logical relationships with temporal sequencing relationships. A quantified event tree or cause-consequence tree applies probabilities to branching points, similar to a quantified fault or success tree, but also applies costs and benefits to branching points and/or final outcomes. This permits a quantitative calculation of overall risk.

Fire Safety Concepts Tree. The Fire Safety Concepts Tree (FSCT) is a logic diagram with a tree-root like structure (NFPA 550 2007). It branches downward from specified fire safety objectives to identify all possible means of achieving fire safety. Fire safety objectives include life safety, property protection, operational continuity, environmental protection, and heritage preservation. The FSCT is very comprehensive in terms of its included objectives, and is widely used to obtain an integrated overview of the key elements of a project before more quantitative risk analysis is begun.

The Tree, as presented in the figures below, shows the elements that must be considered in building fire safety and the interrelationship of those elements. It enables a building to be analyzed or designed by progressively moving through the various levels of events in a logical manner. Its success depends upon the completeness by which each level of events is satisfied. Lower levels on the decision tree, however, do not represent a lower level of importance or performance; they represent a means for achieving the next higher level.

The Tree requires that the "Fire Safety Objective(s)" (goals) be clearly identified. These objectives describe the degree to which the building should protect its occupants, property contents, continuity of operations, and neighbors. The objectives should be stated with enough detail that success or failure in meeting them is clearly defined, rather than stated in broad or general terms.

The life safety objective, for example, might state that all occupants be safeguarded against the intolerable or untenable effects of the fire. It may be further stated that emergency personnel such as fire fighters, who may be expected to stay in areas considered too dangerous for the occupants, should be protected against unexpected collapse of the building or entrapment. A range of specific life safety objectives may be appropriate for varying types of occupancies. Nursing home requirements, for example, are vastly different from those for offices, and both differ from industrial occupancies or storage facilities.

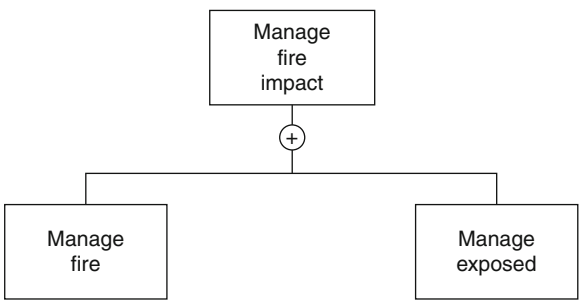


The Tree provides the logic required to achieve fire safety, i.e., it provides conditions whereby the fire safety objectives can be satisfied, but it does not provide the minimum condition required to achieve those objectives. Thus, according to the Tree, the fire

safety objectives can be met if fire ignition can be prevented or if, given ignition, the fire can be managed. This logical "OR" function is represented by the symbol (+) on the Tree.

The "Prevent Fire Ignition" branch essentially is the entry point for what could be elements of a fire prevention code. Most of the concepts described in this branch require continuous monitoring for success. Consequently, the responsibility for satisfactorily achieving the goal of fire prevention is ultimately an owner/occupant responsibility. However, some of the elements along the branch may involve choices of the designer, such as the type of central heating, main cooking, and electrical distribution equipment to be used.

It is impossible to prevent completely the ignition of fires in a building. Therefore, to reach the overall fire safety objective, from a building design viewpoint, a high degree of success in the "Manage Fire Impact" branch assumes a significant role. After an ignition occurs, all considerations shift to that branch to achieve the fire safety objectives.



According to the logic of the Tree, the impact of the fire can be managed through either the "Manage Fire" or "Manage Exposed" branches. The "OR" gate indicates that the objectives may be reached through either or both of the branches, as long as the path selected completely satisfies the fire safety objective. Naturally, it is acceptable to do both, which will increase the probability of success over using only one branch.

The Manage Fire objective can be achieved by any of three different means:

(1) Controlling the combustion process,
(2) Suppressing the fire, or
(3) Controlling the fire by construction.

Here, again, any one of these branches of the Tree will satisfy the Manage Fire event. Thus, for instance, in some fires success is achieved where the building construction controlled the fire. And in other fires, success is achieved by controlling the combustion process, either by controlling the fuel or the environment.

The "Suppress Fire" event is the output of a logical "AND" gate, which signifies that all of the elements in the level immediately below the gate are necessary to achieve the event above the gate. To accomplish the automatic suppression event, for example, all three events–detecting the fire, initiating action, and controlling the fire are necessary. Similarly, to manually suppress the fire, all six events must take place. The omission of any single event is sufficient to break the chain and cause the failure of this automatic suppression event.

In considering the Manage Exposed branch, it can be successful either by limiting the amount or number exposed or by safeguarding the exposed. For example, the number of people as well as the amount or type of property in a space may be restricted. Often this is impractical. If this is the case, the objectives may still be met by incorporating design features to safeguard the exposed.

The exposed people or property may be safeguarded either by moving them to a safe area of refuge or by defending them in place. For example, people in institutionalized occupancies such as hospitals, nursing homes, or prisons must generally be defended in place. To do this, the "Defend Exposed in Place" branch would be considered. On the other hand, alert, mobile individuals, such as those expected in offices or schools, could be moved to safeguard them from fire exposure on either a short term or long range basis depending upon other key design elements, such as a checklist of actions that might be taken. For example, "Provide Safe Destination."

The distinct advantage of the FSCT is its systems approach to fire safety. It considers all aspects of fire safety and shows how they interact to influence achievement of fire safety goals and objectives. Usefulness of the Tree is in providing an overall structure with which to analyze the potential impact of requirements or design concepts on a particular fire safety problem. It can support a decision to use a nontraditional approach to fire safety when accompanied by sound fire protection engineering principles.

The FSCT Tree in its entirety has seven levels of branches. A more complete description of the tree is

found in National Fire Protection Association document, NFPA 550, *Guide to the Fire Safety Concepts Tree* (NFPA 550 2007), which also contains detailed descriptions of tree elements, a glossary, and an administrative action guide.

Fire Risk Indexing

Fire risk indexing (FRI) is representative of the quantitative fire risk assessment that originated with the insurance rating schedule in the early 20th century. The approach has broadened to include a wide variety of applications. In general, fire risk indexing assigns values to selected variables based on professional judgment and past experience. The selected variables represent both positive and negative fire safety features and the assigned values are then operated on by some combination of arithmetic functions to arrive at a single value. This single value can be compared to other similar assessments or to a standard to rank the fire risk. Examples of fire risk indexing are presently included or referenced in current building and fire safety codes. As opposed to the pass/fail criteria of strict code compliance, indexing recognizes the value of building attributes that are superior to code requirements as well as the deficiencies. A broader collection of fire risk indexing models has evolved over the last 30 years.

Indexing refers to the agglomeration of measures of two or more attributes to produce a single summary measure that appropriately reflects all the included attributes. For example, to estimate how cold the weather feels a wind-chill factor combines the wind speed and temperature into a single measure.

Characteristics of FRI. In the spectrum of fire risk assessment methods, FRI is positioned between qualitative checklists and analytical calculation methods. While checklists can be endlessly comprehensive in their scope, this is not always practical or efficient, and the format provides no quantitative decision-making risk information related to either likelihood or severity of consequences. And while analytical calculation can provide a detailed level of resolution that is appealing to fire protection engineers, it is expensive, time-consuming, and highly dependent on large quantities of data, including extensive use of engineering judgments. FRI has the following characteristics:

- Greatly expands the usefulness of checklists by incorporating quantitative measurement
- More practical than analytical calculation methods by facilitating inclusion of fire hazards and safety features for which proven fire models and necessary experimental or field data are generally lacking, e.g., fire department response.
- Can be designed to receive input and adjustment from analytical calculations where appropriate.

Fire risk indexing applications come in many forms. Among the most popular, as described in Chapters 5–10, 12, SFPE (2008), are:

- Fire Safety Evaluation System (FSES)
- International Existing Building Code (IEBC)
- DOW Fire and Explosion Index (FEI)

Decision Analysis Background of FRI. One of the most common and most powerful heuristic decision-making techniques is multiattribute evaluation, an approach that is supported by a large body of knowledge described in the literature of decision analysis and management science. Multiattribute evaluation is used to develop simplified but robust models of complex systems. Values are assigned to important attributes of the problem based on professional judgment and experience. These values are then operated on by some combination of arithmetic functions to arrive at a single score or index. The result can be compared with other similar assessments or to a standard.

As implied above, fire safety decisions require more than one attribute to capture all relevant aspects of the consequences. If there are n attributes (x_1, x_2, x_3, . . . , x_n) for a decision problem, then an evaluation function $E(x_1, x_2, x_3, \ldots, x_n)$ needs to be determined over these measures to conduct a performance assessment. It has been established that if tradeoffs among the attributes do not depend on the levels of the remaining attributes, then a single measure of the overall outcome of a system is given by

$$E(x_1, \ldots, x_i, \ldots, x_n) = \sum_{i=1}^{n} w_i \, R_i \, (x_i)$$

where the w_i are weighting constants greater than zero and the $R_i(x_i)$ are normalizing functions of the attributes. Additional information is in Chapter 13, Rasbash et al. (2004).

Management science has long dealt with this type of problem. A large body of knowledge exists on the subject of Multiattribute Evaluation, closely related

to Multiattribute Decision Analysis, Multicriteria Decision Making, and Multiattribute Utility Theory.

Stress-Strength Models

There are many nondeterministic modeling techniques that have been applied to fire protection engineering problems. These include both stochastic and probabilistic approaches.

Stochastic models of fire growth predict the course of fire development in a building. In these models, various states occur sequentially in space and time according to probability distributions. The most common types of stochastic models used are networks and Markov chains. Other approaches may include random walks, diffusion processes, percolation theory, and epidemiologic models, see Chapter 3–14, SFPE (2008).

Probability models generally deal with a final outcome such as success or failure of a fire safety component, number of fire deaths, economic loss, spatial extent of damage, etc. They consider the outcome as a continuous random variable reaching various levels according to a generated probability distribution. Large values of the variable may follow the extreme-value distribution, see Chapter 5–8, SFPE (2008).

A promising method is the convolution integral or stress-strength model. It is generally well-established in structural integrity analysis and in civil and mechanical engineering applications, thus making it a naturally attractive overall framework for building performance analyses where there are other phenomena as well as fire at issue.

Convolution Integral. Fire safety incorporates many concepts that are not discrete events, including attributes that may exist in a greater or lesser degree, e.g. combustibility, fire resistance, suppression, etc. These attributes can be represented by a probability distribution of occurrence over the range of possible values. A critical value of such an attribute may depend upon the level of another element of the same form. For example, the necessary degree of fire resistance, a continuous variable, is dependent on the level of fire severity to which it is exposed, another continuous variable. One way to combine these elements to calculate the probability that a critical value is reached or exceeded is by using a convolution integral, also referred to as a stress-strength model (Watts 1983).

Let X be a random variable denoting the maximum stress encountered and let Y be a random variable denoting the effective strength. Since the units of stress and strength are the same, their probability density functions may be plotted on the same axes. When strength of the system is y^*, then the reliability of the system (i.e. the probability that the stress will be less than the strength) is the area under the stress curve to the left of y^*:

$$P(X \le y^*) = \int_{-\infty}^{y^*} f(x)dx$$

If the exact strength (y*) is unknown, the reliability is also a function of the strength distribution g(y):

$$P(X \le Y) = \int_{-\infty}^{\infty} \int_{-\infty}^{y} f(x)g(y)dxdy$$
$$= \int_{-\infty}^{\infty} F_x(y)g(y)dy$$

This is the usual form of the stress-strength model.

Stress-Strength Model of a Fire Barrier. Let R be a random variable that represents the fire resistance of the barrier and let S represent the stress or the severity of fire to which the barrier is exposed. Then the characteristic of interest is the probability that the fire resistance will be greater than the fire severity:

$$P(R > S) = P[(R/S) > 1.0]$$
$$= P(X > 1.0) \text{ where } X = R/S$$

and

$\ln X = \ln R - \ln S$, by the properties of logarithms.

If R and S are lognormal random variables, then $\ln R$ and $\ln S$ are normally distributed. It has been shown that a linear combination of independent, normally distributed random variables is also normally distributed. Assuming, therefore, that the fire severity and the fire barrier are independent,

$$Y = \ln X = \ln R - \ln S$$

is a normally distributed random variable with mean $\mu = \mu_{\ln R} - \mu_{\ln S}$

and variance $\sigma^2 = \sigma^2{}_{\ln R} + \sigma^2{}_{\ln S}$.

Now the probability of interest may be expressed in terms of the normal random variable Y:

$$P(X > 1.0) = P(Y > \ln 1.0)$$
$$= P(Y > 0)$$

The standard normal variate is a normally distributed random variable with a zero mean and unit standard deviation. As any normal variate (x) may be represented as a standard normal (z) by the transformation $z = (x\text{-}\mu)/\sigma$.

Thus:

$$P(Y > 0) = P[Z > -(\mu/\sigma)].$$

For any standard normal variable:

$$P(X > x) = P[X \leq (-x)]$$

The probability may then be written in the more usual form:

$$P(R > S) = P[Z \leq (\mu/\sigma)].$$

Thus the probability of a given barrier withstanding a given fire may be represented as a standard normal random variable.

The convolution integral may be used to model the relationship between two elements of fire safety represented by probability distributions. This is referred to in the structural engineering literature as a stress-strength model. The barrier model presented is similar to structural applications. However, this concept is also applicable to the evaluation of suppression systems by defining the convolution of the suppressibility of the system and the suppressibility of the fire. The model is equally suitable in application to ignitibility or other similar fire protection engineering concepts.

On a broader scale, the stress-strength model can be used to evaluate life safety from fire. Two concepts that are commonly used in this regard are ASET and RSET. ASET is a measure of the Available Safe Egress Time, a function of the space and fire growth. RSET is the Required Safety Egress Time, typically determined by calculation of the speed with which evacuation of a building or fire area takes place. If ASET > RSET then the premises are considered safe. Although a safety factor is incorporated in the evaluation, the many assumptions and variations that are inherent in the calculations indicate that a probabilistic treatment such as a stress-strength model, as has been adapted (He 2010), may have useful advantages.

Concluding Remarks

Fire-safety decision making involves objectives, available choices, and a surrounding framework that indicates, with some degree of quantification, how the choices translate into relative success in meeting the objectives. This is a classic OR/MS formulation of a decision problem. Some surrounding frameworks make more extensive use of physical models and laboratory data; they tend to be more attractive and familiar to engineers and, therefore, more appropriate when decision makers or project teams are most attuned to an engineering style of decision making. Other surrounding frameworks rely more on bare-bones logic with limited quantification and that primarily in probabilistic form; they tend to be more attractive to decision makers for whom the engineering aspects are embedded in a much larger statement of the problem. In either setting, there are ways to use OR/MS methods, and there are resources available from other technical fields that can be used in combination with OR/MS methods or in ways similar to traditional OR/MS approaches.

See

- ▶ Decision Analysis
- ▶ Emergency Services
- ▶ Multi-attribute Utility Theory
- ▶ Multiple Criteria Decision Making

References

DOD. (1969). *MIL-STD-882, military standard system safety program requirements*. Washington, DC: US Department of Defense.

He, Y. (2010). Linking safety factor and failure probability for fire safety engineering. *Journal of Fire Protection Engineering, 20*, 199–217.

NFPA 550. (2007). *Guide to the fire safety concepts tree*. Quincy, MA: National Fire Protection Association.

NFPA 551. (2007). *Guide for the evaluation of fire risk assessments*. Quincy, MA: National Fire Protection Association.
Olenick, S. M., & Carpenter, D. J. (2003). An updated international survey of computer models for fire and smoke. *Journal of Fire Protection Engineering, 13*, 87–110.
Rasbash, D., et al. (2004). *Evaluation of fire safety*. Chichester: Wiley.
SFPE. (2006). *Engineering guide: Fire risk assessment*. Bethesda, MD: Society of Fire Protection Engineers.
SFPE. (2007). *SFPE engineering guide to performance-based fire protection* (2nd ed.). Quincy, MA: National Fire Protection Association.
SFPE. (2008). *Handbook of fire protection engineering* (4th ed.). Quincy, MA: Society of Fire Protection Engineers and National Fire Protection Association.
Watts, J. (1983). A probability model for fire safety tree elements. *Hazard Prevention*, November/December, 14–15.

Firing a Rule

The activity of carrying out the actions in a rule's conclusion, once it has been established that the rule's premise is true.

See

- ▶ Artificial Intelligence
- ▶ Expert Systems

First Feasible Solution

The feasible (usually basic) solution used to initiate the Phase 2 procedure of the simplex method. The solution satisfies both $\boldsymbol{Ax} = \boldsymbol{b}$ and $\boldsymbol{x} \geq \boldsymbol{0}$. The first feasible solution is often a product of the Phase I procedure of the simplex method, while in other instances, it is user-supplied or generated by previous solutions of the problem.

See

- ▶ Phase I Procedure
- ▶ Phase II Procedure
- ▶ Simplex Method (Algorithm)

First-Fit Decreasing Algorithm

- ▶ Bin-Packing

First-order Conditions

Conditions involving first derivatives.

Fixed-Charge Problem

A problem in which a one-time cost is incurred only if the associated variable is positive. The fixed cost is added to the linear variable cost. Problems with linear constraints and fixed charges are usually reformulated using subsidiary binary variables.

Fleet Assignment

See

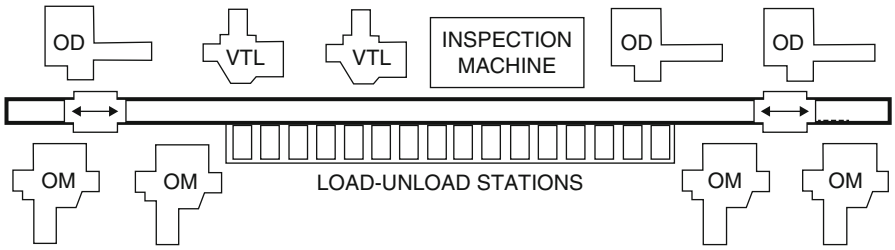
- ▶ Airline Industry Operations Research

Flexible Manufacturing Systems

Kathryn E. Stecke
The University of Texas at Dallas, Richardson, TX, USA

Introduction

In the metal-cutting industry, a flexible manufacturing system (FMS) is an integrated system of machine tools linked by automated material handling. Because of the versatility of the machine tools and the quick (in seconds) cutting tool interchange capability, these systems are quite flexible with respect to the number of part types that can be produced simultaneously and in low (sometimes unit) batch sizes. These systems can be almost as flexible as a job shop, while having the ability to attain nearly the efficiency of a well-balanced assembly line.



Flexible Manufacturing Systems, Fig. 1 Sundstrand/Caterpillar FMS

An FMS consists of several computer numerically controlled machine tools, each capable of performing many operations. Each machine tool has a limited capacity tool magazine that holds all of the cutting tools required to perform each operation. Once the appropriate tools have been loaded in the tool magazines, the machines are under computer control. During system operation, the automatic tool interchange capability of each machine allows no idle set-up time in between consecutive operations or between the use of consecutive tools. When a new tool is required, the tool magazine rotates into position, and the changer automatically interchanges the new tool with the one that is in the spindle in seconds. Each part type that is machined is defined by several operations. Each operation requires several cutting tools (about 5-20). All tools for each operation need to occupy slots in one or more of the machine tool's magazine.

Each cutting tool takes 1, 3, or 5 slots in a magazine. Magazines can have 40-160 slots, typically sixty to eighty. Tools wear and break, so a computer needs to track the "lives" of all tools. A tool that breaks during the cut can severely damage the part and sometimes the machine or spindle. Tools can be delivered to the FMS either manually or automatically, for example, via automated guided vehicles, with the delivered tools manually or automatically loaded into the magazines. Tooling information is discussed in Stecke (1983) and Hirvikorpi et al. (2007).

An FMS has an automated materials handling system that transports parts from machine to machine and into and out of the system. These may consist of wire-guided automated guided vehicles, a conveyor system, or tow-line carts, with a pallet interchange with the machines. The interfaces between the materials handling system and the parts are pallets and fixtures. Pallets sit on the cart and fixtures hold and clamp the parts onto the pallets. Pallets are identical and fixtures are usually of different types. Thus, fixtures are able to hold securely different types of parts and in different orientations. The number of pallets in an FMS defines the maximum amount of work-in-process inventory in the system.

After some machining, parts are often checked at the machine by automatically interchanging a probe into the spindle. The probe does some at-the-machine inspection of the cuts that were made. After several operations, a cart may bring the part to a washing station to remove the chips before either further machining, refixturing, or inspection.

An example of an FMS, built by Sundstrand Machine Tool Company for Caterpillar Tractor Company, is shown in Fig. 1 (Stecke and Solberg 1981). This FMS consists of four 5-axes mills (OM), three 4-axes drills (OD), two vertical turning lathes (VTL), and an inspection station. The parts machined are housings for automatic transmissions—transmission cases, transmission covers, an assembly of these two, and several sizes of each.

Detailed descriptions of several systems can be found in Stecke (1992). A decision to automate should be based on both economic comparisons and strategic considerations. Assuming that management has decided that flexible manufacturing is appropriate for a particular application, perhaps to increase capacity in a certain department producing changing products or for new families of part types, there are many design issues that have to be addressed. Details and descriptions of these design problems are given in Stecke (1985, 1992).

The amount of flexibility that is needed or desired has to be decided, as this helps to determine the degree of automation and the type of FMS to be designed. Impacting the latter decision is the type of automated material handling system that will move the parts from machine to machine and into and out of the system. Browne et al. (1984), Sethi and Sethi (1990), and de Treville et al. (2007) discuss a spectrum of flexibility types and options.

Efficient and accurate mathematical and other models are useful in helping to define the appropriate FMS design. See Buzacott and Yao 1986; Stecke 1983; and Solberg 1977. Following the development and subsequent implementation of the selected FMS design, models are then useful in helping to set up and schedule production through the system.

FMS Planning and Scheduling

Because of the quick automated cutting tool capability, there is negligible set-up time associated with a machine tool in between consecutive operations, as long as all of the cutting tools required for that next operation have previously been loaded into the machine tool's limited capacity tool magazine. However, determining which cutting tools should be placed in which tool magazine and then loading the tools into the magazine requires some planning and system set-up time. Those set-up decisions that have to be made and implemented before the system can begin to manufacture parts are called FMS planning problems (Stecke 1983). When the system has been set-up and can begin production, the remaining problems are those of FMS scheduling.

The first FMS planning problem is to decide which of the part types that have production requirements (either forecasted demand or customer orders) should be manufactured next during the same time over the immediate time period. This information can be used to help determine the amount of pooling among the identical machine tools that can occur. Pooling, or identically tooling all machines that are in the same machine group, has many system benefits. For example, alternative routes for parts are automatically allowed, and machine breakdowns may not cause production to stop. This is because all machine tools in a group, being tooled identically, are able to perform the same operations.

Another FMS planning problem is to determine the relative ratios at which the selected part types should be on the system. Making this decision correctly can help an FMS to attain good utilization. The limited numbers of pallets and fixtures of each fixture type impact these production ratios. Finally, each operation and its associated cutting tools of the selected set of part types has to be assigned to one or more of the machine tools in an intelligent manner. Different loading objectives that can be followed are applicable in different situations. When all of these decisions have been made and the cutting tools loaded into the selected tool magazines, production can begin. Then the following FMS scheduling problems have to be addressed.

These problems are concerned with the operation of the system after it has been set up during the planning stage. One problem is to determine an appropriate policy to input the parts of the selected part types into the FMS, or efficient means to determine which parts to input next. He and Smith (2007) suggest an approach.

Then, applicable algorithms to schedule the operations of all parts through the system have to be determined. Real-time scheduling is usually more appropriate for these automated systems, as opposed to a fixed schedule. Tool breakage, down machine tools, etc., would totally disrupt a fixed schedule. However, a fixed schedule is useful as an initial guideline to follow. Potential scheduling methods range from simple dispatching rules to sophisticated algorithms having look-ahead capabilities. Machine breakdowns and many other system disturbances should be considered when developing scheduling and control procedures. If the system is set-up during the planning phase with sufficient care and flexibility, the scheduling function will be much easier.

FMS control involves the continuous monitoring of the system to be sure that it is doing what was planned for it to do and is meeting the expectations set up for it. For example, during the FMS design phase, policies should be determined to handle breakdown situations of many types. In any case, it is desirable to reallocate operations and reload the cutting tools (if they have to be) so that the tool changing time is minimized. Monitoring procedures for both the processes and cutting tool lives have to be specified, as well as methods to collect data of various types, e.g., monitoring and breakdown. Tool life estimates should be reviewed and updated. Reasons for process errors have to be found—machine or pallet misalignment, cutting tool wear and detection, chip problems—and the problems corrected.

Because the planning and scheduling problems are complex and require a lot of data to be considered, many of these problems have been framed and subdivided within a hierarchy. The solution of each subproblem provides constraints on problems lower in

the hierarchy. The partition of FMS problems into planning (before time zero) and scheduling (after production begins) is one example of a hierarchy. The FMS planning problems are another hierarchical decomposition of a system set-up problem. Stecke (1983, 1985) describes hierarchical and iterative approaches to several of these problems.

FMS Models

Models are useful in identifying key factors that will affect system performance and to provide insights into how a system behaves and how the system components interact. Models should be applied to help determine the appropriate procedures to design and set up a system or strategies to help run a system efficiently.

Depending on the amount of information that is built into a particular model, simulation has the potential to be the most detailed and flexible model, allowing as much detail as desired or necessary to mimic reality. Simulation can also potentially be the most expensive and time-consuming to develop, debug, and run. Many computer runs may be required to investigate the possibilities before a decision is made. Simulation has been used to help design an FMS and to solve operation problems. Stecke (1981) and Schriber and Stecke (1988) have used simulation models to address some FMS planning and scheduling problems.

Both open and closed queueing networks have been used to model an FMS at an aggregate level of detail. These models can take into account the interactions and congestion of parts competing for the same machines and the uncertainty and dynamics of an FMS. Most simple queueing networks require as input, certain average values, such as the average processing time of an operation at a particular machine tool and the average frequency of visits to a machine. The outputs obtained are average values and are useful for evaluating the performance of a suggested system configuration. Such outputs include the steady state expected production rate, mean queue lengths, and machine utilizations.

Solberg (1977) first suggested the use of a simple, single-class, multiserver, closed queueing network to model an FMS. His computer program, called CAN-Q, analyzes product-form queueing network FMS models. A review of related analytical queueing network models is given in Buzacott and Yao (1986).

Some FMS problems have been formulated mathematically, either as nonlinear integer programs or as linear and integer programs (Stecke 1983). Depending on the problems, some formulations are detailed and tractable and, thus, useful; other formulations, however, are detailed and untractable. Heuristics and algorithmic solution approaches have been developed from the exact formulations. Stecke (1992) describes other FMS models. Each model is useful under different circumstances and for different types of problems. For some problems, a hierarchy of models is used to solve them. Several application areas for flexible manufacturing are given in the following sections.

Flexible Manufacturing Applied to Mass Customization

A flexible manufacturing system can be a key tool for companies that aim to compete using mass customization or mass personalization. Mass personalization has been defined to be a strategy where a company can serve profitably a market of one person at a time . . . rather an extreme application of mass customization (Kumar 2007; Chen and Tseng 2007). Kumar and Stecke (2007) develop a methodology that measures the effectiveness of a mass customization and personalization strategy using a mass customization and mass personalization effectiveness index. The ability of an FMS to produce in batches of size one clearly is an advantage in implementing a mass customization strategy. He and Smith (2007) describe an algorithm that can be used for this purpose. An FMS's ease in producing variants of existing products is clearly useful for mass customization.

Planning Capacity in Flexible Manufacturing

For an FMS, if planned well, there is no setup time in between consecutive operations, thus enabling a very high capacity utilization. Various aspects of capacity planning in flexible manufacturing are discussed in Deif and ElMaraghy (2007), Hassanzadeh and Maier-Speredelozzi (2007), and

Zaeh and Mueller (2007). Koltai and Stecke (2008) specify methods to calculate the available capacity in an FMS without needing to specify part routes through the system, providing a robust capacity planning tool. Matta, Tomasella, and Valente (2007) address capacity-level reconfiguration to match market demands. A Markov decision problem considers both capacity expansion and reduction possibilities.

Reconfigurable Manufacturing

A reconfigurable manufacturing system (RMS) is an extension of an FMS where the system can be quickly changed to provide significant capabilities when needed that did not exist before. An example is a rapid machine tool reconfiguration from 3-axis capability to 4-axis capability. This is not a quick or easy accomplishment without significant advance planning and new (reconfigurable) machine tool design. Various aspects of reconfiguration are discussed in Koren (2010) and Kuzgunkaya and ElMaraghy (2007). Youssef and ElMaraghy (2007) suggest approaches to determine an appropriate RMS configuration that includes machine layout, equipment selection, and assignment of operations to machines.

See

- ▶ Job Shop Scheduling
- ▶ Markov Decision Processes
- ▶ Networks of Queues
- ▶ Production Management
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Browne, J., Dubois, D., Rathmill, K., Sethi, S. P., & Stecke, K. E. (1984). Classification of flexible manufacturing systems. *FMS Magazine, 2*, 114–117.

Buzacott, J. A., & Yao, D. D. W. (1986). Flexible manufacturing systems: A review of analytical models. *Management Science, 32*, 890–905.

Chen, S., & Tseng, M. M. (2007). Aligning demand and supply flexibility in custom product co-design. *International Journal of Flexible Manufacturing Systems, 19*, 596–611.

de Treville, S., Bendahan, S., & Vanderhaeghe, A. (2007). Manufacturing flexibility and performance: Bridging the gap between theory and practice. *International Journal of Flexible Manufacturing Systems, 19*, 334–357.

Deif, A. M., & ElMaraghy, H. A. (2007). Assessing capacity scalability policies in RMS using system dynamics. *International Journal of Flexible Manufacturing Systems, 19*, 128–150.

Hassanzadeh, P., & Maier-Speredelozzi, V. (2007). Dynamic flexibility metrics for capability and capacity. *International Journal of Flexible Manufacturing Systems, 19*, 195–216.

He, Y., & Smith, M. L. (2007). A dynamic heuristic-based algorithm to part input sequencing in flexible manufacturing systems for mass customization capability. *International Journal of Flexible Manufacturing Systems, 19*, 392–409.

Hirvikorpi, M., Knuutila, T., Leipälä, T., & Nevalainen, O. S. (2007). Job scheduling and management of wearing tools with stochastic tool lifetimes. *International Journal of Flexible Manufacturing Systems, 19*, 443–462.

Koltai, T., & Stecke, K. E. (2008). Route-independent analysis of available capacity in flexible manufacturing systems. *Production and Operations Management, 17*, 211–223.

Koren, Y. (2010). *The global manufacturing revolution: Product-process-business integration and reconfigurable systems*. New York: John Wiley & Sons.

Kumar, A. (2007). From mass customization to mass personalization: A strategic transformation. *International Journal of Flexible Manufacturing Systems, 19*, 533–546.

Kumar, A., & Stecke, K. E. (2007). Measuring the effectiveness of a mass customization and personalization strategy: A market and organizational-capability-based index. *International Journal of Flexible Manufacturing Systems, 19*, 548–569.

Kuzgunkaya, O., & ElMaraghy, H. A. (2007). Economic and strategic perspectives on investing in RMS and FMS. *International Journal of Flexible Manufacturing Systems, 19*, 217–246.

Matta, A., Tomasella, M., & Valente, A. (2007). Impact of ramp-up on the optimal capacity-related reconfiguration policy. *International Journal of Flexible Manufacturing Systems, 19*, 173–194.

Schriber, T. J., & Stecke, K. E. (1988). Machine utilizations achieved using balanced FMS production ratios. *Annals of Operations Research, 15*, 229–267.

Sethi, A. K., & Sethi, S. P. (1990). Flexibility in manufacturing: A survey. *International Journal of Flexible Manufacturing Systems, 2*, 289–328.

Solberg, J. J. (1977). A mathematical model of computerized manufacturing systems. *Proceedings of the 4th International Conference on Production Research*, Tokyo. Japan.

Stecke, K. E. (1983). Formulation and solution of nonlinear integer production planning problems for flexible manufacturing systems. *Management Science, 29*, 273–288.

Stecke, K. E. (1985). Design, planning, scheduling, and control problems of flexible manufacturing systems. *Annals of Operations Research, 3*, 3–12.

Stecke, K. E. (1992). Flexible manufacturing systems: Design and operating problems and solutions. In W. K. Hodson (Ed.), *Maynard's Industrial Engineering Handbook* (4th ed.). New York: McGraw-Hill.

Stecke, K. E., & Solberg, J. J. (1981). Loading and control policies for a flexible manufacturing system. *International Journal of Production Research, 19*, 481–490.
Youssef, A. M. A., & ElMaraghy, H. A. (2007). Optimal configuration selection for reconfigurable manufacturing systems. *International Journal of Flexible Manufacturing Systems, 19*, 67–106.
Zaeh, M. F., & Mueller, N. (2007). A modeling approach for evaluating capacity flexibilities in uncertain markets. *International Journal of Flexible Manufacturing Systems, 19*, 151–172.

Flight Scheduling

► Airline Industry Operations Research

Float

The amount of time a project job can be delayed without affecting the duration of the overall project. Total float is the difference between the time that is calculated to be available for a work item to be completed and the estimated duration of that item.

See

► Network Planning

Flow

The amount of goods or material that are sent from one node (source) in a network to another node (sink).

See

► Network Optimization

Flow Shop

► Scheduling and Sequencing

Flow Time

► Scheduling and Sequencing

FMS

► Flexible Manufacturing Systems

Forecasting

Andreas Graefe[1], Kesten C. Green[2] and J. Scott Armstrong[3]
[1]LMU Munich, Munich, Germany
[2]University of South Australia, Adelaide, South Australia, Australia
[3]University of Pennsylvania, Philadelphia, PA, USA

Introduction

The field of forecasting is concerned with making statements about matters that are currently unknown. The terms forecast, prediction, projection, and prognosis are interchangeable as commonly used. Forecasting is also concerned with the effective presentation and use of forecasts.

Useful knowledge comes from empirical comparisons of alternatives and this entry is concerned primarily with evidence-based or scientific procedures. Scientific knowledge about forecasting has been summarized as a set of principles that are available at the Forecasting Principles Internet Web site.

Before forecasting, one should consider whether it is necessary. Forecasting is needed only if there is uncertainty; a forecast that the tide will turn is of no value. Forecasts are also unnecessary when one can control events. For example, predicting the temperature in your home does not require forecasting because you can control it. Nevertheless, many situations are uncertain, and proper forecasting procedures can help to reduce and assess uncertainty and thereby help managers to make better decisions.

Forecasting and Planning

Forecasting should not be confused with planning. Whereas planning is concerned with what the planner thinks the future should be like, forecasting is concerned with what it *will* be like. Figure 1 summarizes the appropriate relationships between the two activities.

Managers should start by planning. Forecasting procedures are then used to predict outcomes for the plans. If the managers do not like the forecasts, the planning and forecasting processes can be repeated until a plan is found that leads to forecasts of acceptable outcomes. The best plan can then be implemented and actual outcomes monitored so that the feedback can be used in the next planning period.

Progress in Forecasting

A strong emphasis on empirical comparisons of alternative methods has helped forecasting researchers to achieve many advances in forecasting, especially since 1980. The founding of the *International Institute of Forecasters*, a multidisciplinary society of researchers and practitioners, encouraged useful forecasting research. The Institute established two academic journals in the 1980s: the *Journal of Forecasting* and the *International Journal of Forecasting*. The Institute has also organized an *International Symposium on Forecasting* every year since 1981.

Perhaps the most influential paper in the field of forecasting is the M-competition paper (Makridakis et al. 1982). This paper was based on a study in which forecasters were invited to describe and apply the forecasting method that they thought would be best for deriving forecasts for many times series. Entrants submitted their forecasts to an umpire who calculated the errors. This was the first in a series of M-competition studies, the most recent being the M3-Competition (Ord et al. 2000). The M-competitions initiated a remarkable growth in experimental studies and, consequently, a rapid development in knowledge about forecasting. For a summary of progress in forecasting to 2005, see Armstrong (2006a).

Forecasting Methods

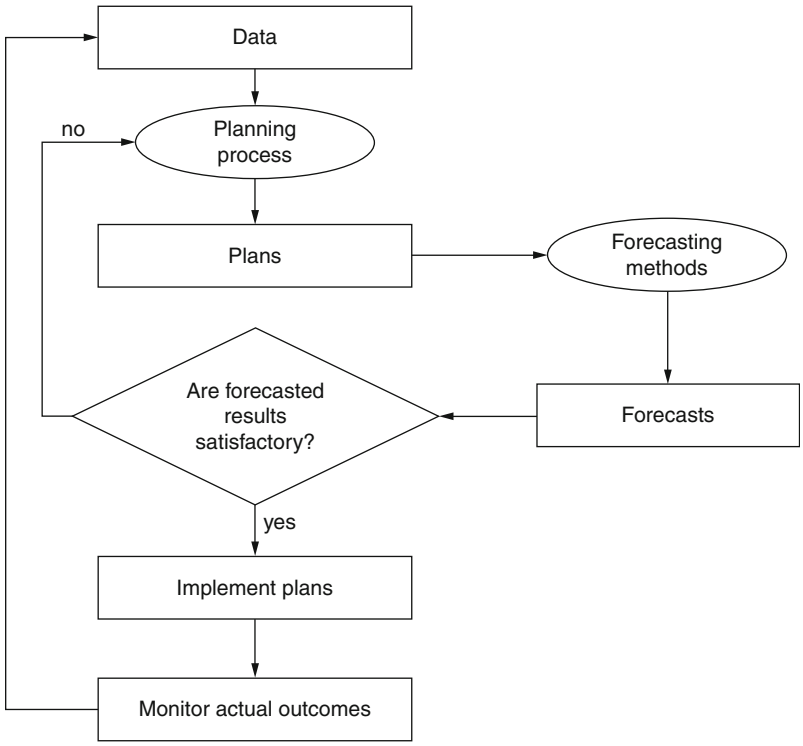
The Methodology Tree for Forecasting (Fig. 2) is a classification schema of all forecasting methods organized on the basis of the source of the knowledge the forecaster has about the situation. Some methods use primarily judgmental or qualitative knowledge while others require statistical data. There is an increasing integration in the use of judgment and statistics in the procedures as one follows the Tree down. Makridakis et al. (1998), and Armstrong (1985) provide instructions on how to use many of the methods.

The most common way to make forecasts is to ask experts to think about a situation and predict what will happen. If experts' forecasts are derived in an unstructured way the approach is referred to as unaided judgment. It is fast, can be inexpensive when few forecasts are needed, and can be appropriate when small changes are expected. It is most likely to be useful when the forecaster knows the situation well, makes frequent forecasts, and gets good feedback about the accuracy of his forecasts, as is the case with short-term weather forecasting and sports betting. Harvey (2001) described principles for improving expert forecasts.

Expert forecasting refers to combining forecasts obtained from experts using validated structured techniques. Which method is most appropriate depends on time constraints, dispersal of knowledge, access to experts, expert motivation, and need for confidentiality. To use expert forecasting methods, one should obtain the services of between five and twenty diverse experts who each have relevant information. Pre-test questions to elicit forecasts from the experts, and specify procedures for combining the forecasts obtained from the experts (e.g., use the median) in advance.

It is best if the experts do not make the their forecasts in a traditional meeting (Armstrong 2006b). The nominal group technique (NGT), developed by Van de Ven and Delbecq (1974), avoids some of the drawbacks that traditional meetings have for forecasting by imposing a structure on the interactions of the experts. Group members work independently and generate individual forecasts. The group then conducts an unstructured discussion to deliberate on the problem. Finally, group members work independently and provide their final forecasts. The NGT forecast is a combination of the individuals' final forecasts.

Consider also the Delphi method for combining the forecasts of experts. Delphi involves at least two



Forecasting, Fig. 1 Framework for forecasting and planning

rounds of anonymous interaction between experts. An administrator summarizes individual forecasts and arguments and reports this feedback to participants after each round. In the light of the feedback, the participating experts provide revised forecasts and further reasoning. The Delphi forecast is a combination of the final round forecasts.

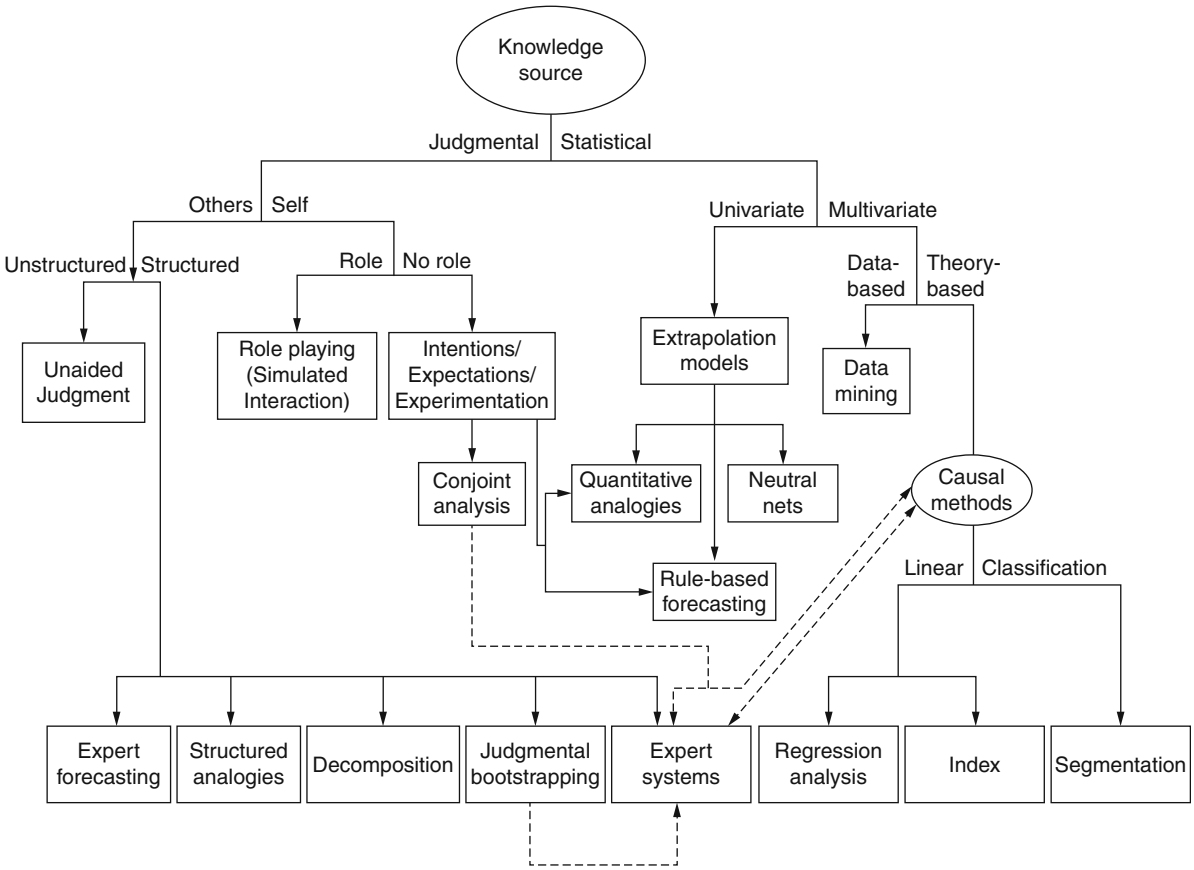
Rowe and Wright (2001) found that Delphi improved accuracy over unstructured groups in five studies, harmed accuracy in one, and the comparison was inconclusive in two. In his summary of the literature, Woudenberg (1991) also found Delphi forecasts to be slightly superior to those from unstructured interactions. Delphi is most suitable if one expects experts to have different information, but it can be conducted as a simple one-round survey for situations in which experts possess similar information. Software for conducting Delphi surveys is available on the Internet, see Principles of Forecasting and Delphi Survey Web sites.

Prediction markets can be useful for combining forecasts to provide continuously updated numerical or probability forecasts. In a prediction market, anonymous participants reveal information by trading contracts whose prices reflect the aggregated group opinion. Incentives to participate in a market may be monetary or non-monetary. Although prediction markets seem promising, to date there has been no published meta-analysis of the accuracy of prediction market forecasts. For a discussion of the relative merits of prediction markets and Delphi see Green et al. (2007).

Where it is not possible or feasible to obtain forecasts from several experts, ask a single expert to provide a second forecast by assuming the first forecast was wrong and specifically considering information that was ignored when making the first forecast. Herzog and Hertwig (2009) called this procedure dialectical bootstrapping.

The structured analogies method uses information about similar situations to obtain forecasts. Experts identify situations that are analogous to a target situation, describe similarities and differences to the target, and then derive an overall similarity rating. The outcome or decision implied by each expert's top-rated analogy is the structured analogies forecast from that expert.

Green and Armstrong (2007) analyzed structured analogies for the difficult problem of forecasting the decisions people will make in conflict situations.



Forecasting, Fig. 2 Methodology Tree for Forecasting

When experts were able to identify two or more analogies and their best analogy was from direct experience, 60% of structured analogies forecasts were accurate compared to 32% of experts' unaided judgment forecasts, the latter being little better than guessing.

Decomposition involves breaking down a forecasting problem into components that are easier to forecast than the aggregate problem. The components are multiplicative (e.g., to forecast a brand's sales, one would separately forecast the relevant population, the purchase rate of the product type, and the market share of the brand). Decomposition is most likely to be useful in situations involving high uncertainty. High uncertainty is common when making forecasts that involve large numbers, such as a country's GDP. Results from three studies involving 15 tests and found that under high uncertainty, judgmental decomposition led to a 42% reduction in error (MacGregor 2001). When the conditions for decomposition were met, decomposition of time series by causal forces reduced forecast errors by two-thirds (Armstrong et al. 2005).

Judgmental bootstrapping is a method for deriving a forecasting model by regressing experts' forecasts against the information the experts used to make their forecasts. The method is useful when expert judgments have predictive validity but data are scarce (e.g., forecasting new products) and outcomes are difficult to observe (e.g., predicting performance of executives). Once developed, judgmental bootstrapping models are a low-cost forecasting method. A meta-analysis found judgmental bootstrapping to be more accurate than unaided judgment in 8 of 11 comparisons. Two tests found no difference, and one found a small loss in accuracy. The typical error reduction was about 6% (Armstrong 2006a).

Expert systems are forecasting rules derived from the reasoning experts use when they make forecasts.

They can be developed using knowledge from diverse sources such as surveys, interviews of experts, or protocol analysis in which the expert explains what he is doing as he makes forecasts. A meta-analysis on the predictive validity of the method found that expert systems were more accurate than unaided judgment in six comparisons, similar in one, and less accurate in another. Expert systems were less accurate than judgmental bootstrapping in two comparisons and similar in two. Expert systems were more accurate than econometric models in one comparison and as accurate in two (Collopy et al. 2001).

If people have valid intentions or expectations about how they would behave in a situation, responses from surveys of intentions or expectations can be used to make forecasts. Both methods are most useful when (1) responses can be obtained from a representative sample, (2) responses are based on good knowledge, (3) respondents have no reason to lie, and (4) new information is unlikely to change people's behavior. The Juster Scale, a zero-to-ten probability scale, is recommended. For example, ask a respondent how likely it is that the respondent will make an international trip in the next 12 months. A quantitative forecast for the population can be derived by averaging the survey responses and multiplying by the population. Intentions are more limited than expectations in that they are most useful when (a) the event is important to the respondent, (b) the behavior is planned, and (c) the respondent can fulfill the plan (e.g., their behavior is not dependent on the behavior of others). Morwitz (2001) provided evidence on the validity of the methods, and guidance on how to conduct the surveys and analyze the data for forecasting.

A meta-analysis involving 47 comparisons with over 10,000 subjects found a strong relationship between intentions and behavior (Kim and Hunter 1993). Another meta-analysis found that purchase intentions provide unbiased predictions of behavior (Wright and MacRae 2007).

One can conduct experiments by varying key causal variables in a systematic way, ensuring that the changes do not correlate with one another. They can be used to estimate relationships and use these estimates to derive forecasts. Experiments can be used for such problems as to predict the effects of different policies or regulatory schemes, or to assess the effectiveness of alternative advertisements. Test markets for new products are a form of experiment.

Role playing involves asking people to think and behave in ways that are consistent with a role and situation described to them. Role playing for the purpose of predicting the behavior of people who are interacting with each other is called simulated interaction. The decisions made in the simulated interactions are used as forecasts of the actual decision. Green (2005) found that 62% of simulated interactions forecasts were accurate for eight diverse conflict situations. By comparison, only 31% of forecasts from the traditional approach (unaided expert judgment) were accurate. Game theory experts' forecasts had a similar level of accuracy: 31%. Experts' unaided judgment forecasts and game theorists' forecasts were little better than chance, which was at 28% for the eight conflicts examined.

Simulated interaction is useful when little or no quantitative data are available, when the situation to forecast is unique or unusual, and when decision makers wish to predict the effects of different policies or strategies (e.g. what pay offer will avoid a strike or what strategy is most likely to induce rebel forces to surrender). Simulated interactions can be conducted quite cheaply by using students to play the roles. In a simplified form, they can also be conducted rapidly. For example, the New Zealand Armed Offenders Squad test different approaches for dealing with an armed stand off by simulating each before choosing which one to employ.

Conjoint analysis is a method for eliciting people's preferences for different possible offerings (e.g. for alternative mobile phone designs or for different political platforms) by exposing people to several combinations of features (e.g. weight, price, and screen size of a mobile phone.) The possibilities can be set up as experiments where variations in each variable are unrelated to variations in other variables. Regression-like analyses are then used in order to predict the combination of features that people will find most desirable.

Extrapolation models use time-series data on the situation of interest (e.g., data on automobile sales from 1947–2010). One extrapolation method is exponential smoothing, which implements the principle that more recent data should be weighted more heavily when forecasting. Quantitative extrapolation methods do not use knowledge about the situation but rather assume that the causal forces that have shaped history will continue to the forecast

horizon. If this assumption turns out to be wrong, forecast errors can be large. As a consequence, one should only extrapolate trends when they are consistent with the prior expectations of domain experts and with long-term trends and variability.

Extrapolation can also be used for cross-sectional data. For example, to predict whether a particular job applicant will last more than a year on the job, one could use the percentage of the last 50 people hired for that type of job who lasted more than a year.

Armstrong (2001b) provides guidance on the use of extrapolation. An example is the advice to be conservative when the situation is uncertain, as is usually the case with long forecast horizons. In such a situation, one should reduce the magnitude of the trend in the data as the forecast horizon increases, a procedure known as trend damping.

Quantitative analogies are similar to structured analogies with the exception that there is ample data to analyze for each analogous situation. Experts identify analogous situations for which time-series or cross-sectional data are available, and rate the similarity of each analogy to the target situation. These inputs are used to derive a forecast. This method is especially useful in situations with little historical data. For example, one could average data from cafés in suburbs identified by experts as similar to a new (target) suburb in order to forecast demand for the services of a café in the target suburb.

Rule-based forecasting combines expert domain knowledge and statistical techniques for extrapolating time series. Experts are used to identify features of the forecasting problem that cannot be identified automatically from the series to be forecast. Primarily, experts are needed to identify the causal forces that are acting on trends in the series. For example, in forecasting the real price of oil, experts might identify that, over the long-term, innovation and discovery act as downward forces but may conclude that, in the short term, political actions will lead to upward pressure on prices. Rule-based forecasting was found to be more accurate than extrapolation methods that did not incorporate expert knowledge, especially for long-term forecasts (Collopy and Armstrong 1992).

Causal models include regression analysis, the index method, and segmentation. These methods are useful if data are available on variables that might affect the situation of interest. Allen and Fildes (2001) found that forecasts from causal models were more accurate than forecasts derived from extrapolating the dependent variable. Theory, prior research, and expert domain knowledge provide information about relationships between the variable to be forecast and explanatory variables. Because causal models can relate planning and decision-making variables to forecasts, they can be used to forecast the effects of different policies.

Regression analysis involves estimating the coefficients of a causal model from historical data. Models consist of one or more regression equations used to represent the relationship between a dependent variable and one or more explanatory variables. They are useful in situations with few important variables and many reliable observations that include data in which the causal variables varied independently of one another.

Important principles for developing regression models are to (1) use prior knowledge and theory, not statistical fit, for selecting variables and for specifying the directions of effects, (2) use simple models, and (3) discard variables if the relationship estimated from the data conflicts with prior evidence on the nature of the relationship.

Because regression models tend to over-fit data, damping the estimated coefficients of a model tends to improve out-of-sample forecast accuracy, particularly if uncertainty is high as occurs when one has small samples and many variables. As this situation is common for many prediction problems, unit (or equal weight) models – the most extreme case of damping – often yield more accurate forecasts than models with statistically fitted regression coefficients (Dana and Dawes 2004). Cuzán and Bundrick (2009) found that equal-weight versions of prominent presidential election forecasting regression models provided more accurate forecasts than the original models.

Damping seasonal factors improves forecast accuracy. Miller and Williams (2003, 2004) developed a procedure for damping seasonal factors estimated from historical data. When they applied the procedure to the 1,428 monthly time series from the M3-Competition, forecasts were more accurate for 68% of the series. Averaging seasonal factors estimated from related series reduced forecast error by about 20% in a study by Bunn and Vassilopoulos (1999) and, in a study by Gorr et al. (2003), pooling of

seasonal crime rate factors across six precincts increased forecast accuracy by 7%.

The index method is suitable for situations in which many causal variables are important. Use prior empirical evidence to identify predictor variables and to assess each variable's directional influence on the outcome. Results from experiments are especially valuable. If possible, draw on findings from meta-analyses. If no data or prior studies are available, independent expert judgments can be used to choose the variables and determine the directions of their effects. If prior knowledge is ambiguous or contradictory and thus does not allow for estimating a variable's directional influence on the outcome, do not include the variable in the model. Index scores are calculated by adding the number of variables that favor the outcome.

The index method is especially useful for selection problems (e.g., which candidate will win an election or which advertisement will pull best). If sufficient historical data are available, index models can be generated to predict numerical outcomes (e.g., a candidate's vote-share) by regressing index scores against historical data. The index method can be used if valid and reliable quantitative data are scarce relative to the number of causal variables or if there is little need for precision in estimating the magnitude of relationships. Armstrong and Graefe (2011) describe the use of the method to make early forecasts of the outcomes of U.S. presidential elections from biographical information about potential candidates. Based on a list of 59 variables, their relative index scores correctly predicted the winner in 27 of the 29 elections from 1896 to 2008.

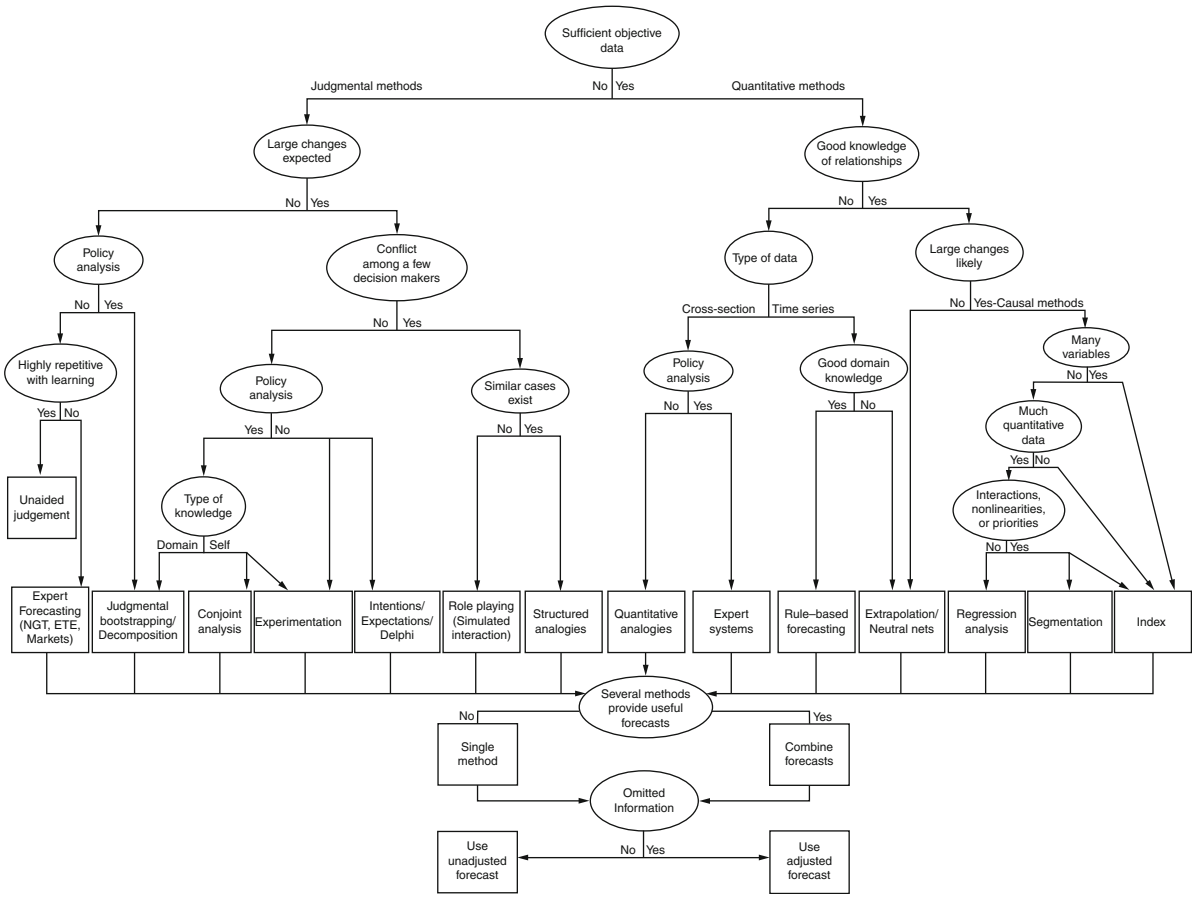
For situations in which some causal variables are much more important than others or if knowing about one causal variable provides knowledge about other causal variables, the take-the-best heuristic (TTB) can be used to infer which of two alternatives is most likely to occur (Gigerenzer and Goldstein 2006). Czerlinksi, Gigerenzer, and Goldstein (1999) compared take-the-best to models estimated using multiple regression and unit-weight models for 20 prediction problems (e.g. forecasting high school drop-out rates, mortality in U.S. cities, salaries of college professors, and obesity among children) for which the number of variables varied between 3 and 19. TTB out-of-sample forecasts were more accurate than forecasts from unit-weight models and forecasts from multiple regression models. Graefe and Armstrong (2012) used TTB to predict the outcomes of the ten U.S. presidential elections from 1972 to 2008 based on information about how well voters believed the candidates would deal with the most important issue facing the country. The TTB model forecasts were similarly accurate to forecasts from methods that incorporate substantially more information, including econometric models and prediction markets.

Segmentation can be applied when a heterogeneous whole can be divided into homogenous parts of roughly equal importance that respond to changes in different ways, and that can be forecast more accurately than the whole. For example, in the airline industry, price changes have different effects on business and personal travelers.

Segmentation is useful when large changes are expected. It is more appropriate than regression analysis when the situation involves interactions among variables, non-linear effects, or causal priorities. Segmentation does, however, require good prior knowledge about the problem and very large sample sizes. Use a priori analysis to identify causal variables and causal priorities. Specify the cut-points (divisions) that will define the segments using a priori analysis in conjunction with inspection of the data. Appropriate forecasting methods should be used to forecast the population and the behavior of individual segments.

Experts prefer the bottom-up approach for segmentation because it allows them to more effectively use their knowledge about the problem. Segmentation is also advantageous because the forecasting errors in the different segments may offset one another. For example, assume that there are ten divisions in a company. Accuracy might be improved by forecasting each division separately, then adding the forecasts. But caution is in order: if the samples of data for the segments are small and the data are erratic, the forecasts might contain very large errors.

Favorable results were obtained in a number of comparative studies on segmentation Armstrong (1985, p. 286–287 and 412–420). Dangerfield and Morris (1992), in their study on bottom-up forecasting, found that forecasts from segmentation were more accurate than global forecasts for 74% of 192 monthly time series from the M-Competition. In a study involving seven teams making estimates of the



Forecasting, Fig. 3 Selection Tree for Forecasting Methods

time required to complete two software projects, Jørgensen (2004) found that the error from the bottom-up forecasts was 51% less than that for the top-down approach.

Selection of Methods

The Forecasting Method Selection Tree, shown in Fig. 3, provides guidance on selecting the best forecasting methods for a given problem. The Tree has been derived from evidence-based principles. Users of this Tree must answer questions about the availability of data and the state of knowledge about the situation for which forecasts are required.

The first question is whether sufficient objective data are available to perform statistical analyses. If not, one must use judgmental methods.

In deciding among judgmental procedures, one must assess whether the future is likely to be substantially different from the past, whether policy analysis is needed and whether the situation involves decision makers who have conflicting interests. Other considerations are whether forecasts are made for recurrent and well-known problems, whether domain knowledge is available, and whether information about similar problems is available.

If much objective data are available and it is therefore possible to use quantitative methods, the forecaster has to determine first whether there is useful knowledge about causal relationships, whether cross-sectional or time-series data are available, and whether large changes are involved.

For situations about which there is little empirical knowledge about relationships, one needs to assess whether policy analysis is required and whether there is expert domain knowledge about the situation.

If there is good prior knowledge about empirical relationships and the future can be expected to differ substantially from the past, the number of variables and

the presence or absence of interactions among them, and the number of observations determine which causal method can be used. For example, regression models that rely on non-experimental data can typically use no more than 3 or 4 variables – even with massive sample sizes. For problems involving many causal variables, variable weights should not be estimated from the dataset (Dana and Dawes 2004). Instead one should to draw on independent sources of evidence (such as empirical studies and experts' domain knowledge) for assessing the impact of each variable on the outcome.

The Forecasting Method Selection Tree provides guidance, but on its own the guidance is not comprehensive. Forecasters may have difficulty identifying which conditions apply to the situation they wish to forecast. In that case, use different methods that draw on different information, and combine the forecasts according to rules determined before the forecasts were made. Imagine you have two forecasts of a quantity: how big would the error of the second forecast need to be in order for the error of the average of the two forecasts to be larger than the error of the first forecast alone? The answer is that the error of the second forecast would need to be bigger than the error of the first forecast if the error was in the same direction (sign) and more than three times the error of the first forecast if the error had the opposite sign.

To increase the likelihood that two forecasts bracket the true value, use forecasts that differ substantially. Batchelor and Dua (1995) found that the extent and probability of error reduction through combining were higher the greater the differences in the underlying theory or method that produced the component forecasts. For example, when combining real GNP forecasts, combining the 5% of forecasts that were most similar in their underlying theory reduced the error compared to the typical forecast by 11%. By comparison, combining the 5% of forecasts that were most diverse in their underlying theory yielded an error reduction of 23%.

Simple averages are a good starting point for combining forecasts. Differential weights should only be used if there is strong evidence about the relative accuracy of forecasts from the different methods. The relative accuracy of forecasts from different methods depends on the conditions. For example, in situations where uncertainty was high, trend extrapolation should be weighted less heavily and naïve extrapolation more heavily (Collopy and Armstrong 1992).

A meta-analysis of 30 studies found that combined forecasts (typically averages of different forecasts from a single type of method) yielded a 12% reduction in error compared to the average error of the components. The reductions of forecast error ranged from 3 to 24%. In addition, the combined forecasts were often more accurate than the most accurate component (Armstrong 2001c). Studies since that meta-analysis suggest that under favorable conditions (i.e., when forecasts are made for an uncertain situation, and many forecasts are available from several valid methods and different data sources) combining reduces errors almost by half (Graefe et al. 2011). Combining forecasts is especially useful if the forecaster wants to avoid large errors and if there is uncertainty about which method will be most accurate.

The final issue addressed in the Selection Tree is whether important information has been omitted in the process of deriving a forecast. One should consider adjustments to the forecasts only when the data do not fully reflect recent events, or experts know about events or changes to come, or key variables were not included in the forecasting process. If one or more of these conditions are met, the forecaster should provide written instructions on how the forecasts will be adjusted, solicit written adjustments from diverse experts, ask for adjustments before the experts see the forecasts, and record reasons for the adjustments (Goodwin 2005).

Judgmental adjustments should be avoided with cross-sectional forecasts such as who will be the most useful employee or whether to operate on a patient. Meehl (1956) summed up the evidence on making predictions about people as follows "... the first rule to follow... is to carefully avoid talking to him, and the second rule is to avoid thinking about him." Lewis (2003) described how Billy Bean, the manager of the Oakland Athletics baseball team followed this advice by contracting players to fit the needs of the team using only performance statistics. The result was a team that won games. Other teams have decided that to be competitive they must also use such methods.

Measuring Accuracy

Use error measures that are relevant to the decision. Ideally, error measures should allow for comparing the benefits from improved accuracy with the costs for obtaining the improvement. This can be difficult to assess in some situations, so one might simply use

the method or methods that provide the most accurate forecasts for the situation. Useful error measures include Mean Absolute Deviation (MAD), Mean (or Median) Absolute Percentage Error (MAPE or MdAPE), and Median Relative Absolute Error (MdRAE). The latter is calculated by dividing the absolute forecast error for a proposed model by the corresponding error for the random walk. The random walk is a simple and easy to understand benchmark model in forecasting. It uses the latest observation as the forecast for all periods in the forecast horizon. These measures are calculated as the mean or median of the following statistics:

$$AD = |F - A| \qquad APE = \left|\frac{F - A}{A}\right| \qquad RAE = \left|\frac{F - A}{F_{rw} - A}\right|$$

where F is the forecast, A is the actual value, and rw is the random walk method.

The selection of a measure depends upon the purpose of the analysis. For example, when making comparisons of accuracy across a set of time series, it is important to control for scale, the relative difficulty of forecasting each series, and the number of forecasts being examined (Armstrong and Collopy 1992).

Two measures of error should be avoided. The first of these, R^2 (which assesses the pattern of the forecasts relative to that of the actual data), is not particularly useful to forecasters and its use does more harm than good when choosing a model to forecast time-series data. The second measure to avoid, Mean Square Error, should not be used because it is unreliable and difficult to explain to decision makers (Armstrong 2001d).

Assessing Forecast Uncertainty

In addition to improving accuracy, forecasting is also concerned with assessing uncertainty. Early approaches to assessing uncertainty used measures of how well forecasts fit historical data as a way to infer forecast uncertainty. This approach can in some cases provide reasonable approximations of prediction intervals for forecasts based on models derived using cross-sectional data, such as with forecasts of how much a house will sell for. However, for time-series data, the historical fit typically leads to prediction intervals that are too narrow. Some empirical studies have shown that over half of actual outcomes are outside the 95% confidence intervals estimated using historical data (Makridakis et al. 1987).

The best approach to assessing forecast uncertainty is to simulate the forecasting situation as closely as possible. Thus, to determine how well one can forecast two years into the future, one examines a sample of ex ante two-year-ahead forecasts and the corresponding actual values. Ex ante means that one looks as if from before and does do not use knowledge about the situation after the starting point for forecasting. Chatfield (2001) describes proper procedures for estimating prediction intervals.

Use of Forecasts

Whether a forecast is used or not depends not only on the intrinsic merit of the forecast, but also the willingness of decision-makers to accept it. Expert reasoning can be persuasive for decision makers. Nevertheless, accurate forecasts are often ignored, particularly when they involve bad news or imply that change is necessary. If possible, one should ask the decision maker to agree on the methods that will be used and to commit to accept forecasts from the agreed upon process.

Scenarios, which are detailed stories about what happened in the future, can increase the chances a forecast will be used by making the forecast seem likely to occur. Techniques for scenario writing include using concrete examples, showing a logical sequence of causal events, using past tense, and having decision makers describe how they would have acted in each scenario (Gregory and Duran 2001).

Concluding Remarks

Progress in forecasting has been due primarily to empirical testing of alternative approaches. Forecasting research has begun to address the conditions under which different methods are most useful. These research strategies have provided many useful findings that have been summarized as principles. The principles can help to improve accuracy, assess uncertainty, and gain the acceptance of forecasts. Often research findings conflict with the expectations of statisticians and decision makers. The conflicts between standard practice on the one hand and empirical findings on the other have meant that practitioners and academics have often been slow to adopt new methods and improved procedures. That state of affairs means that for those who are willing

to use the findings of forecasting research will have many opportunities to improve forecasting and decision-making.

See

- ▶ Delphi Method
- ▶ Expert Systems
- ▶ Exponential Smoothing
- ▶ Regression Analysis

References

Allen, G., & Fildes, R. (2001). Econometric forecasting. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 303–362). Boston: Kluwer Academic Publishers.

Armstrong, J. S. (1985). *Long-range forecasting*. New York: John Wiley.

Armstrong, J. S. (2001a). Judgmental bootstrapping: Inferring experts' rules for forecasting. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 171–192). Boston: Kluwer Academic Publishers.

Armstrong, J. S. (2001b). Extrapolation for time-series and cross-sectional data. In J. S. Armstrong, (Ed.), *Principles of forecasting* (pp. 217–243). Kluwer Academic Publishers.

Armstrong, J. S. (2001c). Combining forecasts. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 417–440). Boston: Kluwer Academic Publishers.

Armstrong, J. S. (2006a). Findings from evidence-based forecasting: Methods for reducing forecast error. *International Journal of Forecasting, 22*, 583–598.

Armstrong, J. S. (2006b). How to make better forecasts and decisions: Avoid face-to-face meetings. *Foresight – The International Journal of Applied Forecasting*. (5), 3–8.

Armstrong, J.S. (2001d). Evaluating forecasting methods. In J. S. Armstrong, (Ed.), *Principles of Forecasting* (pp. 443–472). Boston: Kluwer Academic Publishers.

Armstrong, J. S., & Collopy, F. (1992). Error measures for generalizing about forecasting methods: Empirical comparisons. *International Journal of Forecasting, 8*, 69–80.

Armstrong, J. S., Collopy, F., & Yokum, T. (2005). Decomposition by causal forces: A procedure for forecasting complex time series. *International Journal of Forecasting, 21*, 25–36.

Armstrong, J. S., & Graefe, A. (2011). Predicting elections from biographical information about candidates: A test of the index method. *Journal of Business Research, 64*, 699–706.

Batchelor, R., & Dua, P. (1995). Forecaster diversity and the benefits of combining forecasts. *Management Science, 41*, 68–75.

Bunn, D. W., & Vassilopoulos, A. I. (1999). Comparison of seasonal estimation methods in multi-item short-term forecasting. *International Journal of Forecasting, 15*, 431–443.

Chatfield, C. (2001). Prediction intervals for time-series forecasting. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 475–494). Boston: Kluwer Academic Publishers.

Collopy, F., Adya, M., & Armstrong, J. S. (2001). Expert systems for forecasting. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 285–300). Boston: Kluwer Academic Publishers.

Collopy, F., & Armstrong, J. S. (1992). Rule-based forecasting: Development and validation of an expert systems approach to combining time series extrapolations. *Management Science, 38*, 1394–1414.

Cuzán, A. G., & Bundrick, C. M. (2009). Predicting presidential elections with equally weighted regressors in Fair's equation and the fiscal model. *Political Analysis, 17*, 333–340.

Czerlinski, J., Gigerenzer, G., & Goldstein, D. G. (1999). How good are simple heuristics? In G. Gigerenzer, P. M. Todd & The ABC Research Group (Eds.), *Simple heuristics that make us smart* (pp. 97–118) New York: Oxford University Press.

Dana, J., & Dawes, M. D. (2004). The superiority of simple alternatives to regression for social science predictions. *Journal of Educational and Behavioral Statistics, 29*, 317–331.

Dangerfield, B. J., & Morris, J. S. (1992). Top-down or bottom-up: Aggregate versus disaggregate extrapolations. *International Journal of Forecasting, 8*, 233–241.

Gigerenzer, G., & Goldstein, D. G. (1996). Reasoning the fast and frugal way: Models of bounded rationality. *Psychological Review, 103*, 650–669.

Goodwin, P. (2005). How to integrate management judgment with statistical forecasts. *Foresight – The International Journal of Applied Forecasting*, (1) 8–12.

Gorr, W., Olligschlaeger, A., & Thompson, Y. (2003). Short-term forecasting of crime. *International Journal of Forecasting, 19*, 579–594.

Graefe, A., & Armstrong, J. S. (2012). Predicting elections from the most important issue: A test of the take-the-best heuristic. *Journal of Behavioral Decision Making, 25*, 41–48.

Graefe, A., Armstrong, J. S., Jones, R. J., & Cuzán, A. G. (2011). *Combining forecasts: An application to election forecasts*. The Wharton School, University of Pennsylvania.

Green, K. C. (2005). Game theory, simulated interaction, and unaided judgement for forecasting decisions in conflicts: Further evidence. *International Journal of Forecasting, 21*, 463–472.

Green, K. C., & Armstrong, J. S. (2007). Structured analogies for forecasting. *International Journal of Forecasting, 23*, 365–376.

Green, K. C., Armstrong, J. S., & Graefe, A. (2007). Methods to elicit forecasts from groups: Delphi and prediction markets compared. *Foresight – The International Journal of Applied Forecasting*, (8), 17–20.

Gregory, W. L., & Duran, A. (2001). Scenarios and acceptance of forecasts. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 519–540). Boston: Kluwer Academic Publishers.

Harvey, N. (2001). Improving judgment in forecasting. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 59–80). Boston: Kluwer Academic Publishers.

Herzog, S. M., & Hertwig, R. (2009). The wisdom of many in one mind: Improving individual judgments with dialectical bootstrapping. *Psychological Science, 20*, 231–237.
Jørgensen, M. (2004). Top-down and bottom-up expert estimation of software development effort. *Journal of Information and Software Technology, 46*, 3–16.
Kim, M., & Hunter, J. E. (1993). Relationships among attitudes, behavioral intentions, and behavior: A meta-analysis of past research. *Communication Research, 20*, 331–364.
Lewis, M. (2004). *Moneyball: The art of winning an unfair game*. New York: W. W. Norton & Company.
MacGregor, D. G. (2001). Decomposition in judgmental forecasting and estimation. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 107–124). Boston: Kluwer Academic Publishers.
Makridakis, S., et al. (1982). The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Journal of Forecasting, 1*, 111–153.
Makridakis, S., Hibon, M., Lusk, E., & Belhadjali, M. (1987). Confidence intervals. *International Journal of Forecasting, 3*, 489–508.
Makridakis, S., Wheelwright, S., & Hyndman, R. J. (1998). *Forecasting methods and applications*. New York: John Wiley.
Meehl, P. E. (1956). Wanted: A good cookbook. *American Psychologist, 11*, 263–272.
Miller, D. M., & Williams, D. (2003). Shrinkage estimators of time series seasonal factors and their effect on forecasting accuracy. *International Journal of Forecasting, 19*, 669–684.
Miller, D. M., & Williams, D. (2004). Shrinkage estimators for damping X12-ARIMA seasonals. *International Journal of Forecasting, 20*, 529–549.
Morwitz, V. G. (2001). Methods for forecasting from intentions data. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 33–56). Boston: Kluwer Academic Publishers.
Ord, K., Hibon, M., & Makridakis, S. (2000). The M-3 competition. *International Journal of Forecasting, 16*, 433–436.
Rowe, G., & Wright, G. (2001). Expert opinions in forecasting: The role of the Delphi technique. In J. S. Armstrong (Ed.), *Principles of forecasting* (pp. 125–144). Boston: Kluwer Academic Publishers.
Van de Ven, A. H., & Delbecq, A. L. (1974). The effectiveness of nominal, Delphi, and interacting group decision making processes. *Academy of Management Journal, 17*, 605–621.
Woudenberg, F. (1991). An evaluation of Delphi. *Technological Forecasting and Social Change, 40*, 131–150.
Wright, M., & MacRae, M. (2007). Bias and variability in purchase intention scales. *Journal of the Academy of Marketing Science, 355*, 617–624.

Forward Chaining

An approach to reasoning in which an inference engine determines the effect of current known variable values on unknown variables by firing all rules whose premises can be established as being true.

See

- Artificial Intelligence
- Expert Systems

Forward Kolmogorov Equations

In a continuous time Markov chain $\{X(t)\}$, define $p_{ij}(t)$ as the probability that $X(t+s) = j$, given that $X(s) = i$, for $s, t \geq 0$, and r_{ij} as the transition rate out of state i to state j. Then Kolmogorov's forward equations say that, for all states i,j and times $t \geq 0$, $dp_{ij}(t)/dt = \sum_{k \neq j} r_{kj}\, p_{ik}(t) - v_j\, p_{ij}(t)$, where v_k is the transition rate out of state k, $v_k = \sum_j r_{kj}$.

See

- Markov Chains
- Markov Processes

Forward-Recurrence Time

Suppose events occur at epochs $T_1, T_2, \ldots$ such that the interevent times $T_k - T_{k-1}$ are IID positive random variables. Then the forward recurrence time from an arbitrary time t is the time from t to the next occurrence.

See

- Point Stochastic Processes
- Renewal Process

Fourier Transform

For any function $f(t)$, its Fourier transform is defined as $\widehat{f}(s) = \int_{-\infty}^{\infty} e^{-2\pi i s t} f(t)dt$, which is equal to $E[e^{-2\pi i s X}]$ if $f(t)$ is a probability density function for random variable X, where i denotes the imaginary number $\sqrt{-1}$.

Fourier-Motzkin Elimination Method

A computational procedure for solving a system of linear inequalities.

Fractional Programming

Siegfried Schaible
University of California, Riverside, CA, USA

Introduction

Certain decision problems in OR/MS, as well as other extremum problems, give rise to the optimization of ratios. Constrained ratio optimization problems are commonly called fractional programs. They may involve more than one ratio in the objective function.

One of the earliest fractional programs (though not called so) is an equilibrium model for an expanding economy in which the growth rate is determined as the maximum of the smallest of several output–input ratios (von Neumann 1937, 1945). Since then, but mostly after the classical paper by Charnes and Cooper (1962), some nine hundred publications have appeared in fractional programming; for comprehensive bibliographies, see Schaible (1982, 1993). Monographs solely devoted to fractional programming include Schaible (1978) and Craven (1988).

Almost from the beginning, fractional programming has been discussed in the broader context of generalized concave programming. Ratios, though not concave in general, are often still generalized concave in some sense. An introduction to fractional programming in this context is Avriel et al. (1988).

Notation and Definitions

Suppose f, g and h_j $(j = 1,\ldots, m)$ are real-valued functions which are defined on the subset X of the n-dimensional Euclidean space $\Re^n$ and let $\boldsymbol{h} = (h_1,\ldots, h_m)^T$ where T denotes the transpose. The ratio considered is given by

$$q(\mathrm{x}) = f(\mathrm{x})/g(\mathrm{x}) \tag{1}$$

over the set

$$S = \{x \in X | h(x) \leq 0\} \tag{2}$$

assuming $g(\boldsymbol{x}) > \boldsymbol{0}$ on X. The nonlinear program

$$\sup\{q(x)|x \in S\} \tag{3}$$

is called a (single-ratio) fractional program.

In addition, the following three types of multi-ratio fractional programs are of interest:

$$\sup\left\{\sum_{i=1}^{p} q_i(x)|x \in S\right\}, \tag{4}$$

$$\sup\left\{\min_{1\leq i\leq p.} q_i(x)|x \in S\right\}, \tag{5}$$

and the multi-objective fractional program

$$\sup_{x\in S}\{q_1(x), \ldots, q_p(x)\}. \tag{6}$$

Here $q_i(\boldsymbol{x}) = f_i(\boldsymbol{x})/g_i(\boldsymbol{x})$ $(i = 1,\ldots, p)$ when f_i and g_i are real-valued functions on X with $g_i(\boldsymbol{x}) > 0$. Problem (5) is often referred to as a generalized fractional program (Schaible and Ibaraki 1983).

The focus in fractional programming is the objective function and not the feasible region S. As in most publications, it is assumed that the h_j are convex functions on the convex domain X yielding a convex feasible region S.

Most of the theory for fractional programs (3)–(6) is developed under the assumption that ratios satisfy the following concavity/convexity condition: f is concave and g is convex on the convex set X (f is to be nonnegative if g is not affine-linear, as below). Such problems are called concave fractional programs. It is to be noted, however, that the objective function in these problems is not concave in general; hence they are not concave programs. Problem (3) is called a quadratic fractional program if f and g are quadratic functions and S is a convex polyhedron.

A special case is the linear fractional program where f and g are affine-linear functions and S is a convex polyhedron

$$\sup\{(\boldsymbol{c}^T\boldsymbol{x} + \alpha)/(\boldsymbol{d}^T\boldsymbol{x} + \beta)|\boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}\}. \tag{7}$$

Here $\boldsymbol{c}, \boldsymbol{d} \in \mathfrak{R}^n$, $\boldsymbol{b} \in \mathfrak{R}^m$, $\alpha, \beta \in \mathfrak{R}$, $\boldsymbol{A}$ is an $m \times n$ matrix and the denominator is positive on the feasible region. It will be seen below that linear and concave fractional programs have still many properties in common with linear and concave programs.

Single-Ratio Fractional Programs

The following types of single-ratio fractional programming applications can be found in the literature: economic, non-economic and indirect applications.

Economic Applications — The efficiency of a system is sometimes characterized by a ratio of economical and/or technical terms. Then, maximizing the efficiency leads to a fractional program. Examples of such ratios are: profit/capital, profit/revenue, cost/volume, productivity, relative usage of material, return/cost, return/risk, expected cost/beta-index, (expected) cost/time, profit/time, liquidity, earnings per share, dividend per share, weighted outputs/weighted inputs, income/(investment + consumption), mean/standard deviation.

Such ratios arise in resource allocation, transportation, production, maintenance, inventory, finance, data envelopment analysis, and macroeconomics for example. No longer are these rates merely used to control past economic behavior. Instead, the optimization of rates is getting more attention in decision making for future projects. Depending on the form of the functions in the numerator and denominator, many of the ratio optimization problems above are linear, quadratic or concave fractional programs.

Non-economic Applications — In information theory, the capacity of a communication channel can be defined as the maximal transmission rate, thus giving rise to a (nonquadratic) concave fractional program. In numerical analysis, the eigenvalue problem can be reduced to constrained maximization of the Rayleigh quotient, and hence, leads to a (nonconcave) quadratic fractional program. In physics, maximization of the signal-to-noise ratio gives rise to a concave quadratic fractional program.

Indirect Applications — Fractional programs may also arise in the process of solving other optimization problems involving no ratios. Examples are: subproblems in large-scale mathematical programming, deterministic substitutes in stochastic mathematical programming, subproblems in nondifferentiable convex programming, problems in connection with interior-point methods for linear programming, dual location problems, approximations to numerically intractable portfolio selection problems, bounds on the trauma outcome function for emergency medical facilities.

Depending on the original optimization problem, often linear, quadratic or concave fractional programs are encountered.

Properties

Concave fractional programs have the following properties (Avriel et al. 1988):

Proposition 1: A local maximum is a global maximum since the objective function $q(\boldsymbol{x}) = f(\boldsymbol{x})/g(\boldsymbol{x})$ is semi-strictly quasiconcave.

Proposition 2: A maximum is unique if the numerator $f(\boldsymbol{x})$ is strictly concave or the denominator $g(\boldsymbol{x})$ is strictly convex since in this case the objective function is strictly quasiconcave.

Proposition 3: In case of differentiable functions $f(\boldsymbol{x})$, $g(\boldsymbol{x})$, $h(\boldsymbol{x})$, a solution of the Karush-Kuhn-Tucker conditions is a maximum since the objective function $q(\boldsymbol{x})$ is pseudoconcave.

For **linear fractional programs**, the following additional property holds:

Proposition 4: A maximum is attained at a vertex in case of a (nonempty) bounded feasible region S, since the objective function is quasiconvex (in addition to quasiconcave).

Concave and linear fractional programs share not only the above properties with concave and linear programs, respectively, but they can also be related to these programs through transformations. The first transformation below changes the variables, whereas the second one maintains the same variables, but requires a parameter in the transformed problem.

Introducing the new variables

$$y = [1/g(x)]x, \quad t = 1/g(x), \tag{8}$$

It can be shown (Schaible 1976):

Proposition 5: A concave fractional program (3) with an affine-linear denominator can be reduced to the concave program

$$\sup\{tf(y/t)|th(y/t) \le 0, tg(y/t) = 1, \quad y/t \in X, t > 0\}. \tag{9}$$

If $g(\boldsymbol{x})$ is not affine-linear, an equivalent concave program is obtained for (9) by relaxing the equality in (9) to $tg(\boldsymbol{y/t}) \le 1$.

In the special case of a linear fractional program (7), the equivalent concave program (9) becomes the linear program

$$\sup\{c^T y + \alpha t|Ay - bt \le 0, d^T y + \beta t = 1, \quad y \ge 0, t > 0\} \tag{10}$$

where $t > 0$ can be replaced by $t \ge 0$ if (7) has an optimal solution. The equivalence between (7) and (10) was first established by Charnes and Cooper (1962).

In the second transformation, variables and the feasible region are maintained and a parameter is introduced to separate numerator and denominator (Dinkelbach 1967). Consider

$$\sup\{f(x) - \lambda g(x)|x \in S\}, \ \lambda \in \Re \text{ parameter}. \tag{11}$$

If (3) is a concave, linear, or quadratic fractional program, then (11) is a parametric concave, linear, or quadratic program, respectively.

Suppose $f(\boldsymbol{x})$, $g(\boldsymbol{x})$ are continuous and S is a (non-empty) compact set. Then:

Proposition 6: Problems (3) and (11) have the same optimal solutions where $\lambda = \bar{\lambda}$ is the unique zero of the strictly decreasing, continuous function

$$F(\lambda) = \sup\{f(x) - \lambda g(x)x \in S\}. \tag{12}$$

Turning now to duality for fractional programs, it is noted that standard concave programming duality relations are no longer true in case of concave, or even linear-fractional programs. However, Proposition 5 can be used to introduce duality via the equivalent concave program (9) (Schaible 1976). For a detailed presentation of various duality approaches as well as their use in sensitivity analysis, see Schaible (1978), Avriel et al. (1988), and Craven (1988).

The properties of concave and linear fractional programs in Proposition 1–6 allow for at least four different solution strategies (Martos 1975; Schaible and Ibaraki 1983; Craven 1988):

a. direct solution of the quasiconcave (pseudoconcave) program (3);
b. solution of the equivalent concave (linear) program (9);
c. solution of the dual of (9);
d. solution of the parametric concave (linear) program (11).

In the case of d), rather than applying parametric programming techniques, the iterative method *by* Dinkelbach (1967) can be used. It turns out to be equivalent to Newton's classical method for finding the zero λ of $F(\lambda)$ in (12). Various modifications and computational results were discussed in Schaible and Ibaraki (1983).

Multi-ratio Fractional Programs

Maximizing the Sum of Ratios — Problem (4) arises naturally in decision making when several of the rates above are to be optimized and a compromise is sought that optimizes the weighted sum of these rates. Other applications of this model were given by Schaible (1990).

Unfortunately, none of the above properties of concave fractional programs hold anymore if each ratio is a quotient of a concave and a convex function, even in the linear case. Only some preliminary theoretical and algorithmic results are known for this important, but difficult problem (Craven 1988; Schaible 1990).

Maximizing the Smallest of Several Ratios — Apart from the economic equilibrium model by von Neumann (1937, 1945), problems in financial planning and fund allocation under equity considerations give rise to a generalized fractional program (5) (Schaible 1990). Furthermore, the same model is of interest in numerical mathematics in rational approximation involving the Chebyshev norm. In all these examples, the ratios are quotients of concave and convex functions.

Starting with von Neumann (1937, 1945), several authors have proposed a duality theory for concave generalized fractional programs (Avriel et al. 1988; Craven 1988; Schaible 1990). Though different approaches are employed, most duals coincide and are again a generalized fractional program. The objective function of the primal is semi-strictly quasiconcave and the one of the dual is semi-strictly quasi convex (Avriel et al. 1988). Thus a local optimal solution is global in both the primal and the dual. A duality theory has been established for these non-concave problems which is as rich as the one for concave and linear programs.

Concave generalized fractional programs can be solved in either the primal or the dual by an extension of Dinkelbach's algorithm. In case of more than one ratio, this is no longer identical with Newton's method. It gives rise to a sequence of concave (linear) programs as subproblems. The convergence properties are well analyzed. Computational results for various modifications of this algorithm have been encouraging (Schaible 1990).

Multi-Objective Fractional Programs — Problem (6) arises when several rates above are to be maximized simultaneously and, in contrast to the problems in (4) and (5), a unifying objective function is not considered; instead, the decision maker is to be provided with the set of efficient alternatives, that is, all those feasible solutions for which none of the rates can be increased without decreasing another rate. Some theoretical results are known for (6) in case of ratios of concave and convex functions including duality relations (Craven 1988; Schaible 1990). Also, the connectedness of the set of efficient alternatives has been established under limiting assumptions. Additional theoretical and algorithmic results are known for the case of two ratios (Schaible 1990).

Concluding Remarks

Concave single-ratio fractional programs, as well as concave generalized fractional programs, have been analyzed quite successfully, both theoretically and algorithmically. More research needs to be done for the nonconcave case, sum-of-ratios problem (4), and multi-objective fractional programming (6).

See

- ▶ Data Envelopment Analysis
- ▶ Linear Programming
- ▶ Multiobjective Programming
- ▶ Nonlinear Programming
- ▶ Quadratic Programming

References

Avriel, M., Diewert, W. E., Schaible, S., & Zang, I. (1988). *Generalized concavity*. New York: Plenum.

Bajalinov, E. B. (2003). *Linear-fractional programming: Theory, methods, applications and software*. Norwell, MA: Kluwer.

Charnes, A., & Cooper, W. W. (1962). Programming with linear fractional functionals. *Naval Research Logistics Quarterly, 9*, 181–186.

Craven, B. D. (1988). *Fractional programming*. Berlin: Heldermann Verlag.

Dinkelbach, W. (1967). On nonlinear fractional programming. *Management Science, 13*, 492–498.

Martos, B. (1975). *Nonlinear programming: Theory and methods*. Amsterdam: North-Holland.

Schaible, S. (1976). Duality in fractional programming: A unified approach. *Operations Research, 24*, 452–461.

Schaible, S. (1978). *Analyse und anwendungen von quotientenprogrammen, mathematical systems in economics 42*. Meisenheim: Hain-Verlag.

Schaible, S. (1982). Bibliography in fractional programming. *Zeitschrift für Operations Research, 26*, 211–241.

Schaible, S. (1990). Multi-ratio fractional programming — analysis and applications. In P. Mazzoleni (Ed.), *Proceedings of 13th Annual Conference of Associazione per la Matematica Applicata alle Scienze Economiche e Sociali, Verona/Italy, September 1989* (pp. 47–86). Bologna: Pitagora Editrice.

Schaible, S. (1993). Fractional programming. In R. Horst & P. Pardalos (Eds.), *Handbook of global optimization*. Dordrecht: Kluwer Academic Publishers.

Schaible, S., & Ibaraki, T. (1983). Fractional programming. *European Journal of Operational Research, 12*, 325–338.

von Neumann, J. (1937). Über ein ökonomisches Gleichungssystem und eine Verallgemeinerung des Brouwerschen Fixpunktsatzes. In K. Menger (Ed.), *Ergebnisse eines mathematischen Kolloquiums 8* (pp. 73–83). Leipzig and Vienna.

von Neumann, J. (1945). A model of general economic equilibrium. *Review of Economic Studies, 13*, 1–9.

Fractional-Programming Problem

- ▶ Fractional Programming

Framing

Refers to how a problem is presented to decision makers, or how they formulate it in their minds.

See

- ▶ Choice Theory
- ▶ Decision Analysis in Practice

Frank-Wolfe Method

- ▶ Quadratic Programming

Free Float

The amount of time a designated activity can be delayed without affecting succeeding activities of a project. This will be the float for the final activity of a chain, or for a single activity which does not lie on a chain with equal float.

See

- ▶ Float
- ▶ Network Planning

Free Variable

A variable that can take on any value, as contrasted to a variable that must take on nonnegative values. In a linear-programming problem, a variable that is free can be expressed as the difference between two nonnegative variables. However, when using the simplex method to solve a linear-programming problem with free variables, it is more effective to eliminate those variables by means of constraints in which they appear.

See

- ▶ Unrestricted Variable

Freight Routing

The itinerary of a shipment through a logistics network.

Ftran

The procedure for computing the updated version of the entering column in a simplex iteration, when the *LU* factors of the basis matrix are given in product form. The name FTRAN (forward transformation) derives from the fact that the eta file is scanned forward in the process.

Fuzzy Sets, Systems, and Applications

Costas P. Pappis[1], Constantinos I. Siettos[2] and Thomas K. Dasaklis[1]
[1]University of Piraeus, Piraeus, Greece
[2]National Technical University of Athens, Athens, Greece

Introduction

In classical set theory, an element either does or does not belong to a set, being characterized by a membership in the set that may have one of two values: 1 or 0. Fuzzy sets generalize classical sets (in fuzzy set theory often called crisp sets) by allowing the gradual assessment of the memberships of elements in a set. Thus, by use of a membership function valued in the real unit interval [0, 1], each element is assigned a number in that interval, which measures its grade of membership in the set. Fuzzy systems are systems that are modeled using fuzzy sets. They have been widely used for both research and practical applications, even for industrial purposes.

Fuzzy logic provides a convenient way to build models, decision making systems and controllers, by incorporating qualitative knowledge and heuristics. These inherent characteristics of fuzzy logic offer a very attractive way of handling imprecision in the data and/or complex systems, where the derivation of an accurate model is difficult or even impossible.

Fuzzy sets and systems encompass artificial intelligence, information processing and theories from logic to pure and applied mathematics, such as graph theory, topology, control and optimization. The theory of fuzzy sets was introduced by Lotfi Zadeh (1965, 1973). In Zadeh (1973, p. 1), he stated, "... as the complexity of a system increases, our ability to make precise and yet significant statements about its behavior diminishes until a threshold is reached beyond which precision and significance (or relevance) become almost mutually exclusive characteristics."

Indeed, the derivation of mathematical models that can describe in an efficient manner real world problems is quite often overwhelming or even an impossible task due to the inherent ambiguity of characteristics that these problems may possess. The main advantage of fuzzy logic techniques over more conventional approaches in solving complex, nonlinear and/or ill-defined problems lies in their inherent capability of incorporating a priori qualitative knowledge and expertise about system behavior and dynamics.

This renders fuzzy logic systems almost indispensable for obtaining a more transparent and tactile qualitative insight for systems whose adequate representation with exact mathematical models is poor or impossible. Besides, fuzzy schemes can be used either as enabling to other approaches or as self-reliant methodologies providing thereby a plethora of alternative structures and schemes.

For systems involving nonlinearities and lack of a reliable analytical model, fuzzy logic control has emerged as one of the most promising approaches. Fuzzy inference is a step towards the simulation of human thinking. In fact, fuzzy systems generate nonlinear functions according to a representation theorem by Wang (1992), who stated that any continuous nonlinear function can be approximated as exactly as needed with a finite set of fuzzy variables, values and rules. Therefore, by applying appropriate design procedures, it is always possible to design a fuzzy controller that is suitable for the nonlinear system under control.

Fuzzy Sets, Systems, and Applications, Table 1 Critical points in the foundation years of fuzzy logic

First paper on fuzzy systems (Zadeh 1965)
Linguistic approach (Zadeh 1973)
Fuzzy Logic controller (Assilian and Mamdani 1975)
Table-Based Controller (Mamdani,1977)
Heat Exchanger based on fuzzy logic (Østergaard 1977)
Self-organizing fuzzy controller (Mamdani 1977; Procyk and Mamdani 1979)
Fuzzy logic control for cement production (Holmblad and Østergaard 1982)
Fuzzy controllers on Tokyo subway shuttles (Hitachi 1984)
Fuzzy Chip (Togai and Watanabe 1986)
Hardware implementation of fuzzy system (Yamakawa and Miki 1986)
Hybrid Neural-Fuzzy systems (Kosko 1992)

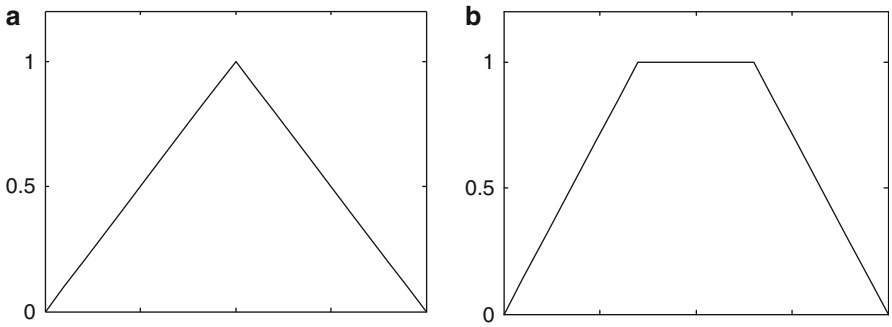
The applications of fuzzy logic have dramatically increased since 1990, ranging from cognitive and decision processes, engineering and industrial applications to systems control, economics and management (Karr and Gentry 1993; Sugeno and Yasukawa 1993; Østergaard 1977; 1990); robotics (Ruan et al. 2003); transportation (Chen et al. 2008); nuclear engineering (Kunsch and Fortemps 2002), medicine (Blanco et al. 2002; Kilic et al. 2002), economics (Gil-Lafuente 2005), see applications section for more. Table 1 depicts some benchmark results in the foundation years of fuzzy logic.

Research on the theory of fuzzy sets is directed towards various disciplines, including possibility theory (Cayrac et al. 1996), fuzzy operators (Pradera et al. 2002; Yager 2002a); fuzzy relations (Naessens et al. 2002; Pedrycz and Vasilakos 2002), measures of information and comparison (Hung 2002; Yager 2002b), non-classical logics (Biacino and Gerla 2002; Novak 2002), algebra (Di Nola et al. 2002); topology (Albrecht 2003).

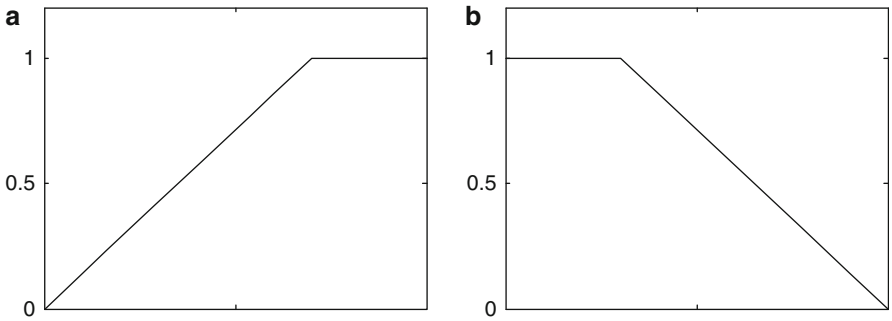
Fuzzy Set Theory

The notion of membership. The membership of an element x in a classical set A is given by:

$$\mu(\mathrm{x}) = \begin{cases} 1 \\ 0 \end{cases}$$



Fuzzy Sets, Systems, and Applications, Fig. 1 (**a**) Triangular; (**b**) Trapezoid membership function



Fuzzy Sets, Systems, and Applications, Fig. 2 (**a**) Monotonically increasing; (**b**) Monotonically decreasing linear membership function

Hence an element is assigned to a set A or not. That can be expressed as:

$$A \cap \bar{A} = \emptyset$$

where the symbol $\cap$ denotes intersection.

On the other hand, fuzzy logic is a logic based on truth-values that are numbers in the closed unit interval [0,1]. Fuzzy logic is thus based on fuzzy sets. A fuzzy set is a set consisting of members with a degree of membership, rather than being either members or not members. The function that ties a number, commonly in the [0,1] interval, to each element of a set (the universe of discourse) is called membership function.

Membership functions. Let X denote the universe of discourse. Every fuzzy set F in X is characterized completely by its membership function.

Definition: The membership function μ_F of a fuzzy set F in X is a function

$$\mu_F : \mathrm{X} \rightarrow [0, 1]$$

The most commonly used membership functions are the following (Dubois and Prade 1980; Zimmermann 1996): Triangular, Trapezoid, Linear, Sigmoidal, Π – type, Gaussian.

The **triangular** membership function (Fig. 1a) is defined as:

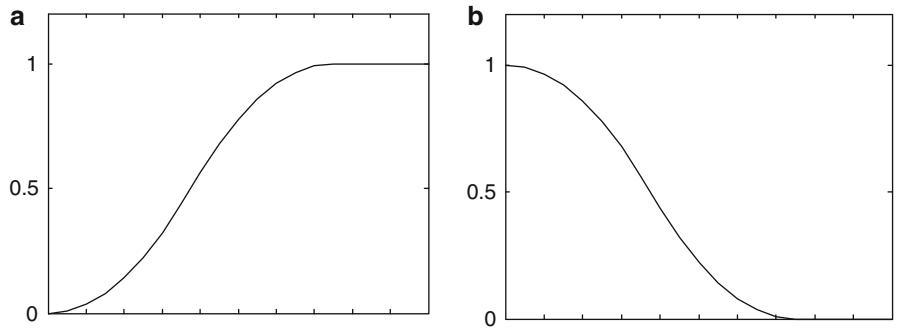
$$\mathrm{Tri}(x;\alpha,\beta,\gamma) = \begin{cases} 0 & x < \alpha \\ \frac{x-a}{\beta-\alpha} & \alpha \le x < \beta \\ -\frac{x-\gamma}{\gamma-\beta} & \beta \le x < \gamma \\ 0 & x \ge \gamma \end{cases}$$

The **trapezoid** membership function (Fig. 1b) is defined as:

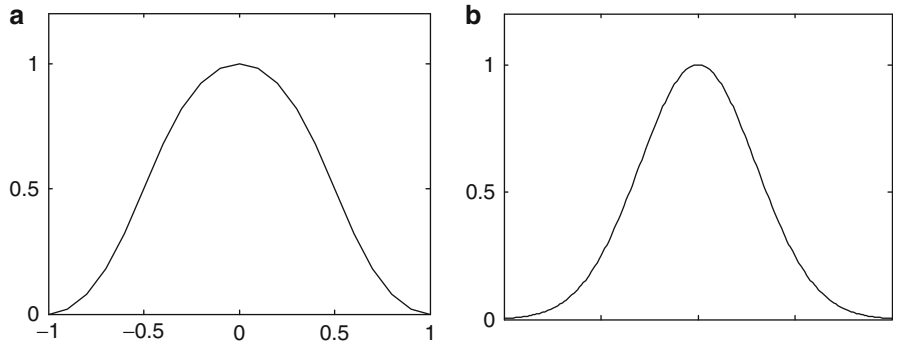
$$\mathrm{Tra}(x;\alpha,\beta,\gamma,\delta) = \begin{cases} 0 & x < \alpha \\ \frac{x-a}{\beta-\alpha} & \alpha \le x < \beta \\ 1 & \beta \le x < \gamma \\ -\frac{x-\delta}{\delta-\gamma} & \gamma \le x < \delta \\ 0 & x \ge \delta \end{cases}$$

The monotonically increasing linear membership function (Fig. 2a) is given by

$$\mathrm{L}(x;\alpha,\beta) = \begin{cases} 0 & x<\alpha \\ \frac{x-a}{\beta-\alpha} & \alpha \le x \le \beta \\ 1 & x > \beta \end{cases}$$



Fuzzy Sets, Systems, and Applications, Fig. 3 (**a**) Monotonically increasing; (**b**) Monotonically decreasing sigmoidal membership function



Fuzzy Sets, Systems, and Applications, Fig. 4 (**a**) Π; (**b**) Gaussian membership function

The monotonically decreasing linear membership function (Fig. 2b) is given by

$$\mathrm{L}(x;\alpha,\beta)=\begin{cases}0 & x<\alpha\\ -\frac{x-\beta}{\beta-\alpha} & \alpha\le x\le\beta\\ 1 & x>\beta\end{cases}$$

The monotonically increasing sigmoidal membership function (Fig. 3a) is given by

$$\mathrm{S}(x;\alpha,\beta,\gamma)=\begin{cases}0 & x<\alpha\\ 2\left(\frac{x-\alpha}{\gamma-\alpha}\right)^2 & \alpha\le x\le\beta\\ 1-2\left(\frac{x-\gamma}{\gamma-\alpha}\right)^2 & \beta\le x\le\gamma\\ 1 & x>\gamma\end{cases}$$

The monotonically decreasing sigmoidal membership function (Fig. 3b) reads:

$$\mathrm{S}(x;\alpha,\beta,\gamma)=\begin{cases}1 & x<\alpha\\ 1-2\left(\frac{x-\alpha}{\gamma-\alpha}\right)^2 & \alpha\le x\le\beta\\ 2\left(\frac{x-\gamma}{\gamma-\alpha}\right)^2 & \beta\le x\le\gamma\\ 0 & x>\gamma\end{cases}$$

The Π- membership function (Fig. 4a) is defined as

$$\Pi(x,\alpha,\beta,\gamma)=\begin{cases}S(x;\gamma-\beta,\frac{\gamma-\beta}{2},\gamma) & x\le\gamma\\ 1-\mathrm{S}(x;\gamma-\beta,\frac{\gamma+\beta}{2},\gamma+\beta) & x\le\gamma\end{cases}$$

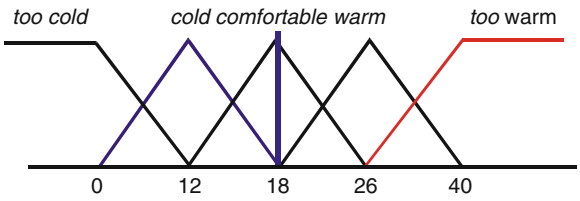
The Gaussian membership function (Fig. 4b) is given by

$$\mathrm{G}(x;k,\gamma)=\exp(-(\gamma-x)^2/2\sigma^2)$$

where σ is the standard deviation

As an example, the room temperature for low-level work activities could be described by the following 5 fuzzy sets characterized by triangular or trapezoid membership functions, where a temperature around 18 °C is a comfortable one, around 26° a warm one (though not during summer!) while above 40 °C is definitely too warm, and around 12 °C can be characterized as a cold while below that too cold (Fig. 5).

Fuzzy logical operations. Fuzzy theory set operations are of utmost importance to the better understanding and design of fuzzy systems. Below the most basic fuzzy set operations are presented, which are defined with respect to their corresponding membership functions.



Fuzzy Sets, Systems, and Applications, Fig. 5 Fuzzy representation of a room temperature

Equality: Two fuzzy sets A and B are ***equal*** on the universe of discourse X if their membership functions are equal for each $x \in X$ iff

$$\forall x \in X : \mu_A(x) = \mu_B(x).$$

Subset: A fuzzy set A is a ***subset*** of B $(A \subseteq B)$ iff

$$\forall x \in X : \mu_A(x) \leq \mu_B(x).$$

Intersection: For the operation of intersection $\cap$ of two fuzzy sets A and B, there is a plethora of definitions in the bibliography. The choice is application dependant.

$$\forall x \in X \mu_{A\cap B} = \left\{ \begin{array}{c} \min(\mu_A(x), \mu_B(x)) \\ \frac{\mu_A(x)+\mu_B(y)}{2} \\ \mu_A(x)\mu_B(y) \\ \ldots\ldots\ldots\ldots \end{array} \right\}$$

Union: The union $\cup$ of two fuzzy sets A and B is also defined in several ways:

$$\forall x \in X : \mu_{A\cup B} = \left\{ \begin{array}{c} \max(\mu_A(x), \mu_B(y)) \\ \frac{2\min(\mu_A(x),\mu_B(y))+4\max(\mu_A(x),\mu_B(y))}{6} \\ \mu_A(x) + \mu_B(y) - \mu_A(x)\mu_B(y) \\ \ldots\ldots\ldots\ldots\ldots\ldots \end{array} \right\}$$

In the third definition, the union is put equal to 1 if the sum is greater than 1.

Complement: The complement A′ of a fuzzy set A is defined as:

$$\forall x \in X : \mu_{\mathrm{A}}(x) = 1 - \mu_A(x)$$

Transformation operators. Another important group of operators that characterize fuzzy set theory are the transformation operators. These act on the membership functions in order to modify the linguistic value of the respective fuzzy set. For example, in the clause "number very close to 10," the transformation operator "very" acts on the linguistic term "close to 10" which corresponds to a fuzzy set. Examples of such operators are given in Table 2 (Ross 1995; Zimmermann 1996).

Fuzzy Sets, Systems, and Applications, Table 2 Examples of transformation operators

Very	$\mu_{\hat{A}}(x) = (\mu_A(x))^n, n > 1$
More/less	$\mu_{\hat{A}}(x) = (\mu_A(x))^n, 0 < n < 1$

Cartesian inner product of fuzzy sets. If $A_1, A_2, \ldots, A_v$ are fuzzy sets defined in $U_1, U_2, \ldots, U_v$, the Cartesian inner product of $A_1, A_2, \ldots, A_v$ is a fuzzy set F in $U_1 x\ U_2 x, \ldots x\ U_v$ with membership function (Yan et al. 1994):

$$\mu_F(u_1, u_2, \ldots u_v) = \cap_i = 1.v\mu_{A_i}(u_i)$$

$$\text{e.g. } \mu_F(u_1,u_2,\ldots,u_v) = \min\{\mu_{A1}(u_1),\mu_{A2}(u_2),\ldots,\mu_{AV}(u_v)\}$$

$$\text{or } \mu_F(u_1,u_2,\ldots,u_v) = \mu_{A1}(u_1)\mu_{A2}(u_2)\ldots\mu_{AV}(u_v)$$

Fuzzy relations. Let U_1 and U_2 be two universes of discourse and the membership function μ_R: $U_1 \times U_2 \rightarrow$ [0,1]. Then a fuzzy relation on $U_1 \times U_2$ is defined as (Zimmerman 1996):

$$R_c = \int_{U\times B} \mu_R(u_1,u_2)/(u_1,u_2) \text{ if } U_1, U_2 \text{ are continuous in space}$$

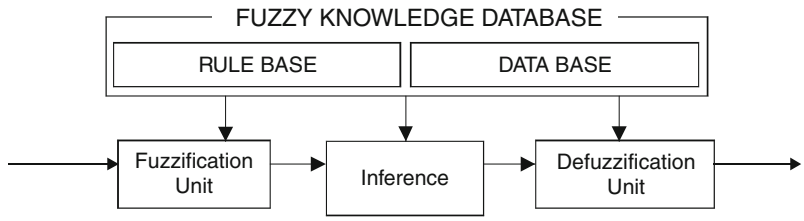
$$\text{or } R_d = \sum_{U\times V} \mu_R(u_1,u_2)/(u_1,u_2) \text{ if } U_1,\ U_2 \text{ are discrete.}$$

Fuzzy Set Composition. Let R_1 and R_2 be two fuzzy relations on $U_1 \times U_2$ and $U_2 \times U_3$ respectively, then the composition of R_1 and R_2 is defined as follows:

$$C = R_1 \circ R_2 = \{(u_1, u_3), \cup(\mu_{R_1}(u_1, u_2) \cap \mu_{R_2}(u_1, u_2))\},$$
$$u_1 \in U_1, u_2 \in U_2, u_3 \in U_3.$$

Implication Rules. Let A and B be two fuzzy sets in U_1,U_2 respectively. Then the implication I:$A => B$ - > $U_1 \times U_2$ is defined as (Zimmerman 1996):

$$I = A \times B = \int_{U_1\times U_2} \mu_A(u_1) \cap \mu_B(u_2)/(u_1, u_2)$$



Fuzzy Sets, Systems, and Applications, Fig. 6 Basic structure of a fuzzy inference system

For example, the rule: "If the error is negative big, then control output is positive big"is an implication error $=>$ control action.

Let the two discrete fuzzy sets $A = \{\Sigma\ \mu_A(u_i)/u_i, i = 1,\ldots,n\}$, $B = \{\Sigma\ \mu_B(v_i)/v_i, i = 1,\ldots,m\}$

Then the implication $A => B$ takes the form:

$$\begin{bmatrix} \mu_A(u_1) \\ \mu_A(u_2) \\ \cdots\cdots \\ \mu_A(u_n) \end{bmatrix} \times [\mu_B(v_1) \quad \mu_B(v_2) \quad \ldots \quad \mu_B(v_m)]$$

$$= \begin{bmatrix} \mu_A(u_1) \wedge \mu_B(v_1) & \mu_A(u_1) \wedge \mu_B(v_2) & \ldots & \mu_A(u_1) \wedge \mu_B(v_m) \\ \mu_A(u_2) \wedge \mu_B(v_1) & \mu_A(u_2) \wedge \mu_B(v_2) & \ldots & \mu_A(u_2) \wedge \mu_B(v_m) \\ \cdots\cdots & \cdots\cdots & \cdots & \cdots \\ \mu_A(u_n) \wedge \mu_B(v_1) & \mu_A(u_n) \wedge \mu_B(v_2) & \ldots & \mu_A(u_n) \wedge \mu_B(v_m) \end{bmatrix}$$

Inference rules. Let R be a fuzzy relation on $U_1 \times U_2$, and let A be a fuzzy set in U_1. The composition

$A^oR = B$,

is a fuzzy set in U_2 representing the implication (conclusion) from the fuzzy set A (fact) based on the implication R (rules). For a multiple input, single output rule base with N rules, the i-th rule is given by

If A_{i1} and ... and A_{ij} and ... and A_{in} then B_i, where

$n =$ the number of input variables x_i

$A_{ij} =$ fuzzy set of input variable x_j in i-th rule

$B_i =$ fuzzy set of output variable y_j in i-th rule

The i-th rule is the implication $I_i = A_i => B_i$, $A_i = \cap_{i=1}^{n} A_{ij}$. Then the implication I_{tot} of N rules is given by $I_{tot} = \cup_{i=1}^{N} R_i = \cup_{i=1}^{N} A_i \rightarrow B_i$.

Fuzzy similarity measures. Fuzzy similarity measures introduce the notion of approximate equality between fuzzy sets. Several fuzzy similarity measures have been proposed, each one with different attributes, see (Pappis and Karacapilidis 1993 and 1995; Wang et al. 1995).

Basic structure of a fuzzy system. The fuzzy knowledge base contains four main types of information: (a) a fuzzification unit (fuzzifier), (b) a fuzzy inference unit, (c) a rule base which essentially maps fuzzy values of the inputs to fuzzy values of the outputs and (d) a defuzzification unit (Fig. 6).

(a) The fuzzifier maps the measured inputs, which usually are crisp values, into the fuzzy linguistic values (fuzzy sets) used by the fuzzy reasoning mechanism.

(b) The fuzzy rules incorporated in the rule base express the input–output relationships usually in an IF-THEN format. For instance, for a two-input, one-output fuzzy system, a fuzzy rule has the general form:

Rule i : IF x is A_i and y is B_i THEN z is C_i

where x and y are input measured variables, z is the controller output variable; A_i, B_i and C_i are linguistic terms (fuzzy sets) such as "negative big", "positive small" or "zero". The if-part of the rule is called condition or premise or antecedent, and the *then-part* is called the consequence or action.

Two are the main approaches in the design of rule bases (Yan et al. 1994):

(i) Heuristic-Mamdani's type approaches (Mamdani 1977; King and Mamdani 1977; Pappis and Mamdani 1977) which provide a convenient way to build fuzzy rules in order to achieve the desired output response, requiring only qualitative knowledge for the behaviour of the system under study.

(ii) Systematic approaches based on Sugeno-type inference systems (Takagi and Sugeno 1985; Pappis and Sugeno 1985; Sugeno and Kang 1988; Sugeno and Yasukawa 1993; Laukoven and Pasino 1995) including hybrid neural-fuzzy frameworks (Kosko 1992), Chebyshev series and Kohonen's networks (Siettos et al. 2002; Alexandridis et al. 2002).

(c) The fuzzy reasoning unit performs various fuzzy logic operations to infer the action for the given fuzzy inputs. During fuzzy inference, the following operations are involved for each fuzzy rule:
   1. Determination of the degree of match between the fuzzy input data and the defined fuzzy sets for each system input variable.
   2. Calculation of the fire strength for each rule based on the degree of match and the connectives (e.g. AND, OR) used with input variables in the antecedent part of the rule.
   3. Derivation of the fuzzy outputs based on the calculated fire strength and the defined fuzzy sets for each output variable in the consequent part of each rule.

(d) The defuzzification unit performs the inverse operation of fuzzification, i.e. extracts the crisp output value from the fuzzy outputs. Several techniques have been proposed for the inference of the fuzzy output based on the rule base. The most commonly used are those of mean of maximum, centroid, and center of sum of areas (Driankov et al. 1993; Ross 1995).

Applications. In developing fuzzy logic and the theory of fuzzy sets, extensive effort has been undertaken by scientists and engineers to deal with a variety of important research topics: inference systems (del Amo et al. 2001); computational linguistics and knowledge representation (Intan and Mukaidono 2002); neural networks (Alpaydin et al. 2002; Oh et al. 2002); genetic algorithms (Spiegel and Sudkamp 2002); information processing (Liu et al. 2002; Hong et al. 2002; Nikravesh et al. 2002); pattern analysis and classification (Gabrys and Bargiela 2002; de Moraes et al. 2002; Pedrycz and Gacek 2002); decision making (Yager 2002b; Wang 2000; Zimmermann et al. 2000; Wang and Lin 2003).

Apart from mathematics and algorithms, fuzzy sets theory has been widely utilized in the context of real life problems (Zimmermann 2001). Such problems are commonly full of uncertainty or approximate reasoning and it is often very difficult to develop mathematical models that can fully describe and incorporate their complexities. In these cases, fuzzy sets theory is essential as it assists on the development of mathematical models that embed imprecision. Fuzzy sets theory has been extensively utilized in the context of operations management, medicine, finance, risk analysis and assessment, water resources management, environmental management, and social sciences. The practical applications of fuzzy logic are wide, indeed, including, among others, process control (Tong et al. 2002); robotics (Ruan et al. 2003); scheduling (Adamopoulos et al. 2000; Karacapilidis et al. 2000; Muthusamy et al. 2003); transportation (Chen et al. 2008); nuclear engineering (Kunsch and Fortemps 2002); medicine (Barro and Marin 2002; Blanco et al. 2002; Kilic et al. 2002); and economics (Kahraman et al. 2002; Gil-Lafuente 2005). A review of specific areas of application of fuzzy sets theory follows.

Management Science: Fuzzy sets theory has been widely utilized in the case of management science. Facets of the theory of fuzzy sets have been incorporated into supply chain management and even strategic management decision-making processes. The potential utilization of fuzzy sets theory has long been noticed in the case of operations management (Zimmerman 1983). More specifically, fuzzy logic has been incorporated into models addressing issues in the context of outsourcing logistics activities (Bottani and Rizzi 2006; Cheng et al. 2008; Liu and Wang 2009); manufacturing and production problems (Lee and Yao 1998; Majozi and Zhu 2005; Liang and Cheng 2009), supply chain modeling (Petrovic et al. 1999; Sevastjanov and Róg 2003; Sheu 2004; Zhang and Lu 2007); optimization of supply chain operations (Silva et al. 2007); and traffic and transportation processes (Teodorović 1994). In addition, fuzzy logic has been used in the context of strategic management (Dutta 1993; Kardaras and Karakostas 1999; Lin and Hsieh 2004; Narukawa and Torra 2007; Xu et al. 2009) as well as marketing (Setnes and Kaymak 2001; Ramkumar et al. 2010). Fuzzy logic has also been applied in industrial applications (Sàrfi et al. 1996; Bansal 2003), including power systems planning (David and Zhao 1991; Ong and Nee 1994; Guan et al. 1995; Ramírez-Rosado and Domínguez-Navarro 2004) and product conformance specification procedures (Bradshaw 1983).

Medical Science: Decision-making processes in medicine science can be hindered due to the complexity of biological systems and high data uncertainty. Fuzzy logic theory provides a means of better modeling related problems. Applications of fuzzy sets theory can be found in the context

of medical diagnosis (Belacel and Boulassel 2001; De et al. 2001; Szmidt and Kacprzyk 2004; Polat et al. 2006); effective management of medical diagnostic problems (Chen 1994) and multiple alternative decision-making problems during medical diagnosis (Cheng and McInnis 1980). Additionally, fuzzy logic has been applied in medical expert systems (Hudson and Cohen 1994) and used for the development of measures for the anxiety induced by a given decision-making process by an individual (Yager 1982). A survey with respect to the application of fuzzy sets theory in medical sciences is given in Abbod et al. (2001).

Social Sciences: Biswas (1995) used fuzzy logic for the evaluation of students' answer scripts. Lalla et al. (2005) utilized fuzzy sets theory for the evaluation of teaching activities, and Smithson (1982) presented tools for applying fuzzy sets concepts to the social and behavioral sciences and examples of their uses.

Financial Science: Here fuzzy sets theory has been utilized in the context of ratio analysis (Gutierrez and Carmona 1988), enterprise financial status synthetic evaluation (Lee and Chang 2009), and in ranking vague economic investment information when a present worth criterion is used (Sorenson and Lavelle 2008). In addition, fuzzy logic has been utilized in the context of multi-objective finance-based scheduling for construction projects under uncertainty (Afshar and Fathi 2009); financial evaluation in the public sector (Ammar et al. 2004); engineering economic decision-making of firms (Kahraman 2008); financial analysis during corporate acquisition processes (McIvor et al. 2004); evaluation of fuzzy financial profitability of load management alternatives (Sheen 2005), development of financial models that assist the identification of different states of one market so that a firm could modify its actions and make successful trades (Van den Berg et al. 2004); financial risk management and credit scoring (Yu et al. 2009) and in other problems in economics and finance (Buckley 1987; Buckley 1992).

Environmental Science: Applications of fuzzy sets theory in environmental science include (Esogbue et al. 1992), where fuzzy sets methodologies were used to solve an optimal flood control planning problem by an integration of structural and non-structural measures with the objective of optimizing the flood damage reduction due to recurrent floods. Koo and Shin (1985) utilized fuzzy sets for multi-objective river quality management under a direct regulation scheme, and Liou et al. (2003) proposed an indicator model for evaluating trends in river quality using two-stage fuzzy set theory to condense efficiently monitoring data. Hanesch et al. (2001) utilized fuzzy c-means cluster analysis and non-linear mapping for tracing the distribution and source of pollutants to assess potential environmental hazards, and McBratney and Odeh (1997) studied the potential applications of fuzzy set theory and fuzzy logic in soil science. Fuzzy logic was applied to timber harvest planning (Boyland et al. 2006) and to environmental risk assessment by Ghomshei and Meech (2000).

Other applications: A sample range of applications of fuzzy sets theory to other real-world problems include: Nguyen (1985) in the context of mining geomechanics decision-processes; Cayrac et al. (1996) for satellite fault diagnosis procedures; McBratney and Moore (1985) for dealing with the continuity of climatic data; Cao and Chen (1983) for meteorological forecasting; developing of electronic video camera image stabilizers (Egusa et al. 1995); handling uncertain image information (Laplante and Sinha 1996); classification of geometric figures and chromosome images through the use of shape-oriented angular and dimensional proximity measures (Lee 1976); the problem of image reconstruction (Nobuhara et al. 2006); the fatigue problem of reinforced concrete decks of bridge structures (Shiraishi et al. 1988); data mining processes (Li and Deogun 2009); disaster control systems planning (Esogbue 1996); etc.

See

- ▶ Artificial Intelligence
- ▶ Control Theory

References

Abbod, M. F., Von Keyserlingk, D. G., Linkens, D. A., & Mahfouf, M. (2001). Survey of utilisation of fuzzy technology in medicine and health care. *Fuzzy Sets and Systems, 120*(3), 331–349.

Adamopoulos, G. I., Pappis, C. P., & Karacapilidis, N. I. (2000). A methodology for solving a range of sequencing problems with uncertain data. In R. Slowinski & M. Hapke (Eds.), *Advances in scheduling and sequencing under fuzziness* (pp. 147–164). Heidelberg: Physica-Verlag.

Afshar, A., & Fathi, H. (2009). Fuzzy multi-objective optimization of finance-based scheduling for construction projects with uncertainties in cost. *Engineering Optimization, 41*(11), 1063–1080.

Albrecht, R. F. (2003). Interfaces between fuzzy topological interpretation of fuzzy sets and intervals. *Fuzzy Sets and Systems, 135*(1), 11–20.

Alexandridis, A., Siettos, C. I., Sarimveis, H., Boudouvis, A. G., & Bafas, G. V. (2002). Modeling of nonlinear process dynamics using Kohonen's neural networks. *Computers and Chemical Engineering, 26*, 479–486.

Alpaydin, G., Dündar, G., & Balkir, S. (2002). Evolution-based design of neural fuzzy networks using self-adapting genetic parameters. *IEEE Transactions on Fuzzy Systems, 10*(2), 211–221.

Ammar, S., Duncombe, W., Jump, B., & Wright, R. (2004). Constructing a fuzzy-knowledge-based-system: An application for assessing the financial condition of public schools. *Expert Systems with Applications, 27*(3), 349–364.

Assilian, S., & Mamdani, E. H. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man–Machine Studies, 7*(1), 1–13.

Bansal, R. C. (2003). Bibliography on the fuzzy set theory applications in power systems (1994–2001). *IEEE Transactions on Power Systems, 18*(4), 1291–1299.

Barro, S., & Marin, R. (2002). *Fuzzy logic in medicine*. Heidelberg: Physica-Verlag.

Belacel, N., & Boulassel, M. R. (2001). Multicriteria fuzzy assignment method: A useful tool to assist medical diagnosis. *Artificial Intelligence in Medicine, 21*(1–3), 201–207.

Biacino, L., & Gerla, G. (2002). Fuzzy logic, continuity and effectiveness. *Archive for Mathematical Logic, 41*, 643–667.

Biswas, R. (1995). An application of fuzzy sets in students' evaluation. *Fuzzy Sets and Systems, 74*(2), 187–194.

Blanco, A., Pelta, D. A., & Verdegay, J. L. (2002). Applying a fuzzy sets-based heuristic to the protein structure prediction problem. *International Journal of Intelligent. Systems, 17*(7), 629–643.

Bottani, E., & Rizzi, A. (2006). A fuzzy TOPSIS methodology to support outsourcing of logistics services. *Supply Chain Management, 11*(4), 294–308.

Boyland, M., Nelson, J., Bunnell, F., & D' Eon, R. G. (2006). An application of fuzzy set theory for seral-class constraints in forest planning models. *Forest Ecology and Management, 223*(1–3), 395–402.

Bradshaw, C. W., Jr. (1983). A fuzzy set theoretic interpretation of economic control limits. *European Journal of Operational Research, 13*(4), 403–408.

Buckley, J. J. (1987). The fuzzy mathematics of finance. *Fuzzy Sets and Systems, 21*(3), 257–273.

Buckley, J. J. (1992). Solving fuzzy equations in economics and finance. *Fuzzy Sets and Systems, 48*(3), 289–296.

Cao, H., & Chen, G. (1983). Some applications of fuzzy sets to meteorological forecasting. *Fuzzy Sets and Systems, 9*(1–3), 1–12.

Cayrac, D., Dubois, D., & Prade, H. (1996). Handling uncertainty with possibility theory and fuzzy sets in a satellite fault diagnosis application. *IEEE Transactions on Fuzzy Systems, 4*(3), 251–269.

Chen, S. M. (1994). A weighted fuzzy reasoning algorithm for medical diagnosis. *Decision Support Systems, 11*(1), 37–43.

Chen, M., Ishii, H., & Wu, C. (2008). Transportation problems on a fuzzy network. *International Journal of Innovative Computing Information and Control, 4*, 1105–1109.

Cheng, J. H., Chen, S. S., & Chuang, Y. W. (2008). An application of fuzzy delphi and fuzzy AHP for multi-criteria evaluation model of fourth party logistics. *WSEAS Transactions on Systems, 7*(5), 466–478.

Cheng, Y. Y. M., & McInnis, B. (1980). Algorithm for multiple attribute, multiple alternative decision problems based on fuzzy sets with application to medical diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-10*(10), 645–650.

David, A. K., & Zhao, R. (1991). An expert system with fuzzy sets for optimal planning. *IEEE Transactions on Power Systems, 6*(2), 59–65.

De Moraes, R. M., Banon, G. J. F., & Sandri, S. A. (2002). Fuzzy expert systems architecture for image classification using mathematical morphology operators. *The Information of the Science, 142*(1/4), 7–21.

De, S. K., Biswas, R., & Roy, A. R. (2001). An application of intuitionistic fuzzy sets in medical diagnosis. *Fuzzy Sets and Systems, 117*(2), 209–213.

Del Amo, A., Comez, D., Montero, J., & Biging, G. (2001). Relevance and redundancy in fuzzy classification systems. *Mathware and Soft Computing, VIII, 3*, 203–216.

Di Nola, A., Esteva, F., Garcia, P., Godo, L., & Sessa, S. (2002). Subvarieties of BL-algebras generated by single component chains. *Archives for Mathematical Logic, 41*, 673–685.

Driankov, D., Hellendoorn, H., & Reinfrank, M. (1993). *An introduction to fuzzy control*. Berlin: Springer.

Dubois, D., & Prade, H. (1980). *Fuzzy sets and systems: Theory and applications*. New York: Academic.

Dutta, S. (1993). Fuzzy logic applications: Technological and strategic issues. *IEEE Transactions on Engineering Management, 40*(3), 237–254.

Egusa, Y., Akahori, H., Morimura, A., & Wakami, N. (1995). Application of fuzzy set theory for an electronic video camera image stabilizer. *IEEE Transactions on Fuzzy Systems, 3*(3), 351–356.

Esogbue, A. O. (1996). Fuzzy sets modeling and optimization for disaster control systems planning. *Fuzzy Sets and Systems, 81*(1), 169–183.

Esogbue, A. O., Theologidu, M., & Guo, K. (1992). On the application of fuzzy sets theory to the optimal flood control problem arising in water resources systems. *Fuzzy Sets and Systems, 48*(2), 155–172.

Gabrys, B., & Bargiela, A. (2002). General fuzzy min-max neural network for clustering and classification. *IEEE Transactions on Neural Networks, 11*(3), 769–783.

Ghomshei, M. M., & Meech, J. A. (2000). Application of fuzzy logic in environmental risk assessment: Some thoughts on fuzzy sets. *Cybernetics and Systems, 31*(3), 317–332.

Gil-Lafuente, A. M. (2005). *Fuzzy logic in financial analysis*. New York: Springer.

Guan, X., Liu, W. H. E., & Papalexopoulos, A. D. (1995). Application of a fuzzy set method in an optimal power flow. *Electric Power Systems Research, 34*(1), 11–18.

Gutierrez, I., & Carmona, S. (1988). A fuzzy set approach to financial ratio analysis. *European Journal of Operational Research, 36*(1), 78–84.

Hanesch, M., Scholger, R., & Dekkers, M. J. (2001). The application of fuzzy C-means cluster analysis and non-linear mapping to a soil data set for the detection of polluted sites. *Physics and Chemistry of the Earth, Part A: Solid Earth and Geodesy, 26*(11–12), 885–891.

Hitachi (1984) http://www.hitachi.com/rev/1999/revjun99/r3_109.pdf

Holmblad, L. P., & Østergaard, J.-J. (1995). The FLS application of fuzzy logic. *Fuzzy Sets and Systems, 70*(2–3), 135–146.

Hong, T. P., Lin, K. Y., & Wang, S. L. (2002). Mining linguistic browsing patterns in the world wide web. *Soft Computing, 6*(5), 329–336.

Hudson, D. L., & Cohen, M. E. (1994). Fuzzy logic in medical expert systems. *IEEE Engineering in Medicine and Biology Magazine, 13*(5), 693–698.

Hung, W. L. (2002). Partial correlation coefficients of intuitionist fuzzy sets. *International Journal of Uncertainty Fuzziness Knowledge-Based Systems, 10*(1), 105–112.

Intan, R., & Mukaidono, M. (2002). On knowledge-based fuzzy sets. *International Journal of Fuzzy Systems, 4*(2), 655–664.

Kahraman, C. (2008). Fuzzy sets in engineering economic decision-making. *Studies in Fuzziness and Soft Computing, 233*, 1–9.

Kahraman, C., Ruan, D., & Tolga, E. (2002). Capital budgeting techniques using discounted fuzzy versus probabilistic cash flows. *The Information of the Science, 142*(1/4), 57–56.

Karacapilidis, N. I., Pappis, C. P., & Adamopoulos, G. I. (2000). Fuzzy set approaches to lot sizing. In R. Slowinski & M. Hapke (Eds.), *Advances in scheduling and sequencing under fuzziness* (pp. 291–304). Heidelberg: Physica-Verlag.

Kardaras, D., & Karakostas, B. (1999). Use of fuzzy cognitive maps to simulate the information systems strategic planning process. *Information and Software Technology, 41*(4), 197–210.

Karr, C. L., & Gentry, E. J. (1993). Fuzzy control of pH using genetic algorithms. *IEEE Transactions on Fuzzy Systems, 1*, 46–53.

Kilic, K., Sproule, B. A., Türksen, I. B., & Naranjo, C. A. (2002). Fuzzy system modeling in pharmacology: An improved algorithm. *Fuzzy Sets and Systems, 130*(2), 253–264.

King, P. J., & Mamdani, E. H. (1977). The application of fuzzy control systems to industrial processes. *Automatica, 13*, 235–242.

Koo, J.-K., & Shin, H.-S. (1985). Application of fuzzy sets to water quality management. *Water Supply, 4*(1), 293–305.

Kosko, B. (1992). *Neural networks and fuzzy systems: A dynamical system approach*. Englewood Cliffs: Prentice-Hall.

Kunsch, P. L., & Fortemps, P. (2002). A Fuzzy decision support system for the economic calculus in radioactive waste management. *The Information of the Science, 142*, 103–116.

Lalla, M., Facchinetti, G., Mastroleo, G., et al. (2005). Ordinal scales and fuzzy set systems to measure agreement: An application to the evaluation of teaching activity. *Quality and Quantity, 38*(5), 577–601.

Laplante, P. A., & Sinha, D. (1996). Extensions to the fuzzy pointed set with applications to image processing. *IEEE Transactions on Systems, Man, and Cybernetics. Part B, Cybernetics, 26*(1), 21–28.

Laukoven, E. G., & Pasino, K. M. (1995). Training fuzzy systems to perform estimation and identification. *Engineering Applications on Artificial Intelligence, 8*(5), 499–514.

Lee, E. T. (1976). An application of fuzzy sets to the classification of geometric figures and chromosome images. *Information Sciences, 10*(2), 95–114.

Lee, M. C., & Chang, J. F. (2009). Agent and multi-agent systems: technologies and applications. *Lecture Notes in Computer Science, 5559*, 542–549.

Lee, H. M., & Yao, J. S. (1998). Economic production quantity for fuzzy demand quantity and fuzzy production quantity. *European Journal of Operational Research, 109*(1), 203–211.

Li, D., & Deogun, J. S. (2009). Applications of fuzzy and rough set theory in data mining. *Studies in Computational Intelligence, 225*, 71–113.

Liang, T. F., & Cheng, H. W. (2009). Application of fuzzy sets to manufacturing/distribution planning decisions with multi-product and multi-time period in supply chains. *Expert Systems with Applications, 36*, 3367–3377.

Lin, C., & Hsieh, P. J. (2004). A fuzzy decision support system for strategic portfolio management. *Decision Support Systems, 38*(3), 383–398.

Liou, S. M., Lo, S. L., & Hu, C. Y. (2003). Application of two-stage fuzzy set theory to river quality evaluation in Taiwan. *Water Research, 37*(6), 1406–1416.

Liu, M., Wan, C., & Wang, L. (2002). Content-based audio classification and retrieval using a fuzzy logic system: Towards multimedia search engines. *Soft Computing, 6*(5), 357–364.

Liu, H. T., & Wang, W. K. (2009). An integrated fuzzy approach for provider evaluation and selection in third-party logistics. *Expert Systems with Applications, 36*(3 PART 1), 4387–4398.

Majozi, T., & Zhu, X. X. (2005). A combined fuzzy set theory and MILP approach in integration of planning and scheduling of batch plants - personnel evaluation and allocation. *Computers and Chemical Engineering, 29*(9), 2029–2047.

Mamdani, E. H. (1977). Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Transactions on Computers, C-26*(12), 1182–1191.

McBratney, A. B., & Moore, A. W. (1985). Application of fuzzy sets to climatic classification. *Agricultural and Forest Meteorology, 35*(1–4), 165–185.

McBratney, A. B., & Odeh, I. O. A. (1997). Application of fuzzy sets in soil science: Fuzzy logic, fuzzy measurements and fuzzy decisions. *Geoderma, 77*(2–4), 85–113.

McIvor, R. T., McCloskey, A. G., Humphreys, P. K., & Maguire, L. P. (2004). Using a fuzzy approach to support financial analysis in the corporate acquisition process. *Expert Systems with Applications, 27*(4), 533–547.

Muthusamy, K., Sung, S. C., Vlach, M., & Ishii, H. (2003). Scheduling with fuzzy delays and fuzzy precedences. *Fuzzy Sets and Systems, 134*(3), 387–395.

Naessens, H., De Meyer, H., & De Baets, B. (2002). Algorithms for the computation of T-transitive closures. *IEEE Transactions on Fuzzy Systems, 10*(4), 541–551.

Narukawa, Y., & Torra, V. (2007). Fuzzy measures and integrals in evaluation of strategies. *Information Sciences, 177*(21), 4686–4695.

Nguyen, V. U. (1985). Some fuzzy set applications in mining geomechanics. *International Journal of Rock Mechanics and Mining Sciences, 22*(6), 369–379.

Nikravesh, M., Loia, V., & Azvine, B. (2002). Fuzzy logic and the internet (FLINT): Internet, world wide web and search engines. *Soft Computing, 6*(5), 287–299.

Nobuhara, H., Bede, B., & Hirota, K. (2006). On various eigen fuzzy sets and their application to image reconstruction. *Information Sciences, 176*(20), 2988–3010.

Novak, V. (2002). Joint consistency of fuzzy theories. *Mathematical Logic Quarterly, 48*, 563–573.

Oh, S. K., Kim, D. W., & Pedrycz, W. (2002). Hybrid fuzzy polynomial neural networks. *International Journal of Uncertainty Fuzziness Knowledge-Based Systems, 10*(3), 257–280.

Ong, S. K., & Nee, A. Y. C. (1994). Application of fuzzy set theory to setup planning. *CIRP Annals - Manufacturing Technology, 43*(1), 137–144.

Østergaard, J. J. (1977). Fuzzy logic control of a heat exchanger system. In M. M. Gupta, G. N. Saridis, & B. R. Gaines (Eds.), *Fuzzy automata and decision processes* (pp. 285–320). Amsterdam: North-Holland.

Østergaard, J. J. (1990) Fuzzy II: The new generation of high level kiln control. *Zement Kalk Gips (Cement-Lime-Gypsum), 43*(11), 539–541.

Pappis, C. P., & Karacapilidis, N. I. (1993). A comparative assessment of measures of similarity of fuzzy values. *Fuzzy Sets and Systems, 56*, 171–174.

Pappis, C. P., & Karacapilidis, N. I. (1995). Application of a similarity measure of fuzzy sets to fuzzy relational equations. *Fuzzy Sets and Systems, 75*, 35–142.

Pappis, C. P., & Mamdani, E. H. (1977). A fuzzy logic controller for a traffic junction. *IEEE Systems Man and Cybernetics, SMC-7*(10), 707–717.

Pappis, C. P., & Sugeno, M. (1985). Fuzzy relational equations and the inverse problem. *Fuzzy Sets and Systems, 15*(1), 79–90.

Pedrycz, W., & Gacek, A. (2002). Temporal granulation and its application to signal analysis. *The Information of the Science, 143*(1/4), 47–71.

Pedrycz, W., & Vasilakos, A. V. (2002). Modularization of fuzzy relational equations. *Soft Computing, 6*(1), 33–37.

Petrovic, D., Roy, R., & Petrovic, R. (1999). Supply chain modelling using fuzzy sets. *International Journal of Production Economics, 59*(1), 443–453.

Polat, K., Şahan, S., & Salih, G. (2006). A new method to medical diagnosis: Artificial immune recognition system (AIRS) with fuzzy weighted pre-processing and application to ECG arrhythmia. *Expert Systems with Applications, 31*(2), 264–269.

Pradera, A., Trillas, E., & Calvo, T. (2002). A general class of triangular norm-based aggregation operators: Quasilinear T-S operators. *International Journal of Approximate Reasoning, 30*(1), 57–72.

Procyk, T. J., & Mamdani, E. H. (1979). A linguistic self-organizing process controller. *Automatica, 15*, 15–30.

Ramírez-Rosado, I. J., & Domínguez-Navarro, J. A. (2004). Possibilistic model based on fuzzy sets for the multiobjective optimal planning of electric power distribution networks. *IEEE Transactions on Power Systems, 19*(4), 1801–1810.

Ramkumar, V., Rajasekar, S., & Swamynathan, S. (2010). Scoring products from reviews through application of fuzzy techniques. *Expert Systems with Applications, 37*(10), 6862–6867.

Ross, T. J. (1995). *Fuzzy logic with engineering applications*. New York: McGraw-Hill.

Ruan, D., Zhou, C., & Gupta, M. M. (2003). Fuzzy set techniques for intelligent robotic systems. *Fuzzy Sets and Systems, 134*(1), 1–4.

Sàrfi, R. J., Salama, M. M. A., & Chikhani, A. Y. (1996). Applications of fuzzy sets theory in power systems planning and operation: A critical review to assist in implementation. *Electric Power Systems Research, 39*(2), 89–101.

Setnes, M., & Kaymak, U. (2001). Fuzzy modeling of client preference from large data sets: An application to target selection in direct marketing. *IEEE Transactions on Fuzzy Systems, 9*(1), 153–163.

Sevastjanov, P. V., & Róg, P. (2003). Fuzzy modeling of manufacturing and logistic systems. *Mathematics and Computers in Simulation, 63*(6), 569–585.

Sheen, J. N. (2005). Fuzzy-financial decision-making: Load management programs case study. *IEEE Transactions on Power Systems, 20*(4), 1808–1817.

Sheu, J. B. (2004). A hybrid fuzzy-based approach for identifying global logistics strategies. *Transportation Research Part E: Logistics and Transportation Review, 40*(1), 39–61.

Shiraishi, N., Furuta, H., & Ozaki, Y. (1988). Application of fuzzy set theory to fatigue analysis of bridge structures. *Information Sciences, 45*(2), 175–184.

Siettos, C. I., Boudouvis, A. G., & Bafas, G. V. (2002). Approximation of fuzzy control systems using truncated Chebyshev series. *Fuzzy Sets and Systems, Fuzzy Sets and Systems, 126*, 89–104.

Silva, C. A., Sousa, J. M. C., & Runkler, T. A. (2007). Optimization of logistic systems using fuzzy weighted aggregation. *Fuzzy Sets and Systems, 158*(17), 1947–1960.

Smithson, M. (1982). Applications of fuzzy set concepts to behavioral sciences. *Mathematical Social Sciences, 2*(3), 257–274.

Sorenson, G. E., & Lavelle, J. P. (2008). A comparison of fuzzy set and probabilistic paradigms for ranking vague economic investment information using a present worth criterion. *The Engineering Economist, 53*(1), 42–67.

Spiegel, D., & Sudkamp, T. (2002). Employing locality in the evolutionary generation of fuzzy rule bases. *IEEE Transactions on Systems Man Cybernet – Part B: Cybernetics, 32*(3), 296–305.

Sugeno, M., & Kang, G. T. (1988). Structure identification of fuzzy model. *Fuzzy Sets and Systems, 28*, 15–23.

Sugeno, M., & Yasukawa, T. (1993). A fuzzy-logic-based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems, 1*(1), 7–31.

Szmidt, E., & Kacprzyk, J. (2004). A similarity measure for intuitionistic fuzzy sets and its application in supporting medical diagnostic reasoning. *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), 3070*, 388–393.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its application to modelling and control. *IEEE Transactions on Systems Man Cybernetics, 15*, 116–132.

Teodorović, D. (1994). Fuzzy sets theory applications in traffic and transportation. *European Journal of Operational Research, 74*(3), 379–390.

Togai, M., & Watanabe, H. (1986). Expert systems on a chip: An engine for real-time approximate reasoning. *IEEE Expert Magazine, 1*, 55–62.

Tong, S., Wang, T., & Li, H. X. (2002). Fuzzy robust tracking control for uncertain nonlinear systems. *International Journal of Approximate Reasoning, 30*, 73–90.

Van den Berg, J., Kaymak, U., & Van Den Bergh, W. M. (2004). Financial markets analysis by using a probabilistic fuzzy modelling approach. *International Journal of Approximate Reasoning, 35*(3), 291–305.

Wang, L. X. (1992) Fuzzy systems are universal approximators. *Proceedings of IEEE International Conference on Fuzzy Systems, San Diego*, 1163–1170.

Wang, H. F. (2000). Fuzzy multicriteria decision making – an overview. *Journal of Intelligent and Fuzzy Systems, 9*(1/2), 61–84.

Wang, W., De Baets, B., & Kerre, E. (1995). A comparative study of similarity measures. *Fuzzy Sets and Systems, 73*, 259–268.

Wang, J., & Lin, Y. I. (2003). A fuzzy multicriteria group decision making approach to select configuration items for software development. *Fuzzy Sets and Systems, 134*(3), 343–363.

Xu, X., Liu, X., & Yan, C. (2009). Applications of axiomatic fuzzy set clustering method on management strategic analysis. *European Journal of Operational Research, 198*(1), 297–304.

Yager, R. R. (1982). Measuring tranquility and anxiety in decision making: An application of fuzzy sets. *International Journal of General Systems, 8*(3), 139–146.

Yager, R. R. (2002a). The power average operator. *IEEE Transactions on Systems Man Cybernetics-Part A: Systems Humans, 31*(6), 724–730.

Yager, R. R. (2002b). On the valuation of alternatives for decision-making under uncertainty. *International Journal of Intelligent Systems, 17*(7), 687–707.

Yamakawa, T., & Miki, T. (1986). The current mode fuzzy logic integrated circuits fabricated by the standard CMOS process. *IEEE Transactions on Computers, C-35*(2), 161–167.

Yan, J., Ryan, M., & Power, J. (1994). *Using fuzzy logic*. Upper Saddle River: Prentice Hall.

Yu, L., Wang, S., & Lai, K. K. (2009). An intelligent-agent-based fuzzy group decision making model for financial multicriteria decision support: The case of credit scoring. *European Journal of Operational Research, 195*(3), 942–959.

Zadeh, L. A. (1965). Fuzzy sets. *Infection Control, 8*, 338–353.

Zadeh, L. A. (1973). Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man, and Cybernetics, 3*, 28–44.

Zhang, G., & Lu, J. (2007). Model and approach of fuzzy bilevel decision making for logistics planning problem. *Journal of Enterprise Information Management, 20*(2), 178–197.

Zimmerman, H. J. (1983). Using fuzzy sets in operational research. *European Journal of Operational Research, 13*(3), 201–216.

Zimmermann, H. J. (1996). *Fuzzy set theory and its applications* (3rd ed.). Norwell, MA: Kluwer.

Zimmermann, H. J. (2001). *Fuzzy set theory–and its applications*. Netherlands: Springer.

Zimmermann, H. J., Ruan, D., & Huang, C. (Eds.). (2000). *Fuzzy sets and operations research for decision support: Key selected papers*. Beijing: Beijing Normal University Press.

G

GA

► Genetic Algorithms

Game Theory

William F. Lucas[1] and Sauleh A. Siddiqui[2]
[1]Claremont Graduate University, Claremont, CA, USA
[2]University of Maryland, College Park, MD, USA

Introduction

Game theory studies situations involving conflict and cooperation. The three main elements of a game are players, strategies, and payoffs. Games arise when two or more decision makers (players) select from various courses of action (called strategies) which in turn result in likely outcomes (expressed as payoffs). There must be at least two interacting participants with different goals in order to have a game. Game theory makes use of the vocabulary from common parlor games and sports. It is, nevertheless, a serious mathematical subject with a broad spectrum of applications in the social, behavioral, managerial, financial, system, and military sciences.

Game theory differs from classical optimization subjects in that it involves two or more players with different objectives. It also extends the traditional uses of probability and statistics beyond the study of one-person decisions in the realm of statistical uncertainty. This latter case is often referred to as games of chance or games against nature in contrast to the games of skill studied in game theory. Many aspects of social and physical science can be viewed as zero-person games since actions are frequently specified by various laws that are not under human control.

Game theory presumes that conflict is not an evil in itself and as such unworthy of study. Rather, that this topic arises naturally when individuals have free will, different desires, and the freedom of choice. Furthermore, this subject often provides guidelines to aid in the resolution of conflict. Game theory also assumes that the players can quantify potential outcomes (as in measurement theory or utility theory), that they are rational in the sense that they seek to maximize their payoffs, and skillful enough to undertake the necessary calculations. The theory of games attempts to describe what is optimal strategic behavior, the nature of equilibrium outcomes, the formation and stability of coalitions, as well as fairness.

There are many different ways to classify games. A significant difference exists between the two-person games and the multiperson ones (also called the n-person games when $n \geq 3$). There is a major distinction depending upon whether games are played in a cooperative or noncooperative manner. The nature of the types or amount of information available to the players is very fundamental in the analysis of games, and this relates to whether the best way to play involves pure or randomized strategies.

A further way to classify games is if they are played over one period or repeatedly over time. Static games are single period games in which all players move simultaneously without observing any other moves.

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Dynamic games are multiperiod games where players interact by playing simultaneous moves numerous times. Unlike static games, players have some information about past moves and payoffs so they may change strategies as the game proceeds. Evolutionary Game Theory (discussed below) provides a specialized framework for studying dynamic games.

Information and Strategies

Any possible way a player can play completely through a game is called a pure strategy for this player. It is an overall plan specifying the actions (moves) to be taken in all eventualities which can conceivably arise. In theory such pure strategies suffice to solve many popular recreational games such as checkers which have perfect information. A game has perfect information if throughout its play all the rules, possible choices, and past history of play by any player are known to all of the participants. In this case there are no unknown positions or hidden moves, and thus no need for secrecy, deception, or bluffing.

The first general theorem in game theory was published by the logician Ernst Zermelo in 1913. It states that there is an optimal pure strategy for playing any finite game with perfect information. An elementary game with perfect information such as tic-tac-toe soon becomes no real challenge. Each player soon discovers a strategy that prevents the other from winning. From then on this game always results in a draw. On the other hand, Zermelo's theorem is an example of an existence theorem. It does not provide a practical way to determine an optimal pure strategy for many interesting but complex games with perfect information such as chess. Furthermore, one cannot even spell out one pure strategy for chess — one that lists a possible response to all legal moves by an opponent. The challenge of such games comes from the bewildering complexity and imagination involved.

Many other games like the card games known as poker, however, do not have perfect information. Secrecy, deception, randomness, and bluffing are in order. Another level of interest and a new notion of strategic choice enter. Pure strategies no longer suffice for optimal play. The main fundamental concept for such noncooperative games is that of a mixed strategy. A mixed strategy for a player is a probability distribution over the player's pure strategies. The idea is that a player will pick a particular pure strategy with some given probability. This greatly enlarges the realm of strategies from which each player can choose. The tradeoff, on the other hand, is that the players must now view their potential payoffs as averages. They thus resort to maximizing their gains in terms of expected values in a statistical sense. These ideas are best illustrated by the theory of matrix games.

Matrix Games

The best known games are the two-person, zero-sum games. Any game is called zero-sum when the particular payoffs to the players always sum to zero. In the case of two players this states that one's winnings equal the other's losses. There is clearly no room for cooperation in this case. These games are also referred to as strictly competitive or antagonistic. They arise in many sorts of duels, inspections, searches, business competitions, and voting situations, as well as most parlor games and sports contests.

These games are characterized by an $m \times n$ table of numbers and are accordingly referred to as matrix games. The rows of the table correspond to the pure strategies for the first player, denoted by I. The columns are likewise identified with the pure strategies of the second player, II. The numbers within the table itself are the corresponding payoffs received by player I from player II. A negative number in the matrix means that I makes a (positive) payment to II. Each player seeks to select a strategy so as to maximize the player's payoff.

The theory of matrix games can be illustrated by the following 2×2 zero-sum game of matching coins. Player I has two pure strategies: to show heads H or tails T. Player II can likewise select H or T. If the two players' coins match with either two heads or else two tails, then player I wins \$3 or \$1, respectively, from player II. If the coins do not match (one H and one T), then player II collects \$2 from player I. This game can be represented by Fig. 1.

The worse thing that can happen to player I is the maximin value of -2. (This is the largest of the smallest numbers from each row.) Similarly, the minimax value for player II is 1. This is the smallest

		Player II H	Player II T	Row minima
Player I	H	3	–2	–2
	T	–2	1	–2
Column maxima:		3	1	

Game Theory, Fig. 1 A matrix game

loss player II can guarantee and it occurs when player II plays the second column T (while I plays the second row T). There is a gap of \$3 between this maximin value of \$1 and the minimax value of − \$2. Both players can win some of this gap of 3 units if they resort to mixed strategies and are willing to evaluate their payoffs in terms of expected values.

If either player uses the optimal mixed strategy of playing H with probability 3/8 and T with probability 5/8, the player can ensure an average payoff of

$$3(3/8) - 2(5/8) = -1/8 = -2(3/8) + 1(5/8)$$

against any strategy by the opposing player. Using mixed strategies the players can close the gap between -2 and 1 to the game's (expected) value of $-1/8$. This game favors player II who should average a gain of 12.5¢ per play. This game is not fair in the sense that optimal play does not produce an expected outcome of 0. The optimal mixed strategies (3/8, 5/8) for players I and II along with the value $-1/8$ are called the solution of this matrix game. (In general, the two players will not have the same optimal mixed strategy as is the case for this game with a symmetric payoff matrix.)

The main theoretical result for matrix games is the famous minimax theorem proved by John von Neumann in 1928. It states that any matrix game has a solution in terms of mixed strategies. Each player has an optimal mixed strategy which guarantees that the player will achieve the value of the game (in the statistical sense of expected values).

Von Neumann also observed in 1947 that the duality theorem in linear programming is equivalent to his minimax theorem. Furthermore, it is known that the subjects of matrix games and linear programming are entirely equivalent mathematically. Various algorithms are known for solving $m \times n$ matrix games. However, one typically expresses the solution for a matrix game in terms of a pair of dual linear programs and employs one of the popular algorithms used in the latter subject.

Noncooperative Games

When games are not zero-sum or have more than two players, then it is essential to distinguish between whether they are played in a cooperative or noncooperative manner. To cooperate means the players are able to communicate (negotiate or bargain) and correlate their strategy choices before they play. Also, that any agreements made are binding (enforceable). In contrast, each player in a noncooperative game chooses a strategy unaware of the selection made by the other players.

For noncooperative games the primary ingredient to any notion of solution is that of an equilibrium point. A set of (pure or mixed) strategies, one for each player, is said to be in equilibrium if no one player can change strategy unilaterally to obtain a higher payoff. Unfortunately, equilibrium outcomes do not always possess every property that one would desire for a satisfactory concept of solution. Nevertheless, this idea of equilibrium seems crucial to the very notion of what can be called a solution to a noncooperative game. It is the social science analogy to the idea of equilibrium or stability in mechanical systems.

The difficulties that might arise with equilibrium outcomes are illustrated by the following two 2×2, nonzero-sum, two-person games known as the prisoner's dilemma and chicken. These are the driving forces behind escalation (arms races and price wars) and confrontation, respectively. In these two games each player has two strategies: to compromise C or to defect D. The resulting payoffs are indicated in Figs. 2, 3 where it is assumed that each player prefers the outcome of 4 over 3 over 2 over 1. The payoffs in these tables give a pair of numbers (a, b) where a is the payoff to player I (the row player) and b is for player II (the column player). For example, if players I and II select the strategies D and C, respectively, in Chicken (Fig. 3) they obtain the respective payoffs of 4 and 2.

In either of these games, the best overall outcome for the two players when taken together is the strategy pair (C,C) where each compromises and in turn receives the second best payoff of 3. This would be the likely outcome if these games were played

		Player II C	Player II D
Player I	C	(3,3)	(1,4)
	D	(4,1)	(2,2)

Game Theory, Fig. 2 The prisoner's dilemma

		Player II C	Player II D
Player I	C	(3,3)	(2,4)
	D	(4,2)	(1,1)

Game Theory, Fig. 3 The game of chicken

cooperatively. This result, however, is not in equilibrium. Either player can achieve the higher payoff of 4 if the player alone were to switch from strategy C to D. In the prisoner's dilemma the dominant strategy for each player is D. One does better individually by selecting D, no matter what the other chooses. This leads to each receiving 2, their second worse payoff. In chicken the two (pure) strategy pairs (C,D) and (D,C) both lead to an equilibrium result. No one player can switch strategy and do better in either case. However, these two outcomes are not inter-changeable. If both players select D in an attempt to reach the particular equilibrium that would pay them 4, then the resulting strategy pair (D,D) leads to their worst payoffs (1,1). Of the 78 possible static 2×2 games, these two are the most troublesome.

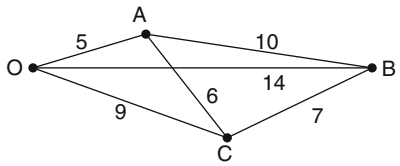
Some noncooperative games have no equilibrium in pure strategies. In 1950 John F. Nash extended von Neumann's minimax theorem for two-person, zero-sum games to prove that every finite multiperson, general-sum game has at least one equilibrium outcome in mixed strategies. Algorithms to calculate equilibria involve nonlinear techniques and often use path-following approaches that may be approximate in nature. There are also many refinements and extensions of the idea of equilibrium described here, and these concepts are fundamental to quantitative approaches in modern economics and politics, as well as system analysis and operations research.

The 1994 Nobel Memorial Prize in the Economic Sciences acknowledged the role of noncooperative game theory. It honored Nash for his fundamental theoretical contributions, as well as some important extensions. Later developments included the repeated play of games with incomplete information introduced by John C. Harsanyi, and aspects of dynamical interaction and evolutionary stability of equilibrium introduced by Reinhard Selten. The 2005 Nobel Prize was again awarded to two game theorists, Robert Aumann and Thomas Schelling, for contributions of game theory to understanding conflict and cooperation.

Cooperative Games

If the players in a game are allowed to cooperate, they typically agree to undertake joint action for the purpose of mutual gain. In this case coalition formation is a common activity, and the additional worth that can accrue to any potential coalition is of primary interest. In practice, the players often solve some optimization problem or consider some noncooperative game in order to arrive at the amount of additional value available from cooperation. The problem that remains concerns how this newly obtained wealth will be, or should be, divided among the players. This latter aspect is again a competition as each participant seeks to maximize their own gain. This may involve negotiations, bargaining, threats, arbitration, coalitional realignments, attempts to arrive at stable allocations or coalition structures, as well as appeals to different ideas about fairness. It is thus not surprising that several different models and solution concepts have been proposed for multiperson cooperative games.

The first general model and idea of a solution for the multiperson cooperative game was presented in the monumental book by John von Neumann and Oskar Morgenstern in 1944 (3rd ed., 1953). Their approach is referred to as the n-person game in characteristic function form. One begins with a set $N = \{1, 2, \ldots, n\}$ of n players who are indicated by 1, 2, …, and n. A characteristic function v assigns a value $v(S)$ to each subset S of N. This number $v(S)$ represents the worth achievable by the coalition S, independent of the remaining players in the complementary set $N - S$. In this context they proposed a notion of solution that they called a solution and which is often referred to now as a stable set. Stable sets proved to have some



Game Theory, Fig. 4 A cost allocation game

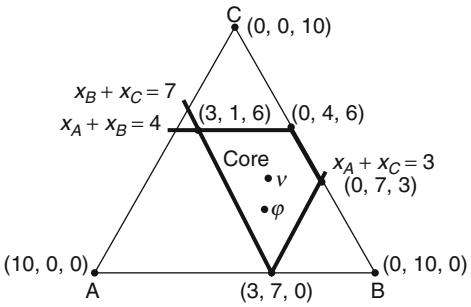
difficulties of both a theoretical and practical nature. They are also rather mathematically involved, and, thus, are not presented here. They are, nevertheless, still a useful tool, especially for the class of games for which the core (see below) is nonexistent. Dozens of alternate solution concepts have since been proposed for these coalition games, and five of these have received the most attention. Three of these solution concepts will be described in the context of the following three-person illustration.

Three neighboring towns A, B, and C plan to tap into an additional water source at O. The costs (in \$100,000) for installing the alternate segments of water pipe appear on the edges in Fig. 4. The joint costs for the various subsets of the three-person coalition $\{A, B, C\}$ are obtained by finding the minimal cost spanning tree for each such coalition. The total cost for the coalition $\{A, B, C\}$ is $c(ABC) = 18$ and it is realized by the link $OACB$. Similarly, the minimal costs for the six other coalitions are:

$$c(AB) = 15 \text{ via } OAB,\ c(AC) = 11 \text{ via } OAC,$$
$$c(BC) = 16 \text{ via } OCB,\ c(A) = 5 \text{ via } OA,$$
$$c(B) = 14 \text{ via } OB, \text{ and } c(C) = 9 \text{ via } OC.$$

(Expressions like $c(\{A, B, C\})$ and $v(\{A, B, C\})$ are shortened to $c(ABC)$ and $v(ABC)$, respectively.)

One can reformulate this problem in terms of the savings available by means of cooperation. Each coalition considers what it saves in a joint project over what it would have cost its members if they were each to make a separate connection to the source O. This savings game has the following characteristic function: $v(ABC) = 10$ [$= c(A) + c(B) + c(C) - c(ABC)$], $v(AB) = 4$, $v(AC) = 3$, $v(BC) = 7$, and $v(A) = v(B) = v(C) = 0$. The three towns can save $10 \times \$100{,}000 = \$1{,}000{,}000$ by acting together. The problem is: how should these savings be allocated to the individual towns? How does one select the three numbers (x_A, x_B, x_C) in the imputation set determined by the relations $x_A + x_B + x_C = 10 = v(ABC)$, $x_A \geq 0 = v(A)$, $x_B \geq 0 = v(B)$, and $x_C \geq 0 = v(C)$? These points are pictured by the large triangle in Fig. 5.



Game Theory, Fig. 5 Solutions to the cost game

One solution concept for cooperative games is called the core. The core consists of those allocations in the imputation set for which every coalition S receives or exceeds its value $v(S)$. No coalition has the capability to improve its total allocation at a core point by going off on its own. For the savings game, the core consists of all imputations (x_A, x_B, x_C) that satisfy the inequalities $x_A + x_B \geq 4 = v(AB)$, $x_A + x_C \geq 3 = v(AC)$, and $x_B + x_C \geq 7 = v(BC)$. This is the four-sided region in Fig. 5. Note that the core is not a unique allocation, and for some games it can be the empty set. However, the core is always non-empty for such cost allocation games.

Another popular solution concept is called the nucleolus. The nucleolus is the one allocation in the center of the core. For the savings game, the nucleolus is the imputation $v = (6/4, 19/4, 15/4)$. (The nucleolus is also defined for those games with empty cores as the unique imputation where the core would first appear when each proper coalition S has its value $v(S)$ decreased uniformly.) If the nucleolus for the savings game is translated back to a cost allocation for the original problem, the following allocation is obtained:

$$\$100{,}000[(5, 14, 9) - v]$$
$$= (\$350{,}000; \$925{,}000; \$525{,}000).$$

In 1951, Lloyd S. Shapley introduced a solution concept that also provides a fair and unique outcome for the savings game. The Shapley value in general gives the average of each player's marginal contribution taken over all possible orderings of a set N of n players. Each one of the $n!$ orderings

(permutations) is a way the full coalition N could build up, one player at a time. There are six orderings of the three towns A, B, and C: (CBA), (BCA), (CAB), (BAC), (ACB), and (ABC). The Shapley value φ_A for town A is accordingly $6\varphi_A = 2[v(ABC) - v(BC)] + [v(AB) - v(B)] + [v(AC) - v(C)] + 2[v(A) - 0] = (2{\cdot}3) + 3 + 4 + (2{\cdot}0) = 13$. A similar calculation gives $6\varphi_B = (2{\cdot}7) + 4 + 7 + (2{\cdot}0) = 25$ and $6\varphi_C = (2{\cdot}6) + 3 + 7 + (2{\cdot}0) = 22$. The Shapley value φ for the savings game is $\varphi = (\varphi_A, \varphi_B, \varphi_C) = (13/6, 25/6, 22/6)$. Note that this point is in the core of this game, although this is not always the case for cost allocation problems. In the original cost problem, φ corresponds to the result ($283,333; $983,333; $533,333).

The various solution concepts for multiperson cooperative games have been applied throughout economics, political science, and operations research. The core is important in the study of economic markets. The nucleolus is viewed as a fair outcome for many bargaining situations. The Shapley value has also been employed as a measure of power for voting systems, where the core is typically the empty set.

Dynamic Games

All previous examples used so far have involved static games. A dynamic game can be thought of as a static game that is repeated for a finite or infinite amount of time. A similar equilibrium concept for a dynamic game can be developed by extending the idea for a static game. The central issue in all dynamic games is credibility. Since players may analyze previous moves and payoffs, strategies trying to predict the other players' moves become very important. Hence, non-credible players often lose out. Since many static games make up a dynamic game, the concept of a subgame equilibrium becomes useful as a means for equilibrium in a dynamic game. A set of strategies for a dynamic game is said to be in subgame equilibrium if all subgames (static portions and their combinations) are in equilibrium. Dynamic games can be played with perfect or imperfect information and can be cooperative or noncooperative.

Consider the finite dynamic version of the Prisoner's Dilemma, which is the static Prisoner's Dilemma repeated N times, and both players know that the game will end after this. Again, as in the static version, the best overall outcome is for both players to play C in each of the N turns. However, as in the static game, this outcome is not in equilibrium. As an example of an optimal deviation, suppose both players play C ($N - 1$) times. In the final move, player I can choose to deviate to D and get a total payoff of $3(N - 1) + 4$ which would have been more than his or her payoff if he or she would have played C for the final move. Player 2 has a similar deviation, in fact this deviation need not take place at the last move and could be played at any stage. One subgame equilibrium of the finitely repeated Prisoner's Dilemma is that both players play D for all the N moves. Clearly, no player has any optimal deviation at any one point in the game. However, to prove this is an actual subgame equilibrium (so that there exist no sequence of strategies that is an optimal deviation from playing D in all moves), a technique called backward induction is used.

After the last turn, no further interaction will be possible. Hence, in the final turn, both players choose the dominant outcome D, and play the equilibrium for the static game. Knowing that they both will defect in the last period, both players play D in the second to last period as well, since they know no matter what happens, the last turn will involve both of them defecting. Hence, using this inductive procedure, it's easy to prove that both players playing D for all periods is a subgame equilibrium. This process of starting from the final move and moving backwards is known as backward induction. Note that backward induction is only a valid technique in a finite dynamic game, as there is no final move in an infinitely repeated game.

Dynamic games can also involve moving sequentially. In a Stackelberg Duopoly, two players choose quantities of production to maximize individual profits. However, one player, the leader, gets to move first and the second player, the follower, moves after observing the first player's move. In the Stackelberg game, the leader often has a higher payoff in a subgame equilibrium. Using backward induction, for a simple finite Stackelberg game, it is easy to figure out the subgame equilibrium. A Stackelberg game is also an example of a mathematical program with equilibrium constraints.

For infinitely repeated dynamic games, the analysis tends to be slightly different. Backward induction can no longer be used to find and prove that a strategy is a subgame equilibrium. To study infinitely repeated games, a discount factor (less than or equal to one) is

introduced to discount future payoffs. Strategies known as trigger strategies comprise equilibria of such games. One widely studied strategy is a tit-for-tat strategy, in which a player cooperates until the opponent stops cooperating. Then the player does not cooperate until the opponent starts cooperating again. Another trigger strategy is a grim strategy, where a player cooperates until the opponent stops cooperating, after which the player proceeds to not cooperate forever. For example, in an infinitely repeated Prisoner's Dilemma, subgame equilibria can be different from both players playing D for all turns. Various grim strategies and tit-for-tat strategies can be devised as subgame equilibria depending on the value of the discount factor.

Dynamic games without perfect information can also be studied. In such games, there is a sender of information and a receiver. The sender observes their type, and sends a message to the receiver about it. The receiver chooses an action after predicting what the senders type will be based on the message received. An example is Spence's (1976) model of job market signaling. A job applicant (sender) knowing about their productive ability (type) sends details regarding their education (message) to an employer or a market of employers (receiver). The wage paid by the employer is then the action of the receiver.

One of the problems with traditional game theory, as outlined above, is that it often does not mirror the actual decisions by human beings and institutions. To achieve equilibrium, players are assumed to be extremely rational and have perfect foresight. Many experiments show that this is not true. Another shortcoming is the presence of multiple equilibria in game theory. There is no consistent method for determining which equilibrium is picked when the game is played out in practical life with real players. The next section provides a brief overview of how recently developed theory has attempted to resolve these problems.

Evolutionary Game Theory

Evolutionary Game Theory is the study of dynamic games that focuses on strategy development by players who interact, usually as part of a large population. By looking at strategy development as opposed to finding equilibrium points, evolutionary game theory overcomes the problem of selection from multiple equilibria. It provides a framework for studying how strategies evolve and which equilibrium point gets picked. Also, by studying strategy, evolutionary game theory does not assume that the players are excessively rational. Fisher (1930) was the first to use evolutionary game theory to understand why some species of mammals have an equal sex ratio even though the majority of males never mate. At other instances, evolutionary game theory has done well to model the behavior of animals, which are assumed to be even less rational than human beings.

The first mainstream introduction of evolutionary game theory was made by Smith and Price (1973) when they published "The logic of animal conflict." While it was of great interest to evolutionary biology, many economists also recognized its use in modeling human behavior. Axelrod (1984) was one of the first economists to apply this theory to cooperation among human players. The essential features of all models in evolutionary game theory is repetition of player moves and the adaptive behaviors and strategies resulting from these repetitions in large populations. A strategy that can withstand changes, or mutations, is regarded as an evolutionary stable strategy. This helps biologists understand the decision making of animals and organisms and helps social scientists extract information about decision-making processes in humans.

The Prisoner's Dilemma also has an evolutionary counterpart. Consider the dynamic Prisoner's Dilemma played repeatedly by a large population. It can be shown that all players cooperating is an unstable equilibrium, in that if a small percentage of the population deviates from cooperating, the dynamics will drive the entire population to defect. Thus defecting in the evolutionary version of the Prisoner's Dilemma is a stable equilibrium. There are evolutionary versions of signaling games and other dynamic games as well. Again, evolutionary game theory analyzes the process of choosing equilibria as opposed to finding them, which is the focus of traditional game theory.

See

- ▶ Decision Analysis
- ▶ Duality Theorem
- ▶ Prisoner's Dilemma
- ▶ Utility Theory

References

Aumann, R. J., & Hart, S. (Eds.). (1992, 1994, 1995). *Handbook of game theory: With application to economics* (Vols. 1, 2 and 3). Amsterdam: North-Holland.

Axelrod, R. (1984). *The evolution of cooperation*. New York: Basic Books.

Fisher, R. A. (1930). *The genetic theory of natural selection*. Oxford: Clarendon Press.

Gibbons, R. (1992). *Game theory for applied economists*. Princeton, NJ: Princeton University Press.

Harsanyi, J. C. (1967–1968). Games with incomplete information played by bayesian players. *Management Science*, *14*, 159–182, 302–334, and 486–502.

Lucas, W. F. (1971). Some recent developments in n-person game theory. *SIAM Review, 13*, 491–523.

Lucas, W. F. (1995). The 50th anniversary of TGEB. *Games and Economic Behavior, 8*, 264–268.

Luce, R. D., & Raiffa, H. (1957). *Games and decision*. New York: Wiley. Reprinted by Dover 1989.

McDonald, J. (1975). *The game of business*. Garden City, NY: Double-day. Reprinted by Anchor, 1977.

Ordeshook, P. J. (Ed.). (1978). *Game theory and political science*. New York: New York University Press.

Selten, R. (1975). Reexamination of the perfectness concept for equilibrium points in extensive games. *International Journal of Game Theory, 4*, 25–55.

Smith, J., & Price, G. (1973). The logic of animal conflict. *Nature, 246*, 15–18.

Spence, A. M. (1973). Job market signaling. *Quarterly Journal of Economics, 87*, 355–374.

von Neumann, J., & Morgenstern, O. (1953). *Theory of games and economic behavior* (3rd ed.). New Jersey: Princeton University Press.

Williams, J. D. (1954). *The compleat strategyst (sic)*. New York: McGraw-Hill. Revised edition, 1966; Dover edition, 1986.

Gaming

William Schwabe
RAND Corporation, Santa Monica, CA, USA

Introduction

Abt (1970) broadly defined a game as "an activity among two or more independent decision-makers seeking to achieve their objectives in some limiting context." Gaming involves the activity itself, whereas game theory uses mathematics to seek the best strategies, the sets of decisions that decision-making players might make.

Games are played for entertainment, sport, teaching, training, and research. As a research method, gaming is used by psychologists, educators, and sociologists interested in how people learn and play games and by operations researchers, other analysts, and decision makers interested in developing, exploring, and testing policies, strategies, hypotheses, and other ideas.

As an OR/MS method, gaming is controversial, often practiced more as an art than a science. Few methods have been so inadequately named, prompting ridicule from skeptics and attempts by adherents to call it something more serious sounding or descriptive, such as operational gaming, simulation gaming, free-form gaming, and, in defense analysis, war gaming and political-military gaming. Although gaming has not been made as scientifically rigorous nor as universally accepted as adherents hoped for decades ago, it has helped importantly in developing strategy, in pretesting policies before actual implementation, and in communicating understanding of operational complexities.

Research games are often played as part of the planning process in developing important policies and strategies for organizations in competitive situations. Accordingly, the results — and sometimes the existence of gaming — are not publicized. For example, several Iraq-Kuwait scenarios were gamed in 1990 before Iraq actually attacked, but they are not fully documented in the open (unclassified) literature. Sources of information on research gaming include special interest sections at operations research/management science professional conferences, reports and bibliographies published by organizations with a tradition of gaming (such as the Naval War College, RAND, and others), the journal *Simulation & Games*, and various books and articles. Shubik (1975) presents comprehensive discussions of gaming, including a game theory background for gaming, analytical, and behavioral models, and examples of games used for a variety of purposes. Brewer and Shubik (1979) provides an historical review of the use of military war games. Dunnigan (2000) details the designing and playing of commercial and professional war games.

Learning from Gaming

People can learn from gaming by designing the game, by playing it, or by analyzing the play or

results (Perla and Barrett 1985). Greenblat (1988) discusses game design as a five-stage process: (1) setting objectives of and constraints on the game, (2) conceptual model development, (3) decisions about representation, (4) construction and refinement, and (5) documentation. Because a game is meant to model one or more important aspects of something that is operationally complex, game design is usually an intense intellectual exercise in analysis. Analysts commonly learn a great deal from the process of designing a game, as is true with other types of model design. Gredler (1994) discusses how Lewinian or Piagetian theories of learning can be applied to game design.

Most games have two or more teams, each representing a decision-making entity, such as a country, a military command, or a business firm, with from one to hundreds of players on a team. Players may be assigned specific roles — a leader of a country, a CEO in a regulated industry, a local warlord — in which they can criticize or embrace the policies of their own or competing governments or organizations. Formal games have rigid rules for play, while seminar or free-form games, have few rules.

Play of a game is usually divided into moves, each being a period of real time during which game time (often posited to be in the future) is assumed to be frozen. Game time is usually advanced further into the future between successive moves; however, it can be advantageous to roll back game time for the final move, to allow players to apply what they learned about possible future consequences in formulating better near-future policies, initiatives, or options (Molander et al. 1996).

Moves usually begin with teams being presented with information that players are asked to accept as true for the purposes of the game and use as a basis for their deliberations and decisions. This information is often in the form of a scenario, a story about how the future might plausibly evolve, carefully crafted to help people "recognize and adapt to changing aspects of our present environment" (Schwartz 1991). The set of decisions made by a player or team during a move period is sometimes called its move. Game administrators, who usually include researchers who designed the game and will analyze its results, are commonly called controllers or referees. Games have usually been played with all participants at one site; however, distributed games can be played with remotely located players communicating via electronic mail or other means.

Despite the make believe aspects of gaming, players often become intellectually (and sometimes emotionally) caught up in the game, engaging in intense, goal-focused thought and discussion. In the process they learn about the issues, about their teammates, and about themselves. Controllers often learn what can go wrong operationally and how signals and other forms of communications between teams can be misunderstood.

Analysis usually begins with a critique at the end of the last game move, attended by players, controllers, and observers. The game director may ask each team leader to present their analysis of what the team saw as the major issues, how they analyzed their options, what they decided, and what results they expected. Whether analysis is more formal than this depends in part on whether the game was designed as an experiment, to yield data or other observations suitable for analysis. If a series of games is played, then there is opportunity for comparative analysis.

Why Game?

Unlike many other techniques of analysis, gaming is not a solution method. The output of a game is not a forecast or prediction, solution, or rigorous validation. The output of a good game is increased understanding.

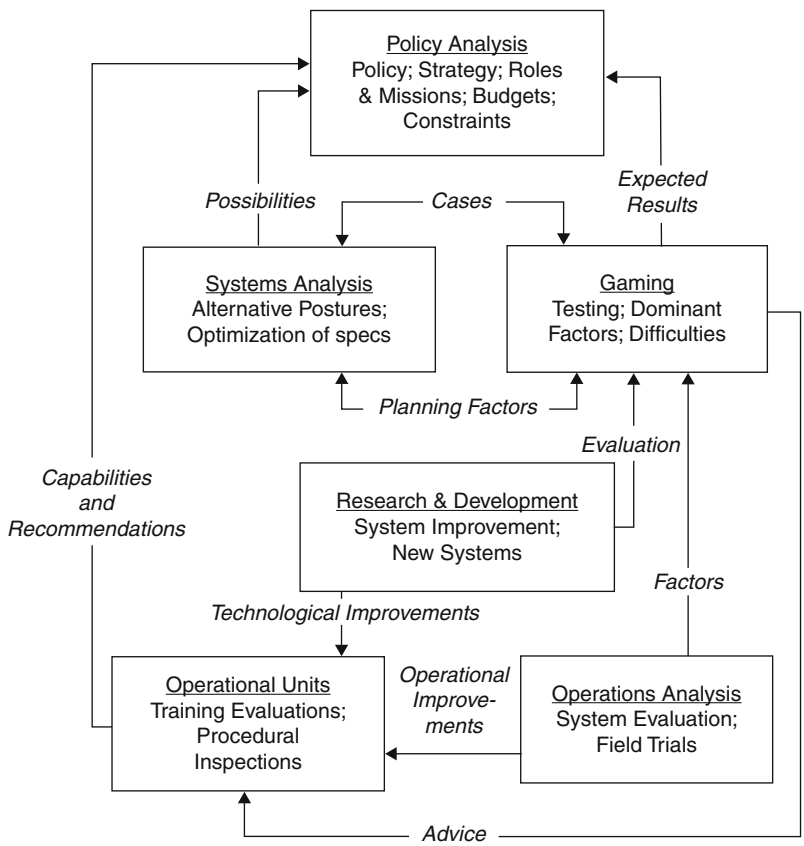
Gaming can do several things: reveal errors or omissions in concept; explore assumptions and uncover the implicit ones; draw out divided opinion; examine the feasibility of an operational concept; identify areas that are particularly sensitive or in which information is lacking (Quade 1975); pool the knowledge of several experts; suggest questions or hypotheses for further study; identify the values or measures of effectiveness (MOEs) that people care about; breadboard approval-winning or implementation of policies; or test strategies for long-term consequences. No matter how rigorous the analysis, gaming can do something an individual cannot do, namely, list the things that would never occur to the individual. It can help identify all the ways that a carefully composed statement can be misinterpreted. As Levine, Schelling, and

Jones (1991) noted, it can also "generate the phenomena of understanding and misunderstanding, perception and misperception, bargaining, demonstrations, dares and challenger's accommodation, coercion and intimidation, conveyance of intent, and uncertainty about what each other has already done or decided on. There are some things that just cannot be done by a single person or by a team that works together."

Prospects for Gaming

The popularity of gaming is cyclical. Its use, however, appears to be on an upward path. Video conferencing and electronic mail networks open possibilities for less expensive games with broader participation, including international play. Advances in computers and software make it easier to develop models to support games, to use them on the fly during games to update scenarios, to query data files in response to player questions during games, and to prepare presentation graphics during the games and for post-game critiques. Videotaping has been used to present scenario updates to players in newscast format and to present pre-taped briefings by experts to players. Expert systems are used to support some games, but the use of artificial intelligence, rule-based agents in gaming is not as active as it was in the 1980s.

Gaming has often not been as well integrated into studies using other methodologies as might be warranted. Gaming is but one form of analysis to inform policy, managerial, or operational decisions. Figure 1, adapted from Paxson (1963), summarizes some of the relationships between gaming and other analysis. Regardless of whether gaming ever achieves the rigor early proponents sought, it appears to have continuing value. Gaming can often respond to changing operational or strategic contexts more rapidly than other methods. Challenges remain in making games less demanding of player time (especially important in enlisting senior officials



Gaming, Fig. 1 Gaming and analysis relationships

as players), in reducing costs of games (including travel costs), and in using game results responsibly and effectively in analyses to inform decisions.

See

- ▶ Battle Modeling
- ▶ Game Theory
- ▶ Military Operations Other Than War
- ▶ Military Operations Research
- ▶ RAND Corporation
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Abt, C. C. (1970). *Serious games*. New York: Viking.

Brewer, G., & Shubik, M. (1979). *The war game: A critique of military problem solving*. Cambridge, MA: Harvard University Press.

Dunnigan, J. (2000). How to play and design commercial and professional Wargames. In *Wargames handbook* (3rd ed.). Lincoln, NE: Writer's Club Press.

Gredler, M. (1994). *Designing and evaluating games and simulations: A process approach*. Houston, TX: Gulf Publishing.

Greenblat, C. S. (1988). *Designing games and simulations: An illustrated handbook*. Newbury Park, CA: Sage Publications.

Levine, R., Schelling, T., & Jones, W. (1991). Crisis games 27 years later. *Report P-7719*. The RAND Corporation, Santa Monica, CA.

Molander, R. C., Riddile, A. S., & Wilson, P. A. (1996). Strategic information warfare: A new face of war. *Report MR-661-OSD*, The RAND Corporation, Santa Monica, CA.

Paxson, E. W. (1963). War gaming. *Report RM-3489-PR*, The RAND Corporation, Santa Monica, CA.

Perla, P., & Barrett, P. R. T. (1985). An introduction to Wargaming and its uses. *Report CRM 85-91*, Center for Naval Analyses, Alexandria, VA.

Quade, E. S. (1975). *Analysis for public decisions*. New York: Elsevier.

Schwartz, P. (1991). *The art of the long view: Planning for the future in an uncertain world*. New York: Currency Doubleday.

Shubik, M. (1975). *Games for society, business and war: Towards a theory of gaming*. New York: Elsevier.

Gamma Distribution

A continuous random variable is said to have a gamma distribution if its probability density can be written in the form $f(t) = a(at)^{b-1} e^{-at}/\Gamma(b)$ where a and b are any positive real numbers and $\Gamma(b)$ is the gamma function evaluated at b. The constant b is called the shape parameter, while a (or various equivalents) is called the scale parameter. If b happens to be a positive integer, then $\Gamma(b) = (b-1)!$ and this gamma distribution is also called an Erlang distribution. Furthermore, if b is either an integer or half-integer (1/2, 3/2, etc.) and $a = 1/2$, the resultant gamma distribution is equivalent to the classical χ^2 distribution of statistics.

See

- ▶ Erlang Distribution

GAMS

General Algebraic Modeling System. An algebraic modeling language for mathematical programming that supports numerous commercial and open source software solvers, including BARON, COIN-OR, CPLEX, Gurobi, MINOS, SNOPT and KNITRO.

Gantt Charts

Steven Nahmias
Santa Clara University, Santa Clara, CA, USA

Introduction

There are three well-known types of Gantt charts: the Gantt load chart, the Gantt layout chart, and the Gantt project chart. A Gantt chart is essentially a bar chart laid on its side. The horizontal axis corresponds to time and the vertical axis to a collection of related activities, machines, employees, or other resource. Bars are used to represent load durations or activity starting and ending times. The Gantt chart is appealing in that it is easy to interpret and can provide a visual summary of a complex schedule.

In principal, the three types of Gantt charts are similar, but each has a somewhat different application. The load chart is used to show the

amount of work assigned to resources (typically equipment) over a given amount of time. Sequencing issues are ignored here. Load charts are useful for showing work assigned to a project, but do not show the progress of an on-going project. The Gantt layout chart is used to block out reserved times on facilities and is one means of keeping track of the progress of an ongoing project.

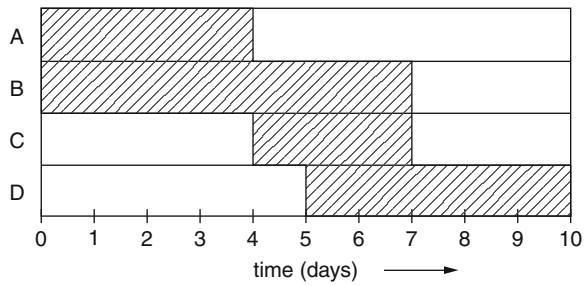
The most popular type of Gantt chart is the Gantt project chart. A Gantt project chart is used to show the starting and ending times of all the activities comprising a project. It can be used to monitor the progress of a project and determine where stumbling blocks may be. Next is an example of a Gantt project chart.

Example

Suppose that a project consists of four activities: A, B, C, and D requiring, respectively, 4, 7, 3, and 5 days. Figure 1 is a Gantt chart representing the starting and ending times for these activities.

According to this chart, A and B are started at day 0 and are scheduled to be completed, respectively, at the start of days 4 and 7. Activity C is begun when A ends on day 4 and is completed at the start of day 7, while D is started on day five and completed on day 10. While the chart shows the start and finish times of each activity, it has the shortcoming of not showing precedence relationships. Specifically, is C required to wait for the completion of A or could C have been scheduled earlier? Does D require A to be completed before it could start? Should D have been started on day 4 instead of day 5? Because of this significant limitation, years later professionals recognized that networks were much more powerful ways of representing projects, since precedence constraints could be incorporated directly into a network structure. Both the Critical Path Method (CPM) and Project Evaluation and Review Technique (PERT) overcome this shortcoming of Gantt charts. Even though it has limitations as a planning tool, the Gantt chart is still one of the most convenient ways to represent a schedule once it is determined.

While Gantt charts vary considerably in format and structure, this example contains the basic elements of all Gantt project charts. Invariably, the horizontal axis corresponds to time, and the vertical axis to a set of



Gantt Charts, Fig. 1 Four-activity Gantt chart

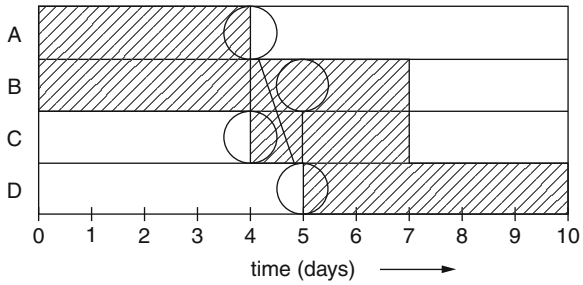
activities (or machines or resources for the other types of Gantt charts). Interpret activities very broadly here. They may be parts of a project, they may be part numbers, or machines or personnel. The bars generally correspond to beginning and ending times for activities, but might have different interpretations in other contexts. For example, they may correspond to a work shifts for personnel or delivery and shipment times for parts.

Implementation Issues

There are some issues one must be concerned with when trying to implement a Gantt chart. One is the way time is measured and scaled. In the example above, time is shown in numbers of elapsed days from an arbitrary day labeled day 0. In practice, it is more common, however, for time to be measured in calendar days. The horizontal axis would correspond to specific calendar dates. While calendar dating makes starting and ending times more explicit, there are other issues to consider as well. How long is a day? In most work environments, a work day is 8 hours. In other contexts, a day may be 24 hours. Another issue is whether operations continue during weekends. There are several ways to handle this problem. The easiest is just to exclude weekend dates from the chart. The interested reader should refer to Battersby (1967) and Clark (1952) where these and related issues are discussed in detail.

Extensions

An extension of the Gantt project chart which was the precursor of modern networks is the milestone chart.



Gantt Charts, Fig. 2 Milestone Gantt chart

Networks are collections of nodes and directed arcs. In the context of project planning, nodes represent completion of a collection of activities, and directed arcs to the durations of specific activities. Developed in the 1940s by the U.S. Navy, the milestone chart is a Gantt chart with circles representing key time periods which occur during the completion of an activity. The milestones could then be linked, in much the same way that nodes are linked on a project network. An example of a milestone Gantt chart is shown in Fig. 2.

The vertical line linking Activities A and C means that C cannot be started before A is complete. In the case of activities B and D, the vertical line there implies that D cannot be started until five days after B has started. (This suggests that B should be represented as two activities.) The diagonal line connecting A and D means that D cannot be started until A is completed.

Uses

The Gantt chart was the precursor to several commercial graphical control systems, many of which are found today adorning the walls of manufacturing facilities throughout the United States. In his classic work, Moore (1967) noted several commercial variations of the Gantt chart available in the 1960s including Productrol boards, Schedugraphs, and Boardmasters. All use a time scale across the top and horizontal lines to picture machines, schedules or orders or whatever is being graphed. Although very popular in the 1950s, these manual techniques have lost favor because computers can quickly update and print progress charts.

History

The concept was originally developed by Henry L. Gantt, a contemporary of Frederick Taylor's, a major force in the development of scientific methods for operations and production control. Gantt developed the idea of a bar chart to monitor project status while he was affiliated with the Army Bureau of Ordnance during World War I. His original intent was to display graphically the status of munitions program for that day. Gantt recognized that time was a key variable against which the progress of a program could be assessed. Gantt's development was certainly a first key step in the development of scientific methods for project management, a powerful tool for project planning. Both CPM and PERT were consequences of the kind of planning recommended by Gantt. Dozens of texts have been written on the subject and the methods have been applied to a large variety of industries. Personal computer software products are widely available which make extensive use of Gantt charts to display schedules. An overview of project planning is given in Nahmias (2009), and more details on project planning techniques can be found in Moder, Phillips, and Davis (1983).

See

- Critical Path Method (CPM)
- Network Planning
- Program Evaluation and Review Technique (PERT)

References

Battersby, A. (1967). *Network analysis for planning and scheduling* (2nd ed.). London: Macmillan.

Clark, W. (1952). *The Gantt chart: A working tool for management*. New York: Pitman Publishing.

Moder, J. J., Phillips, C. R., & Davis, E. W. (1983). *Project management with CPM, PERT, and precedence diagramming* (3rd ed.). New York: Van Nostrand Reinhold.

Moore, F. G. (1967). *Manufacturing management* (4th ed.). Burr Ridge, IL: Irwin/McGraw-Hill.

Nahmias, S. (1997). *Production and operations analysis* (3rd ed.). Burr Ridge, IL: Irwin/McGraw-Hill.

Nahmias, S. (2009). *Production and Operations Analysis* (6th ed). McGraw Hill.

Gaussian Elimination

A computational procedure for reducing a set of $(m \times m)$ linear equations $\boldsymbol{Ax} = \boldsymbol{b}$ to the form $\boldsymbol{MAx} = \boldsymbol{Mb}$, where $\boldsymbol{MA} = \boldsymbol{U}$ is an upper triangular matrix. The variables of the solution vector are found by solving the resulting triangular system for one variable in the last equation, and back-substituting in the next to last equation, and so on. Some form of elimination is central to the simplex method for solving linear-programming problems.

See

- ▶ Matrices and Matrix Algebra
- ▶ Simplex Method (Algorithm)

Gauss-Jordan Elimination Method

A computational procedure for reducing a set of $(m \times m)$ linear equations $\boldsymbol{Ax} = \boldsymbol{b}$ to the explicit solution form of $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$.

See

- ▶ Gaussian Elimination

Gene

In genetic algorithms, the unit of inheritance, carried by chromosomes (i.e., solutions); a piece of the genetic material that determines the inheritance of a particular characteristic.

See

- ▶ Genetic Algorithms

Generalized Erlangian Distribution

The probability distribution of a finite sum of independent, exponentially distributed random variables whose parameters may not be the same. Sometimes, the term is also used for a convex sum of Erlang distributions, which, however, is more often called a mixture.

Generalized Upper-Bounded (GUB) Problem

A linear-programming problem with a set of constraints of the form $\sum_{j \in \boldsymbol{J}} x_j = 1$, where $\boldsymbol{J}$ is a subset of the indices $j = 1, 2, \ldots, n$ and each j can appear at most once in some $\boldsymbol{J}$. This problem is called a GUB problem and a special adaptation of the simplex method is available that reduces the computational burden of having a large number of GUB constraints.

Generating Function

- ▶ Probability Generating Function

Generator (of a Markov Chain/Process)

The matrix of state-transition rates (intensities).

See

- ▶ Markov Chain Equations
- ▶ Markov Chain Monte Carlo
- ▶ Markov Chains
- ▶ Markov Processes

Genetic Algorithms

Probabilistic algorithms from the class of Evolutionary Algorithms.

See

- ▶ Evolutionary Algorithms

Geographic Information Systems

Paul Gray[1], Thomas A. Horan[1] and James B. Pick[2]
[1]Claremont Graduate University, Claremont, CA, USA
[2]University of Redlands, Redlands, CA, USA

Introduction

The most widely known and used form of a Geographic Information System (GIS), one everyone takes for granted, is the navigation system in our cars - Global Positioning System (GPS) - that shows where you are and tells you how to get to where you are going. Google Maps and Google Earth, two other GIS examples, became standards almost overnight in 2005. Geographic Information Systems, however, are more pervasive and important than that gadget in your car or the free maps available on the Internet; they have millions of users. GIS software combines database management systems, map layers, and visualization to support cartographic display, spatial query, and analytical modeling. They integrate locational, topological, and thematic data to allow constructing, exploiting, and visualizing complex spatial relations among data types at a variety of scales, levels of aggregation, and dimensionality. As such, they are a fundamental tool for Operations Research (OR).

Geographic information technology allows exploiting increasingly available amounts of valuable geographic and spatial data in a form easily comprehended by analysts and decision makers. Among the problems to which GIS has been applied are:

- site selection;
- urban 3D design;
- election administration and redistricting;
- infrastructure management;
- mapping and modeling of Federal geospatial information, such as that of Department of Homeland Security, Department of the Interior, Department of Housing and Urban Development, and Bureau of the Census;
- natural resource exploration;
- public health and safety (e.g., modeling communicable diseases; analyzing natural disasters before, during, and after; emergency response planning; and optimizing dispatch of emergency vehicles);
- real estate marketing, sales, and management;
- renewable energy management (wind, solar, hydro, and geothermal);
- military and defense applications combining GIS with sensors and satellite imagery;
- transportation, fleet management, supply chain, and other logistics; and
- urban and regional planning, including modeling and analysis of cities, populations, urban sprawl, taxation, land use, zoning, utilities and urban 3D design.

GIS Background and Capabilities

The term geographic information system was coined in the 1960s. It now applies to generic capabilities for studying and analyzing spatial phenomena. (Longley et al. 2010; Clarke 2010). They define its capabilities as including:

- collecting, storing, and retrieving spatial location data;
- identifying locations which meet specified criteria;
- exploring relationships among spatial layers and associated data sets;
- analyzing related spatial data to aid in making decisions;
- facilitating assessment of alternatives and their impacts;
- displaying selected environments both visually and numerically.
- modeling spatial phenomena;
- designing solutions to problems based on spatial analysis and visualization.

Databases and Computer Advances

One of the most powerful examples of the essential role of GIS as an earth systems analytic tool is in the efforts to understand the processes of global change. GIS tools combine ecosystem and biological data, meteorological information, distribution of manmade emissions sources, regulatory data, site monitoring data, satellite and aircraft sensing and imaging data. They use the integrated information to simulate the

interactions and impacts of complex natural and manmade systems. The analytic potential of GIS tools is enhanced by the power of GIS visualization. Researchers and practitioners view modeling results in a realistic terrain rendering that facilitates understanding and provides a valuable mode for presenting research results to non-technical audiences (Maguire et al. 2005).

The key to GIS application and the impetus behind its fast growing use is the combination of the availability of massive databases and the continually increasing power of mobile, desktop, and server computer hardware that enable storing and manipulating these databases. The databases are increasingly being stored in large commercial and governmental spatial server farms, such as those of ESRI Inc., Google, and the U.S. Geological Survey. An example of a particularly important national base map is the TIGER files (Topological Integrated Geographic Encoding and Referencing System) located at the U.S. Census Bureau. TIGER is essential to systemizing the many basic geographic layers of government geographies such as census tract boundaries, highways, national boundaries, and administrative districts, so that demographic and economic data can be visualized precisely. The TIGER files are updated and corrected regularly.

Software

Specialized software links attributes (discrete object values) superimposed 2D map layers and higher dimensional geometries through common identifiers. This analysis function permits displaying multiple spatial relationships:

- overlaying combinations of features and recording resulting conditions;
- analyzing networks;
- defining areas in terms of specified criteria; and
- fitting models more accurately through spatial statistics

In practice, for most earth surfaces, a GIS consists of a series of layers, each presenting a particular feature that can be superimposed accurately on top of one another. Each feature constitutes a distinct layer that can be displayed or not, as desired by the analyst. Basic features take the form of points, lines, or polygons, representing the full spectrum of spatial phenomena. For example, lines may represent transportation options, such as railroads or highways. Because the typical map contains vector-based longitude and latitude data, distances traversed along a route are easily calculated and available for inclusion in analysis. A GIS is able to recognize 3D geometry and perform spatial analysis of these more complex forms, for example, to model shadows on the urban landscape from new skyscrapers being built in cities.

A feature supporting analysis of spatial distributions is electronic street addresses, either defined for the ends of blocks or positioned at regular intervals along major thoroughfares. The GIS software typically contains interpolation algorithms enabling the analyst to pinpoint specific phenomena by address, termed geo-coding. There is a competitive industry marketing software and databases products to transform street addresses, or nine-digit zip codes, into map coordinates. In marketing applications, for example, this capability is used to locate customers or potential customers. In school districting, the location of students can be represented.

The display of variable densities or thematic mapping displays (e.g., expenditures, activity levels, or incidents by selected administrative districts) is a standard GIS software capability. For example, the incidence of people aged 60 and over, based on census data, can be displayed by location representing age categories by different gray scales or colors.

The primary mode of building GIS maps is based on vector models using points, lines, and polygons. An alternative approach is to build raster maps in which real world features are shown as pixels on a grid. Each dot becomes a data point in a raster GIS. Most imaging satellites can now create digital/raster images. Although a raster GIS is typically much more hardware and software intensive than vector GIS, it can answer spatial questions that vector GIS cannot. Whereas vector models compute distance with lines that form a network, a raster GIS does not require a line-based network. Thus, a raster GIS can better measure the distances traveled by wolves in Yellowstone National Park in that they do not follow a network.

Raster GIS models are not inherently superior to vector-based models. Both have strengths and weaknesses. They should be considered different

tools to help solve different spatial questions. Historically, a vector GIS was more prevalent than a raster GIS because of the limitations of hardware and software, but advanced GIS software is able to move back and forth seamlessly between vector and raster models. The general user is not even aware that a change takes place. A well known raster GIS product is GRASS (Geographic Resources Analysis Support System) that was initially developed and maintained as public-domain software by the U.S. Army Construction Engineering Research Laboratories. It is now maintained by a private company. The leading GIS products feature both vector and raster capabilities.

In general, the major features of GIS software can be grouped under the headings of user interface, analysis tools, and data management (Longley et al. 2010). The user interface can vary in complexity from dozens of features for client applications such as Google Earth, and up to many hundreds of features in advanced software such as the Web-based ArcGIS. Analysis tools range from traditional simple tools such as query, distance measurement, buffering, and overlays, up to advanced spatial analysis, graphical model building, 3D visualization, space-time modeling, spatial statistics, integrated routing and inventory models, and network analysis (Longley et al. 2010). Data management features include internal tables, linkages to common spreadsheets, and connections to leading relational databases and geo-databases, that are hybrids that combine relational, object, and spatial capabilities. The software can be implemented at the cloud, server, desktop, and mobile levels, and companies now offer families of products that allow these levels to be connected together. For larger firms, enterprise GIS software is offered based on powerful internal and external servers, and can be coupled with business Enterprise Resource Planning software. GIS software markets are increasingly being supported by outsourcing part or all of it.

Spatial Decision Support System (SDSS) for Decision Making

Spatial Decision Support Systems enlarge the standard DSS model of data management, model management, and knowledge management by adding spatial analysis and spatial data components. Thus, an SDSS refines and strengthens conventional DSS analysis by complementing it with spatial analysis and spatial data. SDSSs can be divided into those systems that rearrange existing information and those that generate new information (Nyerges and Jankowski 2010; Longley et al. 2010). To rearrange is to observe data presented in different ways, as in looking at a map of income distribution by census tract to locate retail outlets or target an advertising campaign. New decision-making outcomes can be the result of overlaying regions and spatially analyzing their features in new ways. For example, in site selection, the decision can be affected by layers that define available water, available electricity, school locations, parking availability, and traffic densities as a function of time of day. Such spatial relations are usually not evident in spreadsheet or tabular data.

SDSSs benefit decision making because they provide access to additional information usually in a different format. As a result, managers are provided not only with additional data, but also with more flexibility in which they can view the data The conventional wisdom on the impact of an SDSS is that the political and ethical/moral underpinnings become more explicit, thereby helping decision makers understand the impacts of the choices made.

Earth Systems Analysis

One of the most powerful examples of the essential role of a GIS as an earth systems analytic tool is in the efforts to understand the processes of global change. GIS tools combine ecosystem and biological data, meteorological information, distribution of manmade emissions sources, regulatory data, site monitoring data, satellite and aircraft sensing, and imaging data. They use the integrated information to simulate the interactions and impacts of complex natural and man-made systems. The analytic potential of GIS tools is enhanced by the power of GIS visualization. Researchers and practitioners view modeling results in realistic terrain rendering that facilitates understanding and provides a valuable mode for presenting research results to non-technical audiences (Maguire et al. 2005).

Use of GIS in Urban Planning and Other Policy Making Activities

> Because GIS technology has matured on multiple fronts over the past decades, we are able to consider more of the relevant conditions and impacts when deciding on how to address [a complex systems of human-environment-society] relationships (Nyerges and Jankowski 2010).

Typical planning models tend to be displays of existing facilities, for example, retailing and transportation. The applications tend to involve structured issues, such as allocation, rather than the indefinite set of options policy makers face.

GIS used in urban planning depends upon a series of powerful urban methods and models: population potential, location quotients, grid analysis, network analysis, Markov chains, gravity models, geo-demographic techniques, central place theory, and visual modeling (Nyerges and Jankowski 2010; Greene and Pick 2006). Cities and regions are taken to be complex systems with structures composed of hierarchical subsystems, primarily spatial and nominally static. Planning for these systems consists of optimizing general systems properties, such as idealized population distributions. Difficulties arise from the interactions of systems at the periphery of any region under study.

The early interest in applying computerized capabilities to urban planning issues faced the obstacles of collecting data, task size in terms of data representation, and difficulty in developing appropriate system models (Brewer 1973; Lee 1973). These operational problems coincided with a changing planning philosophy shifting practitioners' interest toward more pragmatic approaches. There is now less emphasis on optimization and more concern with broader-based issues of equity. This transformation in urban planning thrust is evidenced in current demand for data systems for facility location, emergency services planning, resource management and conservation, and property and tax register recordings. Forrest (1990) listed over 60 distinct systems and problem areas to which a GIS might be applied, ranging across such apparently disparate issues as navigation, political redistricting, hazardous waste management, and wildlife protection. GIS systems are toolkits whose designs include enough flexibility to accommodate these multiple dimensions (Nyerges and Jankowski 2010).

In the 1980s, the predominant GIS platform was a stand-alone powerful workstation. It required a substantial personnel and training investment to build a GIS capability. Survey results show it was not until the late 1980s that many city and regional planning departments adopted and used GIS systems (French and Wiggins 1989, 1990). With the advent in the 1990s of powerful, inexpensive desktop computing and software, GIS tools have become a fundamental part of regional planning department infrastructure. In the first decade of the 21st century, GIS platforms moved to mobile and web-based platforms, powered by spatial servers. Modern GIS software has the capability to integrate spatial data and analysis across these different platforms. A GIS is also reinforced by its integration with increasing types of small sensor and locational devices, such as GPS in vehicles, radio frequency Identification, light detection and ranging, and thermal sensors. They allow real-time data to be input into a GIS with the capability for very fast decision making.

Over the last several years, GIS technology has evolved into a platform that encourages direct citizen engagement in urban planning and other policy activities. Government sources increasingly provide data that can be used by any party to analyze policy issues. For example, the U.S. government now provides an extensive array of data through a Web portal aimed facilitating transparency in government affairs (Federal CIO Council 2009). This portal contains a GIS viewer that allows federal data to be rendered in a GIS format and integrated with other data sources.

3D and Interactive Applications

Performance increases and growth in 3D referenced data sets are facilitating the creation of 3D visual models for GIS (Smith and Friedman 2004). Integrative approaches allow the use of 3D GIS for computer-aided design, architectural design, and city planning. For example, 3D models are used to simulate alternative growth strategies for small towns by using a GIS platform and building 3D and interactive applications (Orton 1999). Simulated buildings and their surrounding material and natural environments can be visualized from many perspectives, engage the

public in participation, and lead to design changes and improvements (Longley et al. 2010). The advent of Google Maps, Google Earth, and Microsoft Virtual Earth in 2005 made 3D GIS capabilities available for mass consumption. Although largely not realized as yet, 3D optimization models combined with GIS offer potential for more precise spatial decision-making.

Interactivity is further enhanced through the World Wide Web. The Web's graphic interface is particularly suited to the visual nature of GIS. The ability to access data from remote locations enables interactive databases to be created that can be queried along many number of dimensions. For example, traffic safety conditions can be analyzed by local communities using an interactive Web site, SafeRoadMaps.org, that allows the users to customize the scale and type of analysis based on local community interests and conditions (Hilton et al. 2009).

Examples of OR Analyses Using GIS

The following examples describe OR studies that deal with communications networks, forest management, and personnel assignments. Two of these studies were carried out in an earlier software environment that had fewer capabilities than are above. Nonetheless, the fundamental ideas are still in use and each new group of analysts needs to read the sources and absorb the approaches into their applications.

Modeling Communications Networks: An example of the use of a GIS and visual interactive modeling is given by Anghern and Lüthi (1990). They built a GIS system, Tolomeo, in a Macintosh environment that incorporates elements of modeling-by-example, an expert systems technique. They applied it to the problem of selecting the number and location of switching centers and routes in a communications network. The model displays the geographical area under consideration, the location of existing transmitting and receiving stations, proposed switching centers (nodes), and the transmission channels (arcs). By making the model object oriented, it was possible to attach to each node and arc data defining such quantities as traffic, cost, and transmission times. Users can redefine the network model interactively on the screen. Furthermore, they can create multiple views of the situation, using different visual metaphors. As the user modifies the model, the underlying data are recalculated to show the implications of proposed changes. Constraints and goals can be introduced so that the calculations take into account constraints and show where the current solution fails to meet goals. The modeling-by-example capability provides suggestions to the user on directions for improvement. These suggestions are based on applying optimization.

Forest Management: Two articles, Fletcher et al. (1999) and Epstein et al. (2006), deal with forest management projects in which a GIS was used in conjunction with other OR techniques. Fletcher et al. was based on work by The Pacific Lumber Co. and its GIS contractor, while Epstein et al. involved collaboration between the University of Chile and Oregon State University, working with a group of forestry companies.

In Fletcher et al., the objective was to develop a 120-year, 12-period forest-ecosystem management plan for the company's properties that met the then new state wildlife, fisheries, and timber resource requirements. The company also wanted to make sure that its harvesting would be optimal and the yields would be self-sustaining. The project was done under the auspices of a large lumber company in far Northern California that controlled extensive stands of redwood trees. The contractor built a model that seamlessly integrated a GIS with a database and a policy alternative model and that allowed adaptive management.

They started with using the GIS to divide the 200,000 acres held by the company into 406 strata types with each stratum containing one type of tree. Using GIS overlays of areas of special concern (e.g., stream barriers, wildlife corridors, owl buffers, watersheds), these strata were further subdivided into 7837 areas that were the fundamental decision units in the model. Each unit was defined by the tree type grown, amount of growth, and yield.

A linear-programming solution was first obtained and displayed on a map. Wildlife biologists then performed spatial integration of wildlife habitat types within the long-range planning model to determine reasonableness of the fragmentation, edge effects, and distribution of the watershed assessment areas.

The Epstein et al. article concerns the problem timber firms face in locating harvesting machinery

and transporting timber over a road network, while trying to reduce total cost and lower environmental impact. Optimization and GIS techniques were combined in a model which, for a forested area, locates harvesting machinery, and spatially optimizes the road network for transport of harvested trees to exit points. A mixed-integer programming model optimizes for the route that would incur the lowest total cost of road building and transportation. The model divides the forest into spatial grid coverage of 10x10 meter terrain cells that have associated topographical and production attributes including timber volume. Each cell is defined by 3D coordinates, thus allowing for estimation of slope angles in the predominantly mountainous timberlands. Slopes constrain the type of harvesting machinery (fixed versus vehicular) and the feasibility of locating roads. The user can re-set the locations of harvesting machinery and modify the maximal harvesting costs and allowable slope angles. A heuristic algorithm computes the cost of road building and timber transport over challenging terrain to exit points. Usual practice involves a forest area of 2,500 acres, with 75,000 spatial cells, 100,000 road vertices, 5,000 potential harvesting tractors and 300 fixed harvesting towers. The software has resulted in average operational cost reductions of 15–20 percent for large timber firms such as Forestal Bio Bio, Forestal Monteaguila (Shell Group), and Bosques Arauco S.A.in Chile.

Dispatching and Home Delivery: The retailing giant, Sears, uses a combination of OR models and a GIS to improve routing and scheduling for its truck delivery services:

- logistic delivery system for delivering newly purchased furniture and appliances, and
- product services system for installing and repairing appliances and providing home improvements and services.

Both systems involve sending trucks to customer homes, the first with over 1000 vehicles making 4 million deliveries annually, and the second with over 12,500 vehicles responding to 15 million service calls annually. Although they are run separately, the two systems are remarkably similar from both OR and GIS perspectives.

The original Sears system dates back to the 1990s (Weigel and Cao 1999). The objectives were (and are) (1) to deliver articles or arrive for service within a customer's time window, (2) minimize operational costs, and (3) provide consistent routes for drivers. In an updated version, they were still in operational use in 2010 (Longley et al. 2010).

The systems operate nationally, distributed among regional offices. Their big task is to make sure that items are delivered and calls are answered within a time window agreed upon with the customer. At a simple level, the systems solve the conventional analytic vehicle-routing problem with time windows (VRPTW) which was solved in the 1980s and l990s. At least they would be if the number of destinations were small and fixed, such as a set of warehouses or retailers. However, since the destinations are individual homes and vary from day to day, the standard OR solution is not adequate. The problems are much larger and must be solved in reasonable time every day. To make the systems efficient, they require mating VRPTW and GIS.

The system uses algorithms to build an origin–destination matrix and to improve sequencing and routing among other factors. Not only must the system take into account the customer's time window, it must also deal with labor constraints such as the available personnel, their skills and time for specific jobs, the down time for lunch and other breaks, and their schedule. In addition, the system must take routing into account, using the GIS since driving time varies, for example, by traffic as a function of time of day, the flatness or hilliness of the route, and much more. The algorithms also include intra-route and inter-route improvement routines, based on Tabu Search.

The system works as follows: When customer data are downloaded, the geo-code module locates the customer's x-y coordinates. Based on the street data stored by the GIS, it calculates the distance between all customers, providing the information for the assignment and route improvement modules. Using distances, driving times, time windows, and personnel specialties and other constraints, the computer generates routes. Since special circumstances rise frequently (e.g., customer changes in time windows, service time) and some constraints may be violated, the routings can be over-ridden manually.

Sears has reaped benefits from this system in on-time performance, much reduced daily route-computation time, reduced miles between stops, more customers handled per truck per day, reduced overtime, and reduced drive time.

Concluding Remarks

Geographic Information Systems provide the base for powerful OR analyses that integrate the visual capabilities of the computer with available spatial information and optimization techniques. The GIS capabilities offer the opportunity to change, fundamentally, the way 2D problems, such as location-allocation, and 3D problems are approached. The rise of Web-services and related cloud architectures has provided a means to take GIS from a highly customized application to one that is becoming a ubiquitous analysis tool for business, government, and individuals.

See

- ▶ Decision Support Systems (DSS)
- ▶ Heuristics
- ▶ Information Systems and Database Design in OR/MS
- ▶ Location Analysis
- ▶ Logistics and Supply Chain Management
- ▶ Vehicle Routing

References

Anghern, A. A., & Lüthi, H.-J. (1990). Intelligent decision support systems: A visual interactive approach. *Interfaces, 20*(6), 17–28.

Anselin, L. (1992). Spatial analysis with GIS: An introduction to application in the social sciences. *National Center for Geographic Information and Analysis Technical Paper*, 92–10.

Antenucci, J. C., Brown, K., Croswell, P. L., Kevany, M. J., & Archer, H. (1991). *Geographic information systems: A guide to the technology*. New York: Van Nostrand Reinhold.

Baker, J., Jones, D., & Burkman, J. (2009). Using visual representations of data to enhance sensemaking in data exploration tasks. Journal of the Association for Information Systems, *10*(7), Article 2.

Batty, M. (1989). Urban modelling and planning. In B. Macmillan. (Ed.). *Remodelling geography* (pp. 147–169). Oxford, UK.

Brewer, G. D. (1973). *Politicians, bureaucrats and the consultant: A critique of urban problem solving*. New York: Basic Books.

Clarke, K. (2011). *Getting started with GIS* (5th ed.). Upper Saddle River, NJ: Prentice Hall.

Dacey, M., & Marble, D. (1969). *Some comments on certain aspects of geographic information systems: Technical report no. 2*. Department of Geography, Northwestern University, Evanston, IL.

Dennis, L., & Gantz, D. (1999). An exploratory data analysis of the relationship between domestic violence incidents and socioeconomic factors. *NESUG '99 NorthEast SAS Users Group 12th Annual Conference Proceedings*, Washington, DC.

Epstein, R., Weintraub, A., Sapuner, P., Nieto, E., Sessions, J., Sessions, J., Bustamante, F., & Musante, H. (2006). A combinatorial heuristic approach for solving real-size machinery location and road design problems in forestry planning. *Operations Research, 54*(6), 1017–1027.

Federal CIO Council, *Data.gov Concept of Operations, US Office of Management and Budget*, The White House, Washington DC, December 9, 2009.

Fletcher, R. L., Alden, H., Holmen, S. P., Angelides, D. P., & Etzenhouser, M. J. (1999). Long-term forest system planning at pacific lumber. *Interfaces, 29*(1), 90–101.

Forrest, E. (Ed.). (1990). *Intelligent infrastructure workbook: A management-level primer of GIS*. Fountain Hills, AZ: AE-C Automation Level Newsletter.

Fotheringham, A. S. (1990). Some random(ish) thoughts on spatial decision support systems, National Center for Geographic Information and Analysis Technical Paper 90–5.

French, S. P., & Wiggins, L. L. (1989). Computer adoption and use in California planning agencies: Implications for education. *Journal of Planning Education and Research, 8*(2), 97–108.

French, S. P., & Wiggins, L. L. (1990). California planning agency experiences with automated mapping and geographic information systems. *Environment and Planning B, 17*(4), 441–450.

Geoffrion, A. M. (1983). Can OR/MS evolve fast enough? *Interfaces, 13*(1), 10–25.

Greene, R. P., & Pick, J. B. (2006). *Exploring the urban community: A GIS approach*. Upper Saddle River, NJ: Prentice Hall.

Hanigan, F. L. (1989). GIS recognized as valuable tool for decision makers. *The GIS Forum 1*, *4*.

Harris, B., & Batty, M. (1992). Locational models, geographic information and planning support systems. *National Center for Geographic Information and Analysis Technical Paper*, 92–1.

Hilton, B., Horan, T., & Schooley, B. (2009). Making traffic safety personal: Visualization and customization of national traffic fatalities. In M. Huang, Q. Nguyen, & K. Zhang (Eds.), *Visual information communication* (pp. 265–282). New York: Springer.

Lee, D. B. (1973). Requiem for large-scale models. *American Institute of Planners, 39*, 163–178.

Longley, P. A., Goodchild, M. F., Maguire, D. J., & Rhind, D. W. (2010). *Geographic information systems and science* (3rd ed.). New York: John Wiley and Sons.

Maguire, D., Batty, M., & Goodchild, M. (2005). *GIS, spatial analysis and modeling*. Redlands, CA: ESRI Press.

Nyerges, T. L., & Jankowski, P. (2010). *Regional and urban GIS: A decision support approach*. New York: The Guilford Press.

Orton. (1999). *Orton family foundation community planning and simulation project*, Rutland, VT.

Smith, G., & Friedman, J. (2004). *3D GIS: A technology whose time has come*. *Earth Observation Magazine*, November.

USGS. (1999). *The future of GIS*. Reston, VA: U.S. Geological Survey.
Weigel, D., & Cao, B. (1999). Applying GIS and OR techniques to solve sears technician-dispatching and home-delivery problems. *Interfaces, 29*(1), 112–130.

Geometric Programming

Joseph G. Ecker
Rensselaer Polytechnic Institute, Troy, NY, USA

Introduction

Early work in geometric programming was stimulated by Zener (1961, 1962) in his investigation of cost minimization techniques for engineering design problems. Subsequent work by Duffin (1962), Duffin and Peterson (1966), and Duffin, Peterson, and Zener (1967) provided the fundamental groundwork of the subject. Geometric programming refers to a class of optimization problems that have the form

$$\begin{array}{ll}(\boldsymbol{P})\text{minimize} & g_0(\boldsymbol{t}) \\ \text{subject to} & g_k(\boldsymbol{t}) \le 1 \text{ and } \boldsymbol{t}>0\end{array}$$

where $\boldsymbol{t} = (\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_m)$ is a vector of variables and, for $k = 0,1,\ldots, p$, the functions $g_k(\boldsymbol{t})$ are sums of terms having the form

$$u_i(\boldsymbol{t}) = c_i \boldsymbol{t}_1^{a_{i1}} \boldsymbol{t}_2^{a_{i2}} \cdots \boldsymbol{t}_m^{a_{im}}$$

where the coefficients $\{c_i\}$ and the exponents $\{a_{ij}\}$ are arbitrary real numbers. The following is an example of a possible geometric program with three variables:

$$\begin{array}{ll}\text{minimize} & g_0(\boldsymbol{t}) = \frac{40}{\boldsymbol{t}_1\boldsymbol{t}_2\boldsymbol{t}_3} + 40\boldsymbol{t}_2\boldsymbol{t}_3 \\ \text{subject to} & g_1(\boldsymbol{t}) = \frac{1}{2}\boldsymbol{t}_1\boldsymbol{t}_3 + \frac{1}{4}\boldsymbol{t}_1\boldsymbol{t}_2 \le 1 \\ \text{and} & \boldsymbol{t}_i > 0, \quad i = 1,2,3.\end{array}$$

The term geometric programming was adopted because of the role that the geometric–arithmetic mean inequality played in the initial development of a duality theory for problems having the above form. Initially, the class of problems was restricted by requiring that the coefficients be positive and the corresponding terms $u_i(\boldsymbol{t})$ were called posynomials. Thus, the term posynomial programming might well have been chosen instead of geometric programming. Many engineering design problems do have the form of a geometric program where the coefficients are positive. Several examples of such problems are given in Duffin, Peterson, and Zener (1967), in the paper on methods, computations, and applications of geometric programming by Ecker (1980), and in the references of the latter paper.

Geometric programs where some of the $\{c_i\}$ coefficients can be negative are called signomial programs and this class of optimization problems was first studied by Passy and Wilde (1967) and Blau and Wilde (1969). The initial theory of geometric programming has been generalized to a much broader class of optimization problems. The review article by Peterson (1976) shows how the approach to developing a duality theory through the use of inequalities can be generalized to a very broad class of problems.

Equivalence of Posynomial and Convex Programs

Posynomial programs can be reformulated so that the objective function g_0 and the constraint functions g_i are convex. The simple transformation

$$t_j = e^{zj} \quad \text{for } j = 1,2,\ldots,m$$

allows each posynomial term to be rewritten in the form

$$u_i(\boldsymbol{t}) = c_i e^{a_{i1}z_1 + a_{i2}z_2 + \cdots + a_{im}z_m}.$$

Let $\boldsymbol{A}$ be the matrix whose ith row $\boldsymbol{A}_i$ gives the exponents of the ith posynomial term, then $u_i(\boldsymbol{t})$ can be written as

$$u_i(\boldsymbol{t}) = c_i e^{\boldsymbol{A}_i z}$$

where $\boldsymbol{z}$ is the column vector with entries z_i. The matrix $\boldsymbol{A}$ is usually called the exponent matrix. Notice that $\boldsymbol{A}$ is $n \times m$ where m is the number of variables and n is the number of posynomial terms.

For our example three-variable problem, the exponent matrix $\boldsymbol{A}$ is given by

$$\boldsymbol{A} = \begin{pmatrix} -1 & -1 & -1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}.$$

Defining $\boldsymbol{x} = \boldsymbol{Az}$, then each geometric program with positive coefficients can be written so that the objective function and all of the constraints are convex functions of the variables $\boldsymbol{x}$ because then each posynomial term can be written as

$$u_i(\boldsymbol{t}) = c_i e^{x_i}$$

where the linear constraints $\boldsymbol{x} = \boldsymbol{Az}$ are added.

The Dual of a Posynomial Program

Through the use of the geometric–arithmetic mean inequality a maximization problem can be generated from the posynomial program (**P**) above. The maximization problem has a dual variable d_i for each posynomial term u_i so that dual vector is given by

$$\boldsymbol{d} = (\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_n)^T.$$

Let

$$L_k = \text{the sum of the variables } d_i$$

corresponding to the kth function $g_k(\boldsymbol{t})$.

The dual program has the form

$$\begin{aligned} \max \quad & v(\boldsymbol{d}) = \tfrac{c_1}{\boldsymbol{d}_1}\tfrac{c_2}{\boldsymbol{d}_2}\cdots\tfrac{c_n}{\boldsymbol{d}_n} L_1^{L_1} L_2^{L_2} \cdots L_p^{L_p} \\ \text{subject to :} \quad & L_0 = 1 \\ & A^T\boldsymbol{d} = 0 \text{ and } \quad \boldsymbol{d} \geq 0. \end{aligned}$$

The three variable example above has the following dual program:

$$\begin{aligned} \max \quad & v(\boldsymbol{d}) = \left(\tfrac{40}{\boldsymbol{d}_1}\right)\left(\tfrac{40}{\boldsymbol{d}_2}\right)\left(\tfrac{1}{2\boldsymbol{d}_3}\right)\left(\tfrac{1}{4\boldsymbol{d}_4}\right)(\boldsymbol{d}_3 + \boldsymbol{d}_4)^{\boldsymbol{d}_3+\boldsymbol{d}_4} \\ \text{subject to :} \quad & \boldsymbol{A}^T\boldsymbol{d} = 0 \\ & \boldsymbol{d}_1 + \boldsymbol{d}_2 = 1 \\ & \boldsymbol{d} \geq 0. \end{aligned}$$

The duality theory showing how to use a solution to the dual program to obtain a solution to the original primal program (**P**) is developed in Duffin, Peterson, and Zener (1967). The problem (**P**) is called canonical if there is a dual vector $\boldsymbol{d}$ satisfying

$$\boldsymbol{d} > 0 \quad \text{with} \quad \boldsymbol{A}^T\boldsymbol{d} = 0.$$

Canonical problems always have a minimizing point $\boldsymbol{t}^*$ and if, the set of all points satisfying the constraints in (**P**) has a non-empty interior, then the following duality results hold:

1. The dual problem has a maximizing vector $\boldsymbol{d}^*$;
2. The maximum value of the dual is equal to the minimum value for the primal program (**P**);
3. Each minimizing point $\boldsymbol{t}$ for (**P**) satisfies $u_i(\boldsymbol{t}) = d_i^* v(\boldsymbol{d}^*)$ for each i corresponding to the terms $u_i(\boldsymbol{t})$ in the objective function, and $u_i(\boldsymbol{t}) = d^*{}_i / L_k(\boldsymbol{d}^*)$ for all i when $L_k(\boldsymbol{d}^*) > 0$.

The right-hand side of each equation in (iii) is a positive constant and, given a solution $\boldsymbol{d}^*$, one can take common logarithms of both sides of the equations to obtain a linear system in the variables $\log(t_i)$. Typically, this linear system has more equations than variables so it uniquely determines a minimizing vector $\boldsymbol{t}^*$.

Computational Methods

The first published algorithm for solving posynomial programs was a method by Frank (1966) that solves the dual problem and then uses the above duality relations to obtain a minimizing point for (**P**). Blau and Wilde (1971) and Rijckaert and Martens (1976) developed similar methods that solve the Karush-Kuhn-Tucker optimality conditions for the dual problem. Other dual methods have been investigated, as for example in Dinkel, Kochenberger, and McCarl (1974) and in Beck and Ecker (1975).

A class of computational methods that solve (**P**) directly are based on the idea of linearizing geometric that was initially proposed by Duffin (1970). Avriel and Williams (1970) and Avriel, Dembo, and Passy (1975) use the idea of condensing each function into a single posynomial term to formulate a linear program that can be used to obtain an approximate solution to (**P**) even if some of the coefficients are negative. For more details on these types of approaches were given in Dembo (1978).

See

- ▶ Convex Optimization
- ▶ Nonlinear Programming

References

Avriel, M., & Williams, A. C. (1970). Complementary geometric programming. *SIAM Journal on Applied Mathematics, 19*, 125–141.

Avriel, M., Dembo, R., & Passy, U. (1975). Solution of generalized geometric programs. *International Journal for Numerical Methods in Engineering, 9*, 149–169.

Beck, P. A., & Ecker, J. G. (1975). A modified concave simplex algorithm for geometric programming. *Journal of Optimization Theory and Applications, 15*, 189–202.

Blau, G. E., & Wilde, D. J. (1969). Generalized polynomial programming. *The Canadian Journal of Chemical Engineering, 47*, 317–326.

Blau, G. E., & Wilde, D. J. (1971). A lagrangian algorithm for equality constrained generalized polynomial optimization. *AIChE Journal, 17*, 235–240.

Dembo, R. S. (1978). Current state of the art of algorithms and computer software for geometric programming. *Journal of Optimization Theory and Applications, 26*, 149–184.

Dinkel, J., Kochenberger, J., & McCarl, B. (1974). An approach to the numerical solution of geometric programming. *Mathematical Programming, 7*, 181–190.

Duffin, R. J. (1962). Cost minimization problems treated by geometric means. *Operations Research, 10*, 668–675.

Duffin, R. J. (1970). Linearizing geometric programs. *SIAM Review, 12*, 211–227.

Duffin, R. J., & Peterson, E. L. (1966). Duality theory for geometric programming. *SIAM Journal on Applied Mathematics, 14*, 1307–1349.

Duffin, R. J., Peterson, E. L., & Zener, C. M. (1967). *Geometric programming*. New York: John Wiley.

Ecker, J. G. (1980). Geometric programming: Methods, computations, and applications. *SIAM Review, 22*, 338–362.

Frank, C. J. (1966). An algorithm for geometric programming. In A. Lavi & T. Vogl (Eds.), *Recent advances in optimization techniques* (pp. 145–162). New York: John Wiley.

Passy, U., & Wilde, D. J. (1967). Generalized polynomial optimizations. *SIAM Journal on Applied Mathematics, 15*, 1344–1356.

Peterson, E. L. (1976). Geometric programming — a survey. *SIAM Review, 18*, 1–51.

Rijckaert, M. J., & Martens, X. M. (1976). A condensation method for generalized geometric programming. *Mathematical Programming, 11*, 89–93.

Zener, C. (1961). A mathematical aid in optimizing engineering design. *Proceedings of the National Academy of Sciences USA, 47*, 537–539.

Zener, C. (1962). A further mathematical aid in optimizing engineering design. *Proceedings of the National Academy of Sciences USA*, *48*, 518–522.

GERT

Graphical Evaluation and Review Technique; model of a network where all the nodes are of the exclusive-or type on their receiving side.

See

- ▶ Network Planning
- ▶ Project Management
- ▶ Research and Development

GIS

- ▶ Geographic Information Systems

Gittins Index

For the multi-armed bandit model under certain assumptions, it can be shown that an index policy specifying the choice of the arm with the highest index is optimal. The earliest result establishing this seminal structural result was Gittins (1979), so the resulting index is most commonly attributed to him.

See

- ▶ Multi-armed Bandit Problem

References

Gittins, J. C. (1979). Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society: Series B (Methodological), 41*(2), 148–177.

Global Balance Equations

A system of steady-state equations for a Markov chain (typically a queueing problem) obtained by balancing the mean flow rates or probability flux in and out of each individual state, symbolically written as $\pi \boldsymbol{Q} = \boldsymbol{0}$.

See

- ▶ Networks of Queues
- ▶ Queueing Theory

Global Climate Change Models

Hans W. Gottinger
International Institute for Technology Management and Economics, Bad Waldsee, Germany

Introduction

For many years, operations research has impacted the conceptual foundations, the scale and scope of models providing normative and predictive explanations on a potentially serious environmental threat known as the greenhouse effect. To assess long-term socio-economic changes due to the prevalence of the greenhouse effect, integrated models of energy, economy and the environment (EEE) of various levels of complexity have been constructed and put to use (Dowlabati 1995).

Structure of Energy-Economy-Environmental (EEE) Models

Early modeling approaches, in the context of carbon dioxide (CO_2) policies, involved an activity analysis model adapted to the CO_2 problem (Nordhaus 1979, 1980). This is an instructive example of the application of simple optimization models to a quantitative, qualitative and integrated analysis of CO_2 strategies. The analysis contains the major ingredients of EEE models of this kind: (1) the dynamics of the CO_2 cycle, the sources of CO_2 and the diffusion of atmospheric CO_2 and the limits of CO_2 concentrations; (2) the CO_2 energy model, for example a multi-sector activity analysis model which involves step-wise linear programming type optimization over a set of equidistant periods; and (3) the development of control strategies based on shadow prices of CO_2 emissions and costs of abatement.

A new generation of global EEE models (Peck and Teisberg 1993; Nordhaus 1993, 1994) contains special features, for example, the explicit consideration of the dynamics of economy and climate and a real interactive link-up between economic and climate dynamics models, relating to a time path of global mean temperature.

A culmination of efforts in the category of global EEE models has been the work of Nordhaus (1993, 1994). His approach centers around the construction, integration, model assessment and policy analysis of his economic control model DICE (Dynamic Integrated Model of Climate and Economy). DICE is a dynamic, intertemporal, optimal, interactive, welfare-economic control model based on structural equation constraints such as population growth, production constraints, capital stock accumulation, and emission constraints [where greenhouse gases (GHGs) are normalized by their carbon dioxide equivalent in terms of their global warming potential (GWP)]. The model contains a critical economy climate interface that links GHG emissions to their accumulation and transport in the atmosphere, the radiative forcing of the GHGs and their links to climate change. To assess the economic impacts such as damages, DICE contains feedbacks from climate change to economics, by specifying the loss of global output due to climate change. The climate part of DICE relates to specifications of General Circulation Models (GCMs), condensed as a minimodel of climate change to have it fit with the economic interface. Given the structure of DICE, Nordhaus puts his model to test. He first estimates damage profiles of GHG induced damages, for particular sectors as well as enticing the entire GDP loss. Furthermore, he looks at the welfare economic implications (net benefits) of seven major policy strategies to control global climatic changes: (1) no controls; (2) optimal policy; (3) ten-year delay of optimal policy; (4) stabilizing emissions at 1990 rates; (5) 20% emission reduction from 1990 levels; (6) geoengineering; and (7) climate stabilization with upper limit of total mean temperature increase by 1.5 °C from 1990. The net benefits vary significantly in size from each other, where it is remarkable that, in general, more interventionist strategies (stabilization), as strongly advocated by environmentalists, fare much worse than less interventionist ones (except for geoengineering which, of course, is hardly to the environmentalist's delight).

The extent of uncertainty in the model parameters gives rise to estimating the impact range on strategic

outcomes, as well as it applies to regulatory decision making on how to optimally impose regulatory controls to minimize over or undershooting of environmental regulation and policy measures (the value of information of waiting vs. acting).

Another string of models and research results of resource economics (on the depletion of non-renewable resources) could be applied with simple modifications to the CO_2 problem. Under two crucial assumptions, the problem of fossil fuel use in the face of increasing carbon dioxide is parallel to the problem of consumption of a limited resource. The first assumption is that the carbon dioxide absorption rate is sufficiently small to be ignored. The second is that CO_2 impacts follow a step pattern, that is, CO_2 (as a pollution stock) has no impact on productivity until a critical level, M_c, is reached; then if the CO_2 level exceeds M_c, production drops sharply (or more extremely, falls to zero).

A model with endogenous neutral technical progress, as in Gottinger (1998a), has been proposed to provide a better explanation of technical changes used to date in EEE models. Such a model originates from a similar attempt by Chiarella (1980). He proved the existence of a steady state growth path and a simple rule governing the rate of investment in research. Research investment along the optimal path should be carried out until the growth rate in the marginal accumulation of technology equals the difference between the marginal product due to an extra unit of research investment and the marginal product of capital.

Another issue is uncertainty. Here again there is a link with models of resource use for a limited, non-renewable resource when the reserve of the resource is unknown. The key finding of models by Loury (1978) and Gilbert (1979) was that plans for resource use based on the expected level of a resource will be overly optimistic. Gilbert's model is conducive to models of fossil fuel use when the critical CO_2 level is uncertain. Under the above assumptions, this problem is equivalent to determining the rate of fossil fuel use when the critical concentration of atmospheric carbon dioxide is unknown. Their results show that the optimal use of fossil fuel is lower when uncertainty is properly considered than when the expected values are assumed to be certainty equivalents.

A significant additional element is the possibility of undertaking exploration to find new reserves. The parallel in the CO_2 problem is R&D to increase the probability of finding a technology for the removal of CO_2 from the atmosphere.

Issues of Uncertainty

Existing EEE models suffer from poor data, indeterminate structure, and a frequent lack of attention to the consequences of uncertainty. The factors linking energy activities to their environmental effects are known only imprecisely. EEE models rely on behavioral assumptions that are widely questioned, and on parameters that can vary substantially from one model or data source to the next.

Most EEE models, including those well-established and highly used, conceal this uncertainty behind a blanket or output detail: a profusion of fuel prices and quantities, sectoral disaggregation, regional detail, growth rates and target figures, which often steer the analysis toward a desired conclusion. Unfortunately, this complexity rarely contributes to a resolution of uncertainty, and may serve only to increase the error and expense. Concerning models of possible greenhouse effects and CO_2 emissions, uncertainty analysis assumes many facets. It involves: (1) changes in climate to be expected; (2) impact of climate change; and (3) costs of adapting to climate change.

It is appropriate to distinguish between uncertainty about occurrences of events and impacts. Policy uncertainty is also of great concern. For the CO_2 problem, some argue that it is premature to think about doing other than intensive research, others claim that the risks of waiting are simply too great. What is the value of reducing scientific uncertainty? Scenario analysis only provides an indirect treatment of uncertainty, all uncertainties are resolved prior to decision-making. But uncertainty, information and decision-making are intimately connected and a comprehensive approach based on Bayesian decision analysis shows promise (Manne and Richels 1990).

Based on the described structure of EEE models, entire families of models have emerged in Europe,

the United States, and through international organizations (OECD, EU, World Bank, etc.), they range from highly aggregated general equilibrium models to multi-sectoral econometric models (Gottinger 1998a, b).

An interesting problem emerges in dealing with outcome uncertainties in climate change, that is with the timing of regulation. Such a design relates to problems of optimal stopping (Conrad 1992). Choosing the level of GHG emission-limiting regulations that will maximize social welfare by optimally balancing the costs of emission control against the benefits of decreased environmental damage is inherently not possible, because of pervasive uncertainty about the likely size of the critical GHG budget, its relationship to the quantity of GHG emitted, the effects of GHG in the atmosphere, and the appropriate valuation of these consequences. Moreover, learning more about each of these areas of uncertainty can be expected through continuing scientific-technological research, and through observation of atmospheric responses to past and current GHG emissions. Because, overall, it is expected that these uncertainties will diminish over time, the appropriate policy is likely to be an incremental and dynamic one. The risk to delaying before further restricting GHG emissions is that, if significant emission reductions become necessary to prevent serious adverse consequences, their cost may be much larger than if emission reductions begin sooner.

On the other hand, the risk to adopting further restrictions now is that these restrictions may later prove to have been unnecessary; the costs incurred would have produced no benefits. The question is analogous to that of whether to purchase insurance, as formulated by Manne and Richels (1992); by imposing additional regulations now, immediate costs are incurred in exchange for a potential reduction in the costs of preventing and adapting to future GHG accumulation. Gottinger (1995, 1996) attempted to provide insight to this question. First, he developed a general formulation of the policy question which can be conceived as an infinite-horizon, stochastic dynamic program with learning (Bertsekas 1976, Part II).

This formulation clarifies the issues, but is mathematically hard to manage. To provide more explicit guidance, advantage is taken of specific features of this problem to develop a simplified decision framework. Because of the long time delay in the relationships between GHG emissions, accumulation and effects in the atmosphere, the policy choice can be structured so that the environmental damages and benefits are approximately the same under each policy. Thus, the framework focuses attention on a comparison of the expected economic costs of alternative regulatory strategies.

In a different model framework, the aspect of learning has been given further attention by Kolstad (1993), with a survey of relevant approaches provided by Arrow et al. (1994).

Philosophy of EEE Modeling

Because the feedback effects of CO_2 are extremely uncertain, many modelers are reluctant to incorporate these effects in their models. In some scenarios, feedback effects might indeed be unimportant. For example, in models with finite horizons, if CO_2 effects are insignificant until after the horizon of the model no modeling of feedback effects is needed. In models which optimize over an infinite horizon, future effects may change current policies, and feedback effects are always of importance. In these same models, however, feedback effects may make solution much more difficult.

In predictive models with long time horizons, feedback effects will also be important. Further, in predictive models that estimate production and energy use at individual points of time, such as those surveyed previously, feedback effects can be easily included. Experience with the inclusion of feedback effects in optimizing models, shows that they usually lower the optimal initial use of fossil fuels. The longterm changes in fossil fuel use due to feedback effects are more uncertain and dependent on the model. In general, one could say that in most models the feedback slows the economy and thus reduces the demand for fossil fuels in the future. If optimization were included in the models discussed, this effect would be likely.

Given the uncertainty in the severity and timing of feedback effects, the sensitivity of individual models to variations in feedback effects is of much interest.

In proposing a step model of CO_2 emissions, the sensitivity of current fossil fuel use to an ultimate limit on atmospheric carbon dioxide was also examined. It was found that current optimal fossil fuel use was significantly affected by different critical levels of CO_2. A study of the impacts of a critical CO_2 level in a more disaggregated optimizing model would be useful. Including optimization in models expands their applicability but may cause analytic problems and controversy. As with many social problems, an acceptable objective function for carbon dioxide control problems is difficult to define. Any definition will seem both inadequate and overly precise and certainly would be controversial. This may be the reason why the models reviewed did not examine optimal policies. On the other hand, statement of an objective function does not hide or confuse other results and can add many new insights. If feedback effects and an objective function are included in a model, a crude optimization can be performed simply by running the model under a variety of policies.

Including optimization raised several new issues for the models. For example, pollution impoverishes but technical progress enriches the future. The optimizing models show how the curvature of the utility function, determined by the consumption elasticity of utility in the models, tends to smooth or even out wealth over time. Without an objective function being stated, the importance of this redistribution effect in determining fossil fuel use policy cannot be examined. In predictive models, a subjective evaluation must be made of the significance and value of a policy. In an optimizing model, the costs and benefits of policies are automatically compared in an explicit manner.

A final benefit of an optimizing model is the identification of multifaceted responses which may be ignored when policy changes are specified exogenously. Integrated optimizing models respond to problems by adjusting numerous policies endogenously. For example, in multiple state models fossil fuel use, research, and capital all respond to changes in the effects of CO_2.

See

- ▶ Economics and Operations Research
- ▶ Electric Power Systems
- ▶ Engineering Applications
- ▶ Environmental Systems Analysis
- ▶ Global Models
- ▶ Large-Scale Systems
- ▶ Risk Assessment
- ▶ Systems Analysis

References

Arrow, K. J., Parikh, S., & Pillet, G. (1994). Decision making framework to address climate change. In G. Pillet, & F. Gassman (Eds.), *Report of the IPCC working group III, montreux meeting PSI-Bericht 94-10*, Paul Scherrer Institute, Wurenlingen, Switzerland.

Bertsekas, D. P. (1976). *Dynamic programming and stochastic control*. New York: Academic.

Chiarella, C. (1980). Optimal depletion of a nonrenewable resource when technological progress is endogenous. In M. C. Kemp & N. V. Long (Eds.), *Exhaustible resources, optimality and trade*. Amsterdam: North Holland.

Conrad, J. M. (1992). Stopping rules and the control of stock pollutants. *Seminar of uncertainty in management of natural resources and the environment*, Central Statistical Bureau, Oslo.

Dowlabati, H. (1995). Integrated assessment models of climate change. *Energy Policy, 23*, 289–296.

Gilbert, R. J. (1979). Optimal depletion of an uncertain stock. *Review of Economic Studies, 46*, 47–57.

Gottinger, H. W. (1995). Regulatory policies under uncertainty, value of information and greenhouse gas emissions. *Energy Policy, 23*, 51–56.

Gottinger, H. W. (1996). Choosing regulatory options when environmental costs are uncertain. *European Journal of Operational Research, 88*, 28–41.

Gottinger, H. W. (1998a). *Global environmental economics*. Boston: Kluwer.

Gottinger, H. W. (1998b). Greenhouse gas economics and computable general equilibrium. *Journal of Policy Modeling, 20*, xx–xx.

Kolstad, Ch. D. (1993). Looking vs. leaping: The timing of CO_2 control in the face of uncertainty and learning. In *Costs, impacts and possible benefits of CO_2 mitigation*, IIASA, Laxenburg, Austria.

Loury, R. C. (1978). The optimum exploration of an unknown reserve. *Review of Economic Studies, 45*, 621–636.

Manne, A. S., & Richels, R. G. (1990). CO_2 emission limits: An economic cost analysis for the USA. *Energy Journal, 11*, xx–xx.

Manne, A. S., & Richels, R. G. (1992). *Buying green-house insurance–the economic costs of CO_2 emission limits*. Cambridge, MA: MIT Press.

Nordhaus, W. D. (1979). *The efficient use of energy resources*. New Haven, CT: Cowles Foundation, Yale University Press.

Nordhaus, W. D. (1980). Thinking about carbon dioxide: Theoretical and empirical aspects of optimal control strategies. *Cowles foundation discussion paper*, No. 565, Yale University, New Haven, CT.

Nordhaus, W. D. (1991). The cost of slowing climate change: A survey. *Energy Journal, 12*, 37–65.

Nordhaus, W. D. (1993). Rolling the 'DICE': Optimal transition path for controlling greenhouse gases. *Resource and Energy Economics, 15*(1), 27–50.
Nordhaus, W. D. (1994). *Managing the global commons: The economics of climate change*. Cambridge, MA: MIT Press.
Peck, S. C., & Teisberg, T. J. (1993). Global warming uncertainties and the value of information: An analysis using CETA. *Resource and Energy Economics, 15*(1), 71–97.

Global Maximum (Minimum)

For an optimization problem, the largest (smallest) value that the objective function can achieve over the feasible region.

See

- ▶ Local Maximum
- ▶ Nonlinear Programming
- ▶ Quadratic Programming

Global Models

Saul I. Gass
University of Maryland, College Park, MD, USA

Global or world models are concerned with the application of systems analysis to policy problems of intra and international interest. Typical problems of concern include population growth, ecological issues (forestry, fisheries, pesticides, insect infestation), energy and water resource availability and uses, the spread of diseases, and environmental models (acid rain, air pollution), Clark and Cole (1975), Holcomb (1976). Global models are usually highly aggregated in their structure and in their data requirements. Such models, however, can be developed by integrating lower-level and more detailed national or regional models. Of related interest are global and regional predictive models that deal with long-range weather or macro-economic activity.

Although the trail of global models leads back to Malthus and his 1798 publication of *An Essay on the Principle of Population*, the modern development of global models begins with the use of systems analysis in the study of global problems, and the availability of specific tools for analysis such as (Forrester 1961) Leontief's Input–output Interindustry Structure, and Dantzig's Linear Programming Model. In particular, Forrester and his associates brought the use of global models to the attention of governmental officials and to the scientific community by their application of the World 2 and World 3 system dynamics models that are described in *World Dynamics* (Forrester 1971) the and *Limits to Growth* (Meadows et al. 1972), respectively.

The World 3 model considers the world as a whole and evaluates five global indicators and their interactions: population, consumption of nonrenewable resources, pollution, food production, and industrialization. The model's calculations lead to the conclusion that sometime in the twenty-first century, the world will witness a steep decline in food per capita and in population. The general pessimistic conclusion reached by the World 3 model (under varying assumptions such as availability of resources) was that the world will soon be hitting resource, economic, and population limits to growth, and measures must be initiated by the world community to avoid calamity. The model indicated a stable future only if such stringent measures as maintaining a stable (zero growth) world population and capital base, and such measures are applied soon (Meadows et al. 1972; Clark and Cole 1975). Criticisms of this conclusion abound and they address the issues of the model's structure, data, aggregation, and methodological approach. (See, for example, Cole et al. 1973; Schwartz and Foin 1972, and a rebuttal by Forrester 1976).

Other global models have been developed in an attempt to overcome some of the limitations and criticisms of the World 3 model; in particular, note the one by Mesarovic and Pestel (1974). This model divided the world into ten regions and enabled some policy options (e.g., energy resource utilization) to be evaluated. Research in global models continues, with one center for such investigations being the International Institute for Applied Systems Analysis (IIASA). IIASA has initiated a database collection for environmental analyses, developed an acid rain model for Europe, forest resource and pest management models, plus econometric and linear-programming-based approaches to global policy modeling (Bruckmann 1980).

The means of encompassing the complex interactions of the global system into a computer-based model will always be open to criticism. As any model is an approximation of the real-world, surely a model that attempts to encompass the whole world or even major subelements cannot do so with much exactitude. One would not expect it to be so. As noted by Mason (1976, p. 4): "We have seen that ultimately there is no objective way to assess world models." But, there is no reason why investigators, building on such past efforts as those described above and others, cannot develop global models that would be of value to the world's policymakers.

See

- Environmental Systems Analysis
- Global Climate Change Models
- Input–Output Analysis
- System Dynamics
- Systems Analysis

References

Bruckmann, G. (Ed.). (1980). *Input–output approaches in global modeling*. Oxford: Pergamon Press.

Churchman, C. W., & Mason, R. O. (Eds.). (1976). *World modeling: A dialogue*. New York: North-Holland.

Clark, J., & Cole, S. (1975). *Global simulation models: A comparative study*. New York: John Wiley.

Cole, H. S. D., Freeman, C., Tahoda, M., & Pavitt, K. L. R. (1973). *Models of doom*. New York: Universe Books.

Forrester, J. W. (1961). *Industrial dynamics*. Cambridge, MA: MIT Press.

Forrester, J. W. (1971). *World dynamics*. Cambridge, MA: MIT Press.

Forrester, J. W. (1976). Educational implications of responses to system dynamics models. In C. W. Churchman & R. O. Mason (Eds.), *World modeling: A dialogue* (pp. 27–35). New York: North-Holland.

Holcomb Research Institute. (1976). *Environmental modeling and decision making*. New York: Praeger.

Mason, R. O. (1976). The search for a world model. In C. W. Church-man & R. O. Mason (Eds.), *World modeling: A dialogue* (pp. 1–9). New York: North-Holland.

Meadows, D. H., Meadows, D. L., Randers, J., & Behrens, W. W., III. (1972). *The limits to growth*. Washington, DC: Signet Books.

Mesarovic, M., & Pestel, E. (1974). *Mankind at the turning point*. New York: E.P. Dutton, Reader's Digest Press.

Schwartz, S. I., & Foin, T. C. (1972). A critical review of the social systems models of jay forrester. *Human Ecology, 1*(2), 161–173.

Global Optimization

Hoang Tuy[1], Steffen Rebennack[2] and Panos M. Pardalos[3]

[1]Vietnam Academy of Science and Technology, Hanoi, Vietnam

[2]Colorado School of Mines, Golden, CO, USA

[3]University of Florida, Gainesville, FL, USA

Introduction

Consider an optimization problem of the general form

$$\min\{f(x)|g_i(x) \le 0, i = 1, \ldots, m,\ x \in X\} \quad \text{(P)}$$

where X is a closed convex set in $\mathbb{R}^n$, $f : \Omega \to \mathbb{R}$, and $g_i : \Omega \to \mathbb{R}, i = 1, \ldots, m$, are continuous functions defined on some open set Ω in $\mathbb{R}^n$ containing X. Setting

$$D = \{x \in X | g_i(x) \le 0, i = 1, \ldots, m\},$$

the problem can also be written as

$$\min\{f(x)|x \in D\}.$$

Any point $\bar{x} \in D$ is called a feasible solution of the problem. A feasible solution $\bar{x}$ is called a global optimal solution if it is the best of all feasible solutions, i.e., if it satisfies

$$f(\bar{x}) \le f(x) \quad \forall x \in D. \tag{1}$$

A feasible solution $\bar{x}$ is called a local optimal solution if it is the best among all feasible solutions in some neighborhood of it, i.e., if there exists a neighborhood W of $\bar{x}$ such that

$$f(\bar{x}) \le f(x) \quad \forall x \in D \cap W. \tag{2}$$

Many important practical problems may have many local optimal solutions with different objective function values. The need may then arise to find the best among them, i.e., a global optimal solution.

In convex (linear, resp.) programs where $X = \mathbb{R}^n, f(x), g_i(x)$ are all convex (linear, resp.),

any local optimal solution is global, and efficient algorithms are routinely used for solving these problems. Aside from these cases, finding a global optimal solution is a hard problem requiring quite different techniques. Due to its importance for many applications, global optimization is an active research field (Horst and Pardalos 1995; Floudas and Gounaris 2009; Pardalos and Coleman 2009).

Some typical examples of global optimization applications in operations research and management science (OR/MS) are presented to show the general mathematical structure of global optimization problems. Next, the general concept of branch and bound, the most popular general purpose method for solving global optimization problems, is described. Finally, different classes of global optimization problems encountered in the applications and requiring specialized treatment are reviewed.

Examples from OR/MS

In most mathematical programming problems of form (P) encountered in OR/MS, each function $f(x), g_i(x)$ represents a cost, a performance (return, benefit, ...), or a negative utility (loss, pollution, ...) that depends on the decision variable $x \in \mathbb{R}^n$. The decision maker may want to determine $x \in \mathbb{R}^n$ so as to minimize the associated cost $f(x)$ subject to constraints $g_i(x) \leq 0$ expressing, for example, the requirement that the total cost of i-th resource involved should not exceed certain acceptable limit, or the expected i-th utility should not be less than a certain required level (in addition to constraints $x \in X$ reflecting other "technical" aspects of the problem). A common phenomenon is that economy of scale (or decreasing return) prevails in certain sectors or within certain scale limits, while diseconomy of scale (or increasing return) prevails in other sectors or beyond certain scale limits. Mathematically, economy of scale or decreasing return is modeled by concave or decreasing functions, diseconomy of scale or increasing return is modeled by convex or increasing functions. In more complex situations increasing and decreasing return phenomena may be copresent, and more general types of functions have to be used that are *dc functions* (differences of convex functions), or *dm functions* (differences of monotonic increasing functions).

Recall that a function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be convex if $f(\alpha x + (1-\alpha)x') \leq \alpha f(x) + (1-\alpha) f(x')$ for any $x, x' \in \mathbb{R}^n$, and any real number α such that $0 \leq \alpha \leq 1$; it is said to be increasing if $f(x) \leq f(x')$ for any $x, x' \in \mathbb{R}^n$ such that $x_i \leq x'_i, i = 1, \ldots, n$. A function $f(x)$ is said to be concave if $-f(x)$ is convex; decreasing if $-f(x)$ is increasing.

Global optimization has been applied to many fields such as biomedicine, economics, energy systems, computational chemistry and biology, and computer science (Floudas and Pardalos 2003; Rebennack et al. 2010a; Rebennack et al. 2010b). Three examples in OR/MS are presented here.

Example 1. (Production-transportation planning) Consider k factories producing a certain good to satisfy the demands d_j, $j = 1, \ldots, m$, of m destination points. The production cost is $g(y_1, \ldots, y_k)$ if the factory i produces y_i units, where $g(\cdot)$ is a concave function because of economies of scale. The transportation cost is a function $c_{ij}(x)$ for x units shipped from factory i to destination point j. In addition, there is a shortage penalty $h(z_1, \ldots, z_m)$ to be paid if the destination point j receives $z_j \neq d_j$ units, where $h(z_1, \ldots, z_m) = \sum_{j=1}^{m} h_j(z_j)$, with $h_j(z_j) \leq 0$ if $z_j \geq d_j$, and $h_j(\cdot)$ is a decreasing nonnegative function in the interval $[0, d_j)$. Usually, the penalty function $h(\cdot)$ is convex, so the total production-transportation cost is the function $f(x, y, z) = \sum_{i=1}^{k} \sum_{j=1}^{m} c_{ij}(x_{ij}) + g(y) + h(z)$. This function should be minimized subject to usual transportation constraints: $\sum_{j=1}^{m} x_{ij} = y_i, i = 1, \ldots, k, \sum_{i=1}^{k} x_{ij} = z_j, j = 1, \ldots, m, x_{ij} \geq 0, y_i, z_j \geq 0 \, \forall i, j$.

Even if the transportation costs $c_{ij}(x)$ are linear, this problem cannot be handled successfully by conventional methods of nonlinear programming. Things become more complicated when the transportation costs $c_{ij}(x)$ along certain arcs (ij) are dc functions (for instance S-shaped functions), or some are concave, others are convex (Holmberg and Tuy 1993).

Example 2. (Location planning) A facility has to be constructed to serve n users located at points $a^j \in S$ of the plane (i.e., $S \subset \mathbb{R}^2$). If the facility is located at $x \in S$, then the attraction of the facility to user j is $q_j(h_j(x))$, where $h_j(x) = \| x - a^j \|$ is the distance

from x to a^j and $q_j : \mathbb{R} \to \mathbb{R}$ is a convex decreasing function (the farther x is away from a^j the less attractive it looks to user j). To determine the location of the facility with maximal total attraction, one has to solve the problem

$$\text{maximize} \sum_{j=1}^{n} q_j(h_j(x)) \quad \text{s.t. } x \in S. \tag{3}$$

The function $\varphi(x) = \sum_{j=1}^{n} q_j(h_j(x))$ is generally neither convex nor concave. However, since the functions $h_j : \mathbb{R}^2 \to \mathbb{R}_+$ and $q_j = \mathbb{R} \to \mathbb{R}$ are convex, it can be shown that their compositions $q_j(h_j(x))$, $j = 1, \ldots, n$, are dc, and $\varphi(x)$ a sum of dc functions, hence itself a dc function.

In practice, some of the points a^j may actually be repulsion rather than attraction points. For example, there may exist a garbage dump, a sewage plant, or a nuclear plant in the area, and one may wish the facility to be located as far away from these points as possible. If J_1 is the set of attraction points, J_2 the set of repulsion points, then instead of (3) one should seek to maximize the function

$$\sum_{j \in J_1} q_j(h_j(x)) - \sum_{j \in J_2} q_j(h_j(x)). \tag{4}$$

An even more complex situation occurs when several facilities must be located. In this case, each user will be served by the nearest facility, so the problem is to determine the locations, say x, y and z, of the facilities, so as to maximize

$$\sum_{j=1}^{n} q_j(\tilde{h}_j(x, y, z)) \tag{5}$$

over $(x, y, z) \in S \times S \times S$, where

$$\tilde{h}_j(x, y, z) = \min\{h_j(x), h_j(y), h_j(z)\}. \tag{6}$$

Again it can be proven that $\tilde{h}_j(x, y, z)$ (pointwise minimum of finitely many convex functions) is a dc function, and $q_j(\tilde{h}_j(x, y, z))$ (convex functions of dc functions) are also dc functions. Therefore, again this multifacility location problem appears to be a dc optimization problem (Chen et al. 1992). Note that this problem is sometimes formulated as a mixed integer program, much more difficult to solve.

Example 3. (Multilevel programming) Many decentralized decision-making systems in economics and other fields involve a "leader" (the higher level decision maker) who controls a variable $x \in \mathbb{R}^p$ and a "follower" (the lower level decision maker) who controls a variable $y \in \mathbb{R}^q$. For each decision x made by the leader, the follower chooses y in order to optimize his own objective function $\varphi(y)$, under a constraint set $\Omega(x)$ associated with the decision x, i.e., the response y of the follower to the decision x of the leader is a vector such that

$$y \in \operatorname{argmin}\{\varphi(y') | y' \in \Omega(x)\}.$$

If the objective of the leader is to minimize a function $f(x, y)$, while his own constraint set is $D \subset \mathbb{R}^p \times \mathbb{R}^q$ then the problem he must solve is to choose x so as to

minimize

$$f(x, y) \tag{7}$$

subject to

$$(x, y) \in D, y \in \Omega(x); \tag{8}$$

$$\varphi(y) \leq \varphi(y') \forall y' \in \Omega(x). \tag{9}$$

Even in the simplest cases when all data are linear: $f(x, y) = c_1 x + d_1 y, \varphi(y) = d_2 y, D = \{(x, y) | A_1 x + B_1 y \leq g_1, x \in R^p_+\}$ while $\Omega(x) = \{y | A_2 x + B_2 y \leq g_2, y \in R^q_+\}$, the feedback relation between upper and lower levels creates nonconvexities that cannot be easily handled by standard methods of nonlinear programming.

If $h(x)$ denotes the optimal value of the lower subproblem, i.e., $h(x) = min\{\varphi(y) | y \in \Omega(x)\}$, then the constraint (9) can also be written as

$$\varphi(y) \leq h(x).$$

Often $\varphi(y)$ is a convex function and, as in the just mentioned linear case, for any x^1, x^2 and any $\alpha \in [0,1]$, then $\alpha\Omega(x^1) + (1-\alpha)\Omega(x^2) \subset \Omega(\alpha x^1 + (1-\alpha)x^2)$. It can then easily be checked that $h(x)$ is a convex function, too, and so the constraint (3), i.e., $\varphi(y) - h(x) \leq 0$, is a dc constraint.

Bilevel, and more generally multilevel, programming problems of the above kind have various applications in economics (Stackelberg duopoly model), agriculture (e.g., fertilizer supply, water supply, agricultural policy), financial management, and network design (Wen and Hsu 1991). Thus a host of problems of practical interest in economics, OR/MS and engineering involve dc functions or dm functions in their description. Other problems reported from computer science (VLSI chip design, databases), wireless communications, system reliability, mechanics (structural optimization), physics (nuclear design, microcluster phenomena in thermodynamics), chemistry (phase and chemical reaction equilibrium), or ecology (design and cost allocation for waste treatment systems) can analogously be identified as dc or dm optimization problems (Floudas and Pardalos 1999; Floudas 2000).

A problem (P) where all functions $f(x), g_1(x), \ldots, g_m(x)$ are dc (dm, resp.) is called a dc (dm, resp.) optimization problem. These are two basic classes of global optimization problems. Practically, every global optimization can be reformulated, possibly via a change of variables, as a dc or a dm optimization problem.

Branch and Bound Methods

A popular method for solving a global optimization problem (P) is by using a BB (branch and bound) procedure.

The Generic BB Procedure

For convenience assume that $X = [a,b] = \{x \in \mathbb{R}^n | a_j \leq x_j \leq b_j, j = 1, \ldots, n\}$, so that the feasible set in problem (P) is

$$D = \{x \in [a,b] | g_i(x) \leq 0, i = 1, \ldots, m\}.$$

The generic BB procedure for solving (P) involves two basic operations: partitioning and bounding.

- **Partitioning**: Starting from the initial box (hyperrectangle) $M_1 = [a,b]$, at each iteration a box is selected and subdivided into two subboxes according to a subdivision rule. Through this partitioning process, a tree is generated with the root at M_1 and the nodes represented by the subboxes that appear as successive descendants of the initial box.
  Let $M = [p,q]$ be a box selected for subdivision in a given iteration. A common subdivision rule called the standard bisection consists in dividing M into two equal subboxes using a hyperplane perpendicular to a longest edge of M at the midpoint of this edge. An important property of this subdivision rule is its exhaustiveness, meaning that any infinite nested sequence of boxes M_k generated by it shrinks to a point (i.e., diam $M_k \to 0$ as $k \to +\infty$).
- **Bounding**: At each iteration, two new boxes appear as a result of the subdivision operation. For each new box $M = [p,q]$, a lower bound is computed for $f(x)$ over the feasible points in M, i.e., a number $\beta(M) \in \mathbb{R} \cup \{-\infty, +\infty\}$ satisfying

$$\beta(M) \leq inf\{f(x) | x \in M \cap D\}, \tag{10}$$

$$M_k \cap D = \emptyset \Rightarrow \beta(M) = +\infty. \tag{11}$$

The latter condition, which is essential, amounts to requiring that $\beta(M) < +\infty$ only if $M \cap D \neq \emptyset$.
The number $\beta(M)$ is usually computed by considering an underestimator $\varphi(x)$ of $f(x)$ over a set $\Omega \supset M$, i.e., a function satisfying $\varphi(x) \leq f(x) \forall x \in \Omega$, and taking $\beta(M) = inf\{\varphi(x) | x \in \Omega\}$, where $\varphi(x)$ and Ω are chosen so that the latter problem can be solved easily. Also to obtain tight bounds it is often necessary to replace the partition set M by a suitable smaller set $M' \subset M$: the procedure is then referred to as a branch-and-reduce algorithm.
While computing the lower bounds, it may happen that some feasible points are obtained: the feasible point with smallest objective function value is then recorded as the current best feasible solution (CBS) and the associated objective function value as the current best objective function value (CBV). Once every current box has been assigned a lower bound, all boxes M with

$\beta(M) > CBV$ (in particular those with $\beta(M) = +\infty$) are pruned (deleted). If no box remains after that, the procedure is terminated and CBS gives a (global) optimal solution. Otherwise, a box with smallest lower bound among all remaining boxes is selected for further subdivision, and a new iteration is started.

Proposition. *If bounding is consistent with branching in the sense that*

$$\beta(M) - \min\{f(x)|x \in M \cap D\} \to 0 \quad \text{as } diam\, M \to 0, \tag{12}$$

then as $k \to +\infty$, the box M_k with smallest lower bound at iteration k shrinks to a point which is an optimal solution, while $\beta(M_k)$ tends to the optimal value $min(P)$ of the problem.

Case of a Nice Feasible Set

Suppose the feasible set is nice, i.e., such that a feasible solution can be computed cheaply (as is the case, for example, when each $g_i(x)$ is convex or each $g_i(x)$ is increasing). Then at each iteration k, a current best feasible solution CBS is available that provides an upper bound for $\min(P)$. In that case $\beta(M_k) \le \min(P) \le$ CBS, so if x^k is CBS at iteration k, then as $k \to +\infty$ the sequence $\{x^k\}$ tends to a limit $\bar{x}$ yielding an optimal solution of the problem. Furthermore, this convergence can be sped up by using instead of the standard bisection an adaptive bisection rule ensuring convergence without condition (12) (Tuy 1998, 2000). Given a tolerance $\eta > 0$, by stopping the procedure when $f(x^k) - \beta(M_k) \le \eta$, an η-optimal solution of the problem, i.e., a feasible solution x^* satisfying $f(x^*) \le min(P) - \eta$, is obtained. Thus, everything works well if the feasible set is nice.

Case of a Hard Feasible Set

By contrast, if the feasible set is such that a feasible solution cannot be computed cheaply, then several difficulties may arise with the generic BB method. Namely, in this case it may not be easy to compute lower bounds satisfying conditions (11) and (12), while failing these conditions the algorithm may converge to an incorrect solution which is infeasible and quite far from the optimum. Another drawback is that since at every iteration no feasible solution is available, no partition set can be pruned aside from those M with $\beta(M) = +\infty$, causing an excessive growth of the size of the collection of partition sets to be stored. Also, the convergence accomplished with an exhaustive subdivision process is in general slow. As a result, in finitely many steps the BB procedure can at best give an (ε, η)-approximate optimal solution, i.e., an $\bar{x}$ satisfying $g_i(\bar{x}) \le \varepsilon, i = 1, \ldots, m$ and $f(\bar{x}) \le \min(P) + \eta$. Unfortunately, such an (ε, η)-optimal solution is not guaranteed to be feasible and close to the true optimum, and, moreover, it may change drastically upon a small change of the tolerances ε and η, causing numerical instability problems in practical implementation.

The way out of these difficulties is to reduce any problem (P) with a hard feasible set to a sequence of problems with a nice feasible set. This is possible by using the following result (Tuy 2010):

By simple manipulations, any dc optimization problem (i.e., any problem (P) *where $f(x)$ and all $g_i(x)$ are dc) can be reformulated as an equivalent dc optimization problem with a convex objective function. Likewise, any dm optimization problem (i.e., any problem* (P) *where $f(x)$ and all $g_i(x)$ are dm) can be reformulated as an equivalent dm optimization problem with an increasing objective function.*

So for studying a dc optimization problem (P), one can without loss of generality assume that $f(x)$ is convex. Setting $g(x) = \min_{i=1,\ldots,m} g_i(x)$, the problem (P) can then be written as

$$\min\{f(x)|g(x) \le 0, x \in [a, b]\}, \qquad (\mathrm{P}')$$

where $g(x)$ is still a dc function by a known property of dc functions (Tuy 1998). Now, given any number $\gamma \ge \min(\mathrm{P}')$, consider the problem

$$\min\{g(x)|f(x) \le \gamma, x \in [a, b]\}. \qquad (\mathrm{Q}_\gamma)$$

Since $f(x)$ is convex, this problem has a nice feasible set and can be solved by the above BB method. Clearly $\min(\mathrm{Q}_\gamma) \le 0$ (because $\gamma \ge \min(\mathrm{P}')$) and if the problem (P') is such that

$$\min(\mathrm{P}') = \inf\{f(x)|g(x) < 0, x \in [a, b]\}$$

(a condition satisfied in most cases), it can easily be shown that $\min(Q_\gamma) = 0$ only if $\min(P') = \gamma$.

Based on this relationship between (P') and (Q_γ), the following method (Tuy 2010) can be used to find an η-optimal solution of (P'), i.e., a feasible solution x^* of (P') such that $f(x^*) \le \min(P') - \eta$.

Suppose a feasible solution $\bar{x}$ of (P') is available. With $\gamma = f(\bar{x}) - \eta$, apply the above described BB procedure for solving (Q_γ). If at some iteration k of this BB procedure, the current best solution x^k satisfies $g(x^k) \le 0$, then x^k is a feasible solution of (P') such that $f(x^k) \le f(\bar{x}) - \eta$. Otherwise, $g(x^k) > 0\ \forall k$ and from (12) it follows that $\min(Q_\gamma) = \lim g(x^k) \ge 0$, hence $\min(P') = \gamma = f(\bar{x}) - \eta$, i.e., $\bar{x}$ is an η-optimal solution of (P').

Thus, given a feasible solution $\bar{x}$ of (P'), by solving (Q_γ) with $\gamma = f(\bar{x}) - \eta$ one can either identify $\bar{x}$ as an η-optimal solution of (P') or find a feasible solution x of (P') such that $f(x) \le f(\bar{x}) - \eta$. Since $\eta > 0$, by repeating this procedure finitely many times, one will eventually obtain an η-optimal solution of (P').

To obtain an initial feasible solution $\bar{x}$ of (P'), it suffices to take any number $\gamma > \min(P')$ and apply the BB procedure for (Q_γ).

That is the basic idea of successive incumbent transcending for solving any dc optimization problem (P) with a hard feasible set. An analogous method can be used for solving a dm optimization problem with a hard feasible set: first rewrite it in the form (P') with an increasing function $f(x)$ and a dm function $g(x)$; then find an η-optimal solution of (P') by successive incumbent transcending via solving problem (Q_γ) with adaptively adjusted γ.

Specific Problem Classes

Aside from general purpose methods for global optimization, more efficient specific methods are available to solve specific problems by exploiting their underlying mathematical structure (Pardalos and Romeijn 2002). Other cases with specialized algorithms not discussed below include concave minimization and optimization with differences of monotonic functions or Lipschitz functions (Horst and Pardalos 1995; Horst and Tuy 1996). Furthermore, there is a rich body of literature on tailored decomposition algorithms for global optimization problems (Rebennack et al. 2009).

Quadratic Optimization

A quadratic optimization problem is a problem (P) where $f(x), g_1(x), \ldots, g_m(x)$ are quadratic functions, i.e., $f(x) = \frac{1}{2}\langle x, Q^0 x\rangle + \langle c^0, x\rangle$, $g_i(x) = \frac{1}{2}\langle x, Q^i x\rangle + \langle c^i, x\rangle + d_i$ with $Q^i, i = 0, 1, \ldots, m$, being symmetric $n \times n$ matrices and $c^i \in \mathbb{R}^n$, $d_i \in \mathbb{R}$

To solve a quadratic optimization problem by BB, the basic question is how to compute lower bounds. Two most used bounding methods are reformulation–linearization (Sherali and Adams 1999) and Lagrangian relaxation (Tuy 1998; Floudas 2000).

- **Reformulation–linearization**: Setting $x_i x_j = y_{ij}$ for every (i, j) with $1 \le i \le j \le n$ and $y = \{y_{ij}\}$, every quadratic function $f(x)$ of $x \in \mathbb{R}^n$ can be expressed as an affine function of x, y, denoted by $[f(x)]_\ell$. For example, $[2x_1x_3 + 3x_1^2 - 5x_2x_3 + 8x_3]_\ell = 2y_{13} + 3y_{11} - 5y_{23} + 8x_3$. Moreover, it can be shown that the constraint $p \le x \le q$ is equivalent to the system of quadratic constraints $g_{ij}(x) = (x_i - p_i)(x_j - q_j) \le 0\ \forall i, j = 1, \ldots, n$. Then for $M = [p, q]$, the problem $\min\{f(x) | g_k(x) \le 0, k = 1 \ldots, m, p \le x \le q\}$ can be rewritten as

$$\min\{[f(x)]_\ell \,|\, [g_k(x)]_\ell \le 0, k = 1 \ldots, h, \; y_{ij} = x_i x_j, 1 \le i \le j \le n\}, \tag{13}$$

where the constraints $g_k(x) \le 0, k = m + 1, \ldots, h$, include all the just-mentioned constraints $g_{ij}(x) \le 0$. Clearly (13) is a linear program, with the additional nonconvex constraints $y_{ij} = x_1 x_j, i \le i \le j \le n$. Therefore, as a lower bound for $\min\{f(x) | g_k(x) \le 0, k = 1, \ldots, m, x \in M\}$, one can take

$$\beta(M) = inf\{[f(x)]_\ell \,|\, [g_k(x)]_\ell \le 0, k = 1 \ldots, h\}.$$

It can be shown that this bounding operation satisfies (11) and (12), so the generic BB algorithm using this bounding method is correct, although its convergence is generally slow, due to the large number of additional variables introduced. However, there are various ways to improve the method, for example by adding implied constraints to (13) (Sherali and Adams 1999) or by using the incumbent transcending approach discussed earlier.

- **Lagrangian Relaxation**: For a given problem (P), the function $L(x,u) = f(x) + \sum_{i=1}^{m} u_i g_i(x)$, $u \in \mathbb{R}^m_+$, is called the Lagrangian, and the problem

$$\sup_{u \in \mathbb{R}^m_+} \inf_{x \in X} L(x,u) \qquad \text{(LP)}$$

the Lagrangian relaxation of (P). It is easily seen that $\min(\text{LP}) \leq \min(\text{P})$, so $\min(\text{LP})$ is a lower bound of $\min(\text{P})$. The Lagrangian relaxation is called exact if $\min(\text{LP}) = \min(\text{P})$. This occurs effectively, for example, when $m = 1$ and $X = \mathbb{R}^n$, i.e., when the problem is

$$\min\{f(x) | g(x) \leq 0, x \in \mathbb{R}^n\}$$

where $f(x), g(x)$ are quadratic functions and there is an $x^* \in \mathbb{R}^n$ such that $g(x^*) < 0$. Indeed, it is known that in this special case, the Lagrangian relaxation is exact and equivalent to a SDP (semi-definite programming problem) that can be solved by efficient methods (Ben-Tal and Nemirovski 2001). So, in particular, the minimization of a nonconvex quadratic function over an ellipsoid is equivalent to a SDP, i.e., essentially a convex problem.

In the general case $m > 1, X = [p,q]$, by replacing the constraint $x \in [p,q]$ with an equivalent system of quadratic inequalities as mentioned above, it can always be assumed that $X = \mathbb{R}^n$. Although the corresponding Lagrangian relaxation is also an SDP, it seems difficult to use exclusively this bounding method to produce a convergent BB algorithm. However, it can be incorporated into a convergent primal-relaxed dual decomposition approach (Floudas 2000).

Also note that a quadratic function of the form $\sum_{0 \leq i \leq j \leq n} c_{ij} x_i x_j$ with $c_{ij} \geq 0$ is an increasing function on $\mathbb{R}^n_+$. Therefore, any quadratic function on $\mathbb{R}^n_+$ is a dm function, and thus a quadratic optimization problem (P) where $X = \mathbb{R}^n_+$ can be viewed alternatively as a dm problem and as such can be solved by the method described earlier.

Multiobjective Programming

A multiobjective program is a generalization of problem (P) where $F(x)$ is a vector of k objective functions

$$\min\{F(x) = [f_1(x), f_2(x), \ldots, f_k(x)]^T \mid g_i(x) \leq 0,\ i = 1, \ldots, m,\ x \in X\}$$

where X represents the set of feasible decisions and $f_i(x) : \Omega \to \mathbb{R}, i = 1 \ldots, k$ are the objective functions that the decision maker wants to minimize. A feasible decision x^0 is said to be efficient (Pareto-optimal) if for any $x \in X$, $f_i(x) \leq f_i(x^0) \forall i$ implies $f_i(x) = f_i(x^0) \forall i$; it is said to be weakly efficient if there is no $x \in X$ such that $f_i(x) < f_i(x^0) \forall i$. An efficient or weakly efficient solution achieves a kind of equilibrium and in certain situations the decision maker may want to find an equilibrium minimizing some objective function. For example, starting from a feasible solution, the decision maker may want to reach an equilibrium by the "cheapest" way (see Thach et al. 1996 for an example in bond portfolio optimization). If X_E denotes the set of efficient solutions, then the goal of the decision maker is to minimize a certain function $h(x)$ over X_E, i.e.,

$$\text{minimize } h(x) \text{ subject to } x \in X_E. \qquad (14)$$

In general, the set X_E is nonconvex, so even if $h(x)$ is linear, this is a difficult global optimization problem (Marler and Arora 2004). A relaxed variant of problem (14) is

$$\text{minimize } h(x) \text{ subject to } x \in X_{WE}, \qquad (15)$$

where X_{WE} denotes the set of weakly efficient solutions. It can be shown that when $f_i(x)$ are linear, (15) is equivalent to the problem

$$\min\{h(x) | x \in X, \lambda \in \Lambda, g(\lambda) - \lambda F(x) \leq 0\},$$

where Λ is a simplex in R^k and $g(\lambda) = \sup\{\lambda F(y) | y \in X\}$ is a convex function (so $g(\lambda) - \lambda F(x)$ is a dc function). This allows the use of the BB method discussed earlier.

Fractional Programming

Fractional programming deals with problems where a ratio of two objective functions has to be optimized. Several different forms of fractional programs can be distinguished (Frenk and Schaible 2004; Stancu-Minasian 1997).

- **Single-ratio fractional programs**: For extended real-valued continuous functions $f(x), h(x) : \Omega \to [-\infty, +\infty]$ with finite value on D, single-ratio fractional programs are given in the general form

$$\inf\left\{\frac{f(x)}{h(x)}\middle| g_i(x) \le 0,\ i = 1, \ldots, m,\ x \in X\right\}. \quad (16)$$

  For the special case where $f(x)$ and $g_i(x)$ are convex functions $\forall i$ and $h(x)$ is a positive concave function in D, (16) is called a single-ratio convex fractional program and is a nonconvex global optimization problem. Note that the ratio $\frac{f(x)}{h(x)}$ is not a convex function in general. Typical applications in OR/MS include the maximization of productivity, maximization of return on investments, maximization of return versus risk, minimization of cost versus time, and maximization of output versus input.
- **Generalized fractional program**: Extending (16) to multiple ratios leads to generalized fractional programs of the form

$$\inf_{x \in D} \sup_{1 \le l \le k} \frac{f_l(x)}{h_l(x)}, \quad (17)$$

  with $f_l(x), h_l(x)$: $\Omega \to [-\infty, +\infty]$ for all l and positive functions $h_l(x)$ for $x \in D$.
- **Sum-of-ratios fractional program**: Minimizing the sum of ratios leads to the following optimization problem

$$\inf\left\{\sum_{l=1}^{p} \frac{f_l(x)}{h_l(x)}\middle| g_i(x) \le 0,\ i = 1, \ldots, m,\ x \in X\right\}, \quad (18)$$

  with the same assumptions on the function $f_l(x)$ and $h_l(x)$ as in the generalized fractional programs. Bond portfolio optimization problems are examples of sum-of-ratios fractional programming problems.

Multiplicative Programming

One standard approach to simultaneously optimizing several objectives without a common scale is to optimize the product of these objectives. This leads to consider multiplicative programming problems of the form

$$\inf\left\{\prod_{l=1}^{p} f_l(x)\middle| g_i(x) \le 0,\ i = 1, \ldots, m,\ x \in \mathbb{R}^n\right\} \quad (19)$$

where $f_l : \mathbb{R}^n \to \mathbb{R}^+$, $g_i : \mathbb{R}^n \to \mathbb{R}$. Practical methods for solving these problems are available when each $f_l(x)$ is either quadratic or affine and each $g_i(x)$ is convex, for $p \le 5$ and $n, m \le 100$ (Konno et al. 1997). In particular, linear multiplicative programming problems, i.e., problems (19) where $p = 2$ and all functions f_l, g_i are affine, can be solved very fast by a variant of the parametric simplex algorithm. Applications of multiplicative programming include bond portfolio optimization and economic analyses.

See

- ▶ Branch and Bound
- ▶ Convex Optimization
- ▶ Fractional Programming
- ▶ Mathematical Programming
- ▶ Multiobjective Programming
- ▶ Nonlinear Programming
- ▶ Quadratic Programming

References

Ben-Tal, A., & Nemirovski, A. (2001). *Lectures on modern convex optimization*. Philadelphia: SIAM/MPS.

Chen, P.-C., Hansen, P., Jaumard, B., & Tuy, H. (1992). Weber's Problem with attraction and repulsion. *Journal of Regional Science, 32*, 467–486.

Floudas, C. A. (2000). *Deterministic global optimization*. Dordrecht/Boston/London: Kluwer Academic Publishers.

Floudas, C. A., & Gounaris, C. E. (2009). A review of recent advances in global optimization. *Journal of Global Optimization, 45*(1), 3–38.

Floudas, C. A., & Pardalos, P. M. (Eds.). (1999). *Handbook of test problems in local and global optimization*. Dordrecht/Boston/London: Kluwer Academic Publishers.

Floudas, C. A., & Pardalos, P. M. (Eds.). (2003). *Frontiers in global optimization*. Kluwer Academic.

Frenk, J. B. G., & Schaible, S. (2004). *Fractional programming*. Erasmus Research Institute of Management (ERIM), ERS-2004-074-LIS.

Holmberg, K., & Tuy, H. (1993). A production-transportation problem with stochastic demands and concave production cost. *Mathematical Programming, 85*, 157–179.
Horst, R., & Pardalos, P.M. (Eds.). (1995). *Handbook of global optimization*. Kluwer Academic.
Horst, R., & Tuy, H. (1996). *Global optimization (Deterministic approaches)*. (3rd ed.). Springer.
Konno, H., Thach, P. T., & Tuy, H. (1997). *Optimization on low rank nonconvex structures*. Kluwer Academic.
Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization, 26*, 369–395.
Pardalos, P. M., & Coleman, T. F. (Eds.). (2009). *Lectures on global optimization: Vol. 55. Fields institute communications series*. American Mathematical Society.
Pardalos, P. M., & Romeijn, H. E. (Eds.). (2002). *Handbook of global optimization*. Springer.
Rebennack, S., Kallrath, J., & Pardalos, P. M. (2009). Column enumeration based decomposition techniques for a class of non-convex MINLP problems. *Journal of Global Optimization, 43*(2–3), 277–297.
Rebennack, S., Pardalos, P. M., Pereira, M. V. F, & Iliadis, N. A. (Eds.), (2010a). *Handbook of power systems I*. Energy Systems series, Springer.
Rebennack, S., Pardalos, P. M., Pereira, M. V. F., & Iliadis, N. A. (Eds.). (2010b). *Handbook of power systems II*. Energy Systems series, Springer.
Sherali, H. D., & Adams, W. P. (1999). *A reformulation–linearization technique for solving discrete and continuous nonconvex problems*. Dordrecht/Boston/London: Kluwer Academic Publishers.
Stancu-Minasian, I. M. (1997). *Fractional programming: Theory, methods and applications*. Dordrecht/Boston/London: Kluwer Academic Publishers.
Thach, P. T., Konno, H., & Yokota, D. (1996). Dual approach to minimization on the set of pareto-optimal solutions. *Journal of Optimization Theory and Applications, 88*(3), 689–707.
Tuy, H. (1998). *Convex analysis and global optimization*. Kluwer (Springer).
Tuy, H. (2010). DC-optimization and robust global optimization. *Journal of Global Optimization, 47*, 485–501. doi:10.1007/s10898-009-9475-2.
Wen, U.-P., & Hsu, S.-T. (1991). Linear bilevel programming problems – A review. *Journal of the Operational Research Society, 42*, 125–133.

Global Solution

An optimal solution over the entire feasible region.

See

► Global Optimization

Goal Constraints

Mathematical expressions consisting of resource utilization rates, decision variables, deviation variables, and targeted minimum and maximum resources levels. They are used to model individual resource goals in a goal programming model.

Goal Programming

Marc J. Schniederjans
University of Nebraska-Lincoln, Lincoln, NE, USA

Introduction

Goal Programming (GP), also called linear goal programming (LGP), can be categorized as a special case of linear programming (LP). The origin of GP as a means of resolving infeasible LP problems attests to its characterization as a LP methodology (Charnes and Cooper 1961). GP is now considered a multi-criteria decision making (MCDM) method (Steuer 1986); it is used to solve multi-variable, constrained resource and similar problems that have multiple goals.

GP Modeling

Similar to LP, the GP model has an objective function, constraints (called goal constraints), and nonnegativity requirements. The GP objective function is commonly expressed in minimization form as (Schniederjans 1984):

$$\text{minimize} \quad Z = \sum_{i=1}^{T} w_{kl} P_k \left(d_i^- + d_i^+\right)$$
$$\text{for} \quad k = 1, \ldots, K;\ l = 1, \ldots, L,$$

where i is the goal constraint index, k is the priority rank index, and l is the index of the deviation variables within priority rank. In the objective function, Z is the summation of all deviations, the w_{kl} are optional mathematical weights used to differentiate deviation variables within a kth priority level, the P_k are optional rankings of deviation variables within goal constraints,

the d_i^- values are the negative deviational variables, and the d_i^+ values are the positive deviational variables. The P_k rankings are called preemptive priorities because they establish an ordinal priority ranking (where $P_1 > P_2 > P_3 > \ldots$ etc.) that orders the systematic optimization of the deviation variables.

The fact that the optimization of the variables in the objective function is ordered by the preemptive priority ranking has given rise to the use of the term satisficing. This term results from the fact that a solution in a GP model satisfies the ranking structure while minimizing deviation from goals. [See Simon (1955) for his original definition of satisficing with respect to rational choice.] One of the best features of GP is that the P_k permit the decision makers to rank goals in accordance with their personal preferences, and even weight the importance of those preferences within goals using w_{kl}. The greater the w_{kl} mathematical weighting, the greater the importance attached to its related deviation variable.

The goal constraints in a GP model can be expressed as:

$$\sum_j a_{ij}x_j + d_i^- - d_i^+ = b_i \quad \text{for all } i.$$

The a_{ij} are the resource utilization rates representing the per unit usage of the related resource b_i, and the x_j are the decision variables to be determined. The goal constraints thus seek minimize the deviations from the each goal constraint's right-hand-side b_i goal targets. In essence, this use of deviation variables minimizes the absolute difference between the right-and left-hand sides of each constraint. The d_i^- are termed underachievement variables and the d_i^+ are overachievement variables. The nonnegativity requirements in a GP model are usually expressed as:

$$\left(x_j, d_i^-, d_i^+\right) \geq 0 \quad \text{for all } i, j.$$

When preemptive priorities are not established, the GP model takes on the form

$$\begin{aligned} \text{minimize} \quad & Z = \sum_i \left(w_i^- d_i^- + w_i^+ d_i^+\right) \\ \text{subject to:} \quad & \sum_j a_{ij}x_j + d_i^- - d_i^+ = b_i \quad \text{for all } i \\ & \left(x_j, d_i^-, d_i^+\right) \geq 0 \quad \text{for all } i, j. \end{aligned}$$

The w_i^- and w_i^+ are positive, negative or zero weights. A constraint may not be a goal constraint in that one can let $d_i^- = d_i^+ = 0$ if the condition must be met exactly, as, for example, a fixed budget condition. Such constraints are said to be hard. In contrast, goal constraints are said to be soft, as their goals can be underachieved, overachieved or met exactly, as, for example, a production requirement.

GP Solution Methods

Different solution methodologies exist to solve a variety of types of GP models. The type of GP model depends on special requirements placed on the decision variables in the model. Borrowing from LP, most of the solution methodologies for GP models are based on the revised simplex method. Simplex based-solution methods for GP problems originate from the sequential goal (preemptive priority) procedure of Lee (1972). There are additional methodologies for solving integer GP problems, *zero-one* GP problems and nonlinear GP problems. Like LP, these special types of GP solution methods are based on revised simplex methods, enumeration methods, and the calculus.

Duality and sensitivity analysis information can also be obtained from the simplex based GP solution methods (Ignizio 1982). Duality in GP models is focused on examining trade-offs in deviation between priorities. The software system by Lee and Shim (1993) computes the marginal trade-offs of revising right-hand-side b_i goal targets to reduce deviation from lower priority goals. There are a variety of LP-based sensitivity analysis procedures for GP. Unique to GP is P_k -sensitivity analysis (Schniederjans 1995). In P_k -sensitivity analysis, alterations in sets of k priority level goals are implemented to examine their ordering effect upon the model's solution. Other issues in modeling and overcoming problems with GP methodologies can be found in Romero (1991).

GP Research and Applications

While both GP modeling and GP solution methods share LP origins, there are two characteristics of GP that differentiate the application of GP from LP

problems: multiple goals and an ordinal ranking of the goals to deal with conflict. Since many business and governmental problems contain the same two characteristics, GP became a very popular methodology in analyzing such problems.

During the 1960s and early 1970s, most research on GP focused on revisions of prior LP-type models, but with a ranking of conflicting goals. The series of case applications presented in Lee (1972) typify the work during this period. The most common applications followed functional areas in business, such as budgeting in accounting, portfolio analysis in finance, production planning in management, and advertising resource allocation in marketing. Having examined these areas of application in depth, interest in GP started to wane. In the late 1970s and early 1980s, however, integer (particularly zero-one) GP methodologies appeared and caused a renewed interest in the application of GP. Zero-one GP solution methods permitted the model to be applied in binary outcome situations and broadened the potential application base of GP. Zero-one GP applications include project selection, personnel selection, and logistics. This period also saw the combining of other operations research and management science methodologies within a GP model: the transportation simplex method, assignment method, network models, nonlinear programming, dynamic programming, simulation, game theory, fuzzy programming and heuristic procedures (Steuer 1986).

In the late 1980s and through the 1990s, GP micro-computer software was developed, placing fairly powerful solution capabilities in the hands of practitioners, thus causing another burst of interest in GP applications: planning in small businesses, improving productivity in service operations, and planning product development. GP engineering applications include: planning flexible manufacturing systems, robot selection, strategic planning, metal cutting and inventory lot sizing. Improvements in GP weighting strategies have been developed using the analytic hierarchy process (AHP) (Liao and Kao 2010) and regression analysis (Garcia et al. 2010). GP has also been combined with the analytic network process (ANP) to solve capital asset pricing problems (Aznar et al. 2010).

A GP model's ability to use personal preference information has made it a very useful tool in dealing with socially sensitive issues. Throughout GP's history, applications and models have illustrated how GP is a powerful tool for analyzing public policy issues. Applications include: weapon system selection (Lee et al. 2010) and labor market satisfaction (Marcenaro-Gutierrez et al. 2010).

See

- Analytic Hierarchy Process
- Linear Programming
- Multiobjective Programming
- Multiple Criteria Decision Making
- Nonlinear Programming
- Regression Analysis
- Simplex Method (Algorithm)

References

Aznar, J., Ferris-Onate, J., & Guijarro, F. (2010). An ANP framework for property pricing combing quantitative and qualitative attributes. *Journal of the Operational Research Society, 61*, 740–755.

Caballero, R., Ruiz, F., & Steuer, R. E. (Eds.). (1997). *Advances in multiple objective and goal programming*. Berlin: Springer.

Charnes, A., & Cooper, W. W. (1961). *Management models and industrial applications of linear programming*. New York: John Wiley.

Garcia, F., Guijarro, F., & Moya, I. (2010). A goal programming approach to estimating performance weights for ranking firms. *Computers and Operations Research, 37*, 1597–1609.

Ignizio, J. (1982). *Linear programming in single-and multiple-objective systems*. Englewood Cliffs, NJ: Prentice-Hall.

Jones, D., & Tamiz, M. (2010). *Practical goal programming*. New York: Springer.

Lee, S. M. (1972). *Goal programming for decision analysis*. Philadelphia: Auerbach Publishers.

Lee, J., Kang, S.-H., Rosenberger, J., & Kim, S. B. (2010). A hybrid approach of goal programming for weapon systems selection. *Computers and Industrial Engineering, 58*, 521–527.

Lee, S. M., & Shim, J. P. (1993). *Micro management science* (3rd ed.). Boston: Allyn and Bacon.

Liao, C.-N., & Kao, H.-P. (2010). Supplier selection model using taguchi loss function, analytical hierarchy process and multi-choice goal programming. *Computers and Industrial Engineering, 58*, 571–577.

Marcenaro-Gutierrez, O. D., Luque, M., & Ruiz, F. (2010). An application of multiobjective programming to the study of workers' satisfaction in the Spanish labor market. *European Journal of Operational Research, 203*, 430–443.

Min, H., & Storbeck, J. (1991). On the origin and persistence of misconceptions in goal programming. *Operational Research Society, 42*, 301–312.
Romero, C. (1991). *Handbook of critical issues in goal programming*. New York: Pergamon.
Schniederjans, M. J. (1984). *Linear goal programming*. Princeton, NJ: Petrocelli Books.
Schniederjans, M. J. (1995). *Goal programming: Methodology and applications*. Boston: Kluwer.
Simon, H. A. (1955). A behavioral model of rational choice. *Quarterly Journal of Economics, 19*, 99–118.
Steuer, R. E. (1986). *Multiple criteria optimization: Theory, computation, and application*. New York: John Wiley.
Tamiz, M. (Ed.). (1996). *Multi-objective programming and goal programming*. Berlin: Springer.
Zanakis, S. H., & Gupta, S. (1985). A categorized bibliographic survey of goal programming. *Omega, 13*, 211–222.

Gomory Cut

A linear constraint that is added to a linear-programming problem to reduce the solution space without cutting off any integer-valued points. Such cutting planes are the basis of many solution procedures that find integer solutions to a linear constrained optimization problem. The idea is to eventually reduce the solution space so that its optimal integer solution corresponds to an extreme point of the reduced solution space.

See

► Integer and Combinatorial Optimization
► Linear Programming

Gordan's Theorem

Let $\boldsymbol{A}$ be an $m \times n$ matrix, then exactly one of the following systems has a solution: (i) $\boldsymbol{Ax} < \boldsymbol{0}$ or (ii) $\boldsymbol{A}^T \boldsymbol{y} = \boldsymbol{0}, \boldsymbol{y} \geq \boldsymbol{0}, \boldsymbol{y} \neq \boldsymbol{0}$.

GP

► Goal Programming

Gradient Vector

For the function $f(\boldsymbol{x})$ of the vector $\boldsymbol{x}$, the gradient is the vector of the first partial derivatives (if they exist) evaluated at a specific point $\boldsymbol{x}^{\boldsymbol{0}}$ and is written as

$$\nabla f\left(\boldsymbol{x}^{\boldsymbol{0}}\right) = \left[\frac{\partial f\left(\boldsymbol{x}^{\boldsymbol{0}}\right)}{\partial x_1}, \frac{\partial f\left(\boldsymbol{x}^{\boldsymbol{0}}\right)}{\partial x_2}, \ldots, \frac{\partial f\left(\boldsymbol{x}^{\boldsymbol{0}}\right)}{\partial x_n}\right].$$

It is normal or perpendicular to the tangent of the contour of $f(\boldsymbol{x})$ that passes through $\boldsymbol{x}^{\boldsymbol{0}}$. Its direction is the direction of maximum increase of $f(\boldsymbol{x})$ and its length is the magnitude of that maximum rate of increase.

Graeco-Latin Square

► Combinatorics

Graph

A graph $G = (V, E)$ consists of a finite set V of vertices (nodes, points) and a set E of edges (arcs, lines) joining different pairs of distinct vertices.

Graph Theory

Douglas R. Shier
Clemson University, Clemson, SC, USA

Introduction

Graph theory is the general study of the interconnection of various elements. While the origins of graph theory can be traced back to the eighteenth century, this area of discrete mathematics experienced most of its tremendous growth during the latter half of the twentieth century. This rapid growth, both in the development of new theory and applications, reflects the fact that graphs can model a wide variety of natural and technological systems.

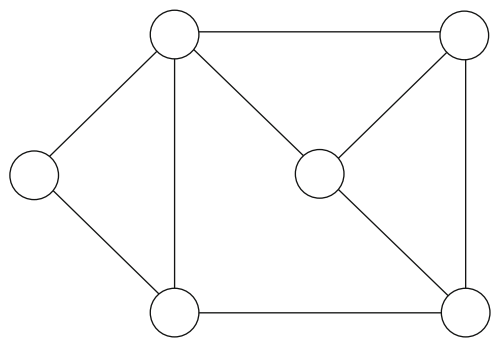
A number of physical systems can be viewed as graphs, composed of nodes (or vertices) connected together by edges (or arcs). For example, a local area computer network defines a graph whose nodes represent individual computers (or peripheral devices) and whose edges represent the physical cables connecting such computers. A telecommunication network consists of access points (and central switching stations) joined by sections of copper wire (and optical fibers); an airline system has airports as its nodes and direct flights as its edges; a street network involves road segments (edges) whose intersections define its nodes; and an electronic switching circuit contains logic gates whose input and output leads form a graph.

In addition, graphs can with equal ease represent logical relationships between elements. For example, the subroutines of a computer program can be represented as nodes of a graph, with edges indicating the flow of control or data between subroutines. A project involving a large number of tasks can be modeled by a graph, with the tasks being nodes and logical precedence relations defining the edges. In an ecological system the edges could indicate which species (nodes) feed upon other species. Examination scheduling at a university can be studied using a graph whose nodes are courses and whose edges indicate whether two courses contain students in common; examinations for such adjacent courses should not be scheduled at the same time. Table 1 shows a sample of other application areas in which graph models provide a useful representation.

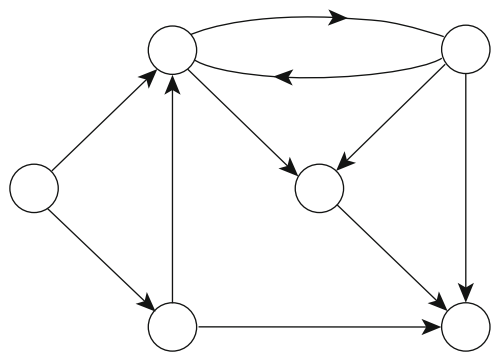
As suggested by these applications, the direct connections between nodes can be bidirectional (such as in making a telephone call, or in traversing a major highway) or there can be a specific orientation implied by the relationship (as the precedence relation in a project graph, or the predator–prey relation in an ecological graph). Consequently, graph theory treats both undirected graphs (in which the underlying relationship between nodes is symmetric) and directed graphs, or digraphs (in which the relationship need not be symmetric). These two graph models are pictured in Figs. 1 and 2, respectively. In this exposition, the focus is on undirected graphs, since the analogous concepts for digraphs are usually apparent. Throughout $G = (N, E)$ will indicate an undirected graph with node set N and edge set E.

Graph Theory, Table 1 Graph models

Application area	Nodes	Edges
Information theory	Strings of binary digits	Single-bit changes
Radio broadcasting	Broadcast stations	Interference
Genealogy	Family members	Parent/child relation
Sociology	Individuals	Interaction patterns
Architecture	Rooms	Accessibility
Electronics	Junctions	Wires
Personnel assignment	Applicants, jobs	Compatibility
Politics	Nations	Alliances
Genetics	Chromosome segments	Overlapping
Engineering	Joints	Beams
Commerce	Web sites	Hyperlinks



Graph Theory, Fig.1 An undirected graph



Graph Theory, Fig. 2 A directed graph

One of the earliest applications of graph theory was to the structure of molecular compounds, in which atoms (nodes) are joined by chemical bonds (edges). The task of identifying which chemical compounds are

structurally the same is reflected in the graph-theoretic concept of isomorphism, meaning that two given graphs are the same up to relabeling of their nodes. In addition, each atom has a valency that indicates the number of other atoms to which it is connected. In graph-theoretic terms this is called the degree of the node, the number of edges with which it is incident. This concept provides a quantifiable measure of local connectivity. For instance, the degree of a node in a communication network indicates the relative burden on that node in transporting information, so a robust communication system would be designed to avoid nodes with large degrees. Since such systems support point-to-point communication, a more global measure of connectivity is also needed. Thus, a fundamental concept is that of a path between nodes i and j: an alternating sequence of nodes and incident edges leading from node i to node j. A cycle is a closed path. The graph G is connected if every pair of distinct nodes is joined by a path in G. The distance between two nodes of G is defined as the smallest number of edges in a path joining the nodes. Then an overall measure of compactness of the graph is given by its diameter: the maximum distance between any two of its nodes.

Eulerian and Hamiltonian Cycles

In certain applications, specific types of paths or cycles are sought in the graph. For example, an Eulerian cycle is a cycle in the graph G that traverses each edge of G exactly once. This concept models the task of efficiently routing trucks for collection of trash throughout a city, since multiple passes along a road are not desirable. Another application occurs in planning police beats, in which every street of an area needs to be patrolled. A Hamiltonian cycle in G is a cycle in the graph that visits each node exactly once. This concept has been applied to the temporal sequencing of artifacts found at archaeological sites, the manufacture of electronic circuit boards, DNA mapping, order picking in a warehouse, and vehicle routing. The concepts of Eulerian paths and Hamiltonian paths are defined analogously.

In designing a logistics system it seems prudent to require several paths joining nodes i and j, thus providing redundant routes for sending messages in case of node or edge failures. For example, an adversary might select various edges (bridges, roads) for destruction in order to disrupt the flow of materiel from node i to node j. An i-j cutset is a minimal subset of edges whose removal disconnects i from j in G. To disrupt communication between i and j in an efficient manner, the adversary might then attack an i-j cutset having the minimum size (number of edges) λ_{ij} (G). The celebrated max flow-min cut theorem of networks (Gross and Yellen 1999) shows that the maximum number of edge-disjoint paths joining i and j equals the minimum number of edges in an i-j cutset. An analogous conclusion holds if the paths are node-disjoint and the cutsets are defined in terms of nodes. This min-max relationship is known as Menger's theorem (Chartrand and Zhang 2005).

Trees

A related concept addresses the connectivity of all nodes, rather than just a specified pair. A tree is a connected graph containing no cycles, and a spanning tree of the graph $G = (N, E)$ is a tree with node set N and whose edge set is a subset of E. Any spanning tree thus supports communication among all nodes of the graph. On the other hand, a cutset of G is a minimal set S of edges whose removal disconnects some pair of nodes in the graph. The edges of S must intersect the edge set of every spanning tree of G. An overall connectivity measure for the graph G is the minimum size $\lambda(G)$ of a cutset in G. Interestingly, the number of spanning trees of a graph can be computed efficiently, using the matrix-tree theorem (Chartrand and Zhang 2005); by contrast, counting the number of cutsets is *NP*-hard.

Trees find many other applications in the theory of graphs. Trees can be used to model the organizational hierarchy of a corporation, the table of contents of a book, the possible evolutionary history of species, or the syntactic structure of languages. Trees also serve as useful data structures for organizing elements of a database for subsequent retrieval and updating. For example, the branch-and-bound method for integer-programming problems is implemented using a tree structure. In addition, trees are used by compilers of computer languages to provide a concise representation of arithmetic expressions. Various ways of traversing trees (in particular depth-first search and breadth-first search) are important in

designing efficient algorithms for network optimization problems. Also of relevance to operations research is the fact that spanning trees correspond exactly to basic solutions for linear programming problems formulated on graphs.

Evolving Networks

What accounts for the structure of real-world networks and how do they continue to evolve? These questions have led to the investigation of small-world networks and scale-free networks. Rather than following the classic random networks studied by Erdös and Rényi (Watts 2003), many important networks such as the World Wide Web, social networks, cellular metabolic networks, and airline networks have characteristics quite different from those of random networks. In one model of random networks, the adjacencies of each node are randomly selected (controlled by a fixed connection probability p); in another model, a graph with n nodes and m edges is randomly selected from among all graphs having n nodes and m edges. Both models lead to quite similar characteristics. Specifically these classical random graphs have small clustering coefficients and small average (shortest) path lengths. The clustering coefficient simply measures for a typical node the average number of neighbors of the node that are in turn neighbors of one another. The average path length measures for a typical node the average number of steps (edges) needed to reach another node in the most efficient way (fewest number of edges).

By contrast, many real-world networks exhibit substantially larger clustering coefficients: e.g., in a social network, friends of a given node are likely to be friends of one another. In groundbreaking work, Watts and Strogatz (1998) studied the prevalence and construction of small-world networks in which nodes are typically not very far apart (small average path length) but have significantly higher clustering coefficients than in random networks. Another model for real-world graphs was first developed by Barabási and Albert (1999). These scale-free networks also have small average path lengths and significant clustering coefficients, yet typically contain a number of hub nodes with relatively large degree. More specifically, in scale-free networks the distribution of node degrees follows a power law: the proportion of nodes having degree k is proportional to $k^{-\beta}$, where $2 \leq \beta \leq 3$ typically holds. By contrast, the distribution of node degrees in small-world networks follows a binomial distribution. A variety of dynamic graph models have been proposed to account for the creation and expansion of scale-free networks. For example, such networks can grow by means of preferential attachment: new nodes are successively added and (probabilistically) linked to existing nodes according to the degrees of the existing nodes. They can also grow by a certain type of copying mechanism.

Embeddings and Colorings

Special types of graphs find application in the layout of circuit boards, in which it is desired to place the components and their connections so that no two wires meet except at a component. This corresponds to an embedding of the graph in the plane so that edges only intersect at nodes. Kuratowski's theorem provides an elegant characterization of which graphs are in fact planar. More generally, any graph G can be decomposed into a number of edge-disjoint planar subgraphs, and the minimum number of such subgraphs is termed the thickness $\theta(G)$ of the graph. For example, highways are designed to minimize the number of overpasses required. Planar graphs G are also of interest because a dual G^* of such graphs can be defined. In particular, the cycles of G are in one-to-one correspondence with the cutsets of G^*.

Coloring the nodes of a graph also arises in several applications. A proper coloring of G with k colors is an assignment of these colors to the nodes of G such that adjacent nodes are colored differently. For example, if the nodes of G represent courses and edges represent conflicts (courses that cannot have examinations scheduled at the same time), then a proper coloring of G with k colors defines a conflict-free schedule using k time periods. In another application, suppose the graph G indicates a compatibility relationship between tasks. A k-coloring of the complement of G (a graph whose edges are those node pairs not appearing as edges of G) then yields a partitioning of the nodes of G into k groups of mutually compatible tasks. The minimum number of colors $\chi(G)$ needed to color G properly is termed the chromatic number of G. The famous four-color conjecture, proposed in 1852 and finally proved in 1976, states that $\chi(G) \leq 4$ holds for any

planar graph G. Coloring problems also arise in the assignment of frequencies of the electromagnetic spectrum. Locations that are nearby must be assigned different frequencies to avoid interference, and the efficient allocation of frequencies then involves coloring the underlying neighborhood graph using the fewest number of colors.

Matchings

A matching is a set of mutually nonadjacent edges in G. A maximum size matching is a matching of G having the largest number of edges. These concepts arise in various applications, such as assigning personnel to jobs, target tracking, crew scheduling, and scheduling on parallel machines. In another example, pairs of pilots are to be assigned to aircraft that serve international routes. Two pilots are considered compatible if they are fluent in a common language and have comparable flight training. Finding a largest set of aircraft to fly with compatible pilots then requires finding a maximum size matching in the associated compatibility graph. In telecommunications, the problem of exchanging unique pieces of information so that all users will quickly know the totality of information can be viewed as a problem of constructing a sequence of matchings; each matching represents a set of simultaneous exchanges that can be carried out in a single time period. The arrangement of simultaneous kidney exchanges between multiple pairs of compatible (but unrelated) individuals has been carried out using maximum size matchings; see kidney paired donations on the World Wide Web.

Optimization

One important aspect of graphs is that they distill the essential adjacency relationships between objects. Another is that they suggest certain optimization problems. It is apparent that determining the connectivity $\lambda(G)$, thickness $\theta(G)$, and chromatic number $\chi(G)$ are graph optimization problems. Other graph optimization problems arise directly from applications in which it might be required to optimally schedule courses, allocate facilities, route goods, or design computer systems, relative to some objective function and subject to appropriate constraints. As a specific example, it might be required to design a minimum diameter communication graph with $\lambda_{ij}(G) \geq k$ for all distinct node pairs i and j, using a fixed number of edges.

More generally, quantitative information may be associated with the nodes and/or edges of a graph, reflecting the cost, time, distance, capacity, or desirability of these components. A variety of graph optimization problems are then apparent: (1) find a spanning tree of G having minimum cost (minimum spanning tree problem); (2) find a minimum length path joining two nodes of G (shortest path problem); (3) find the maximum amount of material that can feasibly flow from an origin node to a destination node (maximum flow problem); (4) find a minimum cost Hamiltonian cycle in G (traveling salesman problem); (5) optimally locate facilities on the edges of G to serve demands arising at the nodes (facility location problem); (6) find a maximum weight set of nonadjacent edges in G (maximum weight matching problem); (7) find a minimum cost traversal of all edges of G such that each edge is used at least once (Chinese postman problem).

There are additional application areas in which graphs are used to model a variety of physical and logical systems. While the discussion has concentrated on undirected graphs, directed graphs are pertinent in other areas, such as in representing the state diagram of a Markov chain. Connectivity in this digraph G can be used to classify the states of the chain, and the (directed) cycle lengths in G define the periodicity of the chain. Directed graphs are also the basis of project planning models. Clearly the hyperlink structure of the Internet can be modeled with directed edges, indicating which Web pages link to others; the ubiquitous PageRank algorithm (Langville and Meyer, 2006) used by search engines to rank the relevance of Web pages for a search query is carried out on this directed graph. In essence, the algorithm calculates the importance of a Web page recursively by taking into account the importance of each Web page linking to it.

The study of optimization problems on graphs and digraphs has in turn stimulated research into the design of effective algorithms for solving such problems, as well as determining when these problems belong to an inherently difficult class of problems (NP-hard problems). In the latter case, it is important to

identify special types of graphs (e.g., planar graphs) for which the computation can be carried out efficiently, even though there may not exist an efficient solution method applicable to all graphs. Several heuristics for the traveling salesman problem utilize the minimum spanning tree of the graph and an associated Eulerian cycle, both of which can be efficiently computed.

Further Reading

The book by Biggs et al. (1976) provides an excellent reference for the history of graph theory. Bondy and Murty (1979), Fulkerson (1975), Gross and Yellen (1999), Michaels and Rosen (1991), and Roberts (1976) discuss a variety of applications of graphs. Wilson and Watkins (1990) and Chartrand and Zhang (2005) give nice introductions to the theory of graphs, with more advanced treatment provided in Diestel (1997) and West (1996). The books by Barabási (2002), Buchanan (2002), and Watts (2003) are very readable introductions to small-world and scale-free networks. Algorithmic aspects of graph theory are discussed in Evans and Minieka (1992).

See

- ▶ Chinese Postman Problem
- ▶ Computational Complexity
- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Markov Chains
- ▶ Matching
- ▶ Network
- ▶ Project Management
- ▶ Traveling Salesman Problem

References

Barabási, A. (2002). *Linked: The new science of networks*. Cambridge: Perseus.

Barabási, A., & Albert, R. (1999). Emergence of scaling in random networks. *Science, 286*, 509–512.

Biggs, N. L., Lloyd, E. K., & Wilson, R. J. (1976). *Graph theory 1736–1936*. Oxford: Clarendon.

Bondy, J. A., & Murty, U. S. R. (1979). *Graph theory with applications*. New York: Elsevier.

Buchanan, M. (2002). *Nexus: Small worlds and the groundbreaking science of networks*. New York: W.W. Norton.

Chartrand, G., & Zhang, P. (2005). *Introduction to graph theory*. Boston: McGraw-Hill.

Diestel, R. (1997). *Graph theory*. New York: Springer.

Evans, J. R., & Minieka, E. (1992). *Optimization algorithms for networks and graphs*. New York: Marcel Dekker.

Fulkerson, D. R. (1975). *Studies in graph theory, parts I–II, volumes 11–12, MAA studies in mathematics*. Washington, DC: Mathematical Association of America.

Gross, J., & Yellen, J. (1999). *Graph theory and its applications*. Boca Raton: CRC Press.

Langville, A. N., & Meyer, C. D. (2006). *Google's pagerank and beyond: The science of search engine rankings*. Princeton, NJ: Princeton University Press.

Michaels, J. G., & Rosen, K. H. (1991). *Applications of discrete mathematics*. New York: McGraw-Hill.

Roberts, F. (1976). *Discrete mathematical models, with applications to social, biological, and environmental problems*. Englewood Cliffs, NJ: Prentice-Hall.

Watts, D. J. (2003). *Six degrees: The science of a connected age*. New York: W.W. Norton.

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of small-world networks. *Nature, 393*, 440–442.

West, D. B. (1996). *Introduction to graph theory*. Upper Saddle River, NJ: Prentice-Hall.

Wilson, R. J., & Watkins, J. J. (1990). *Graphs: An introductory approach*. New York: Wiley.

Graphical Evaluation and Review Technique

- ▶ GERT

Graphics

- ▶ Visualization

Greedy Algorithm

A heuristic algorithm that at every step selects the best choice available at that step without regard to future consequences. A greedy method never rescinds its choices or decisions made earlier. A greedy method is usually applied to an optimization problem for which the method attempts to determine an optimal solution (least cost, maximum value), with no

guarantee that the optimal solution will be found. Kruskal's and Prim's minimum spanning tree algorithms are greedy methods that do produce an optimal solution.

See

- ▶ Algorithm
- ▶ Heuristic Procedure
- ▶ Kruskal's Algorithm
- ▶ Prim's Algorithm

GRG Method

Generalized reduced gradient method.

See

- ▶ Quadratic Programming

Group Decision Computer Technology

Dennis M. Buede
Innovative Decisions, Inc., Vienna, VA, USA
George Mason University, Fairfax, VA, USA

With the rise of computer technology and the success of quantitative decision support technology, there has been a great deal of interest in moving these technologies into the boardroom, so to speak. There is no commonly accepted approach for supporting group decision making, see Dennis and Gallupe 1993; DeSanctis and Gallupe 1987; and Nunamaker et al. 1993. The oldest approach is decision conferencing (Watson and Buede 1987), which started in 1979 and has spread but not exploded. Decision conferencing is a group process that is led by a decision analytic facilitator. The facilitator employs simple decision analysis models to focus the group's discussion on the objectives, options, and uncertainties. The facilitator also mixes analytic activities with creative problem structuring and option generation activities. The decision conference can be as short as two days or may involve several two to three day sessions. References for decision conferences include Phillips (1984) and Reagan-Cirincione (1992).

There are a range of approaches for group decision support that place a computer in the hands of each participant. This approach still employs a group process facilitator, although there is some disagreement about the importance of the facilitator amongst the researchers and practitioners in this area. The computer technology is designed to enhance the productivity of the individual and the communication of information among individuals. The effectiveness of computer technology as a communication medium when the group has a single decision focus is questioned by some. Computer technology, however, opens the group's options in terms of whether they meet at the same place or even at the same time. The major options of the group have been named: same time, same place; same time, different place; different time, same place; and different time, different place. Substantial work and adaptation has taken place in this area, see Dennis and Gallupe 1993; and Nunamaker et al. 1993.

Developments over time have seen these two extremes of decision conferencing and computer-supported and linked individuals merge. The individuals now have button boxes that feed a single computer via infrared beams. The individuals are able to input their votes or numeric judgments into the computer that displays the results and spurs discussion and debate. The results of the group inputs can then be recorded and incorporated into a broader analysis.

Group process support continues to be an expanding research and application area. For the group process to be considered successful, researchers must show that the group acting with decision support can be more effective than the second most effective of the group (Reagan-Cirincione 1992). The group must be provided with both cognitive support and social support in their activities; there is no lack of options for providing this support (see Connolly 1993; Nunamaker et al. 1993; and Huber et al. 1993).

See

- ▶ Group Decision Making

References

Connolly, T. (1993). Behavioral decision theory and group support systems. In L. Jessup & J. Valacich (Eds.), *Group support systems*. New York: Macmillan.

Dennis, A., & Gallupe, R. (1993). A history of group support systems empirical research: Lessons learned and future directions. In L. Jessup & J. Valacich (Eds.), *Group support systems*. New York: Macmillan.

DeSanctis, G., & Gallupe, R. (1987). A foundation for the study of group decision support systems. *Management Science, 33*, 589–609.

Nunamaker, J., Dennis, A., Valacich, J., Vogel, D., & George, J. (1993). Group support systems research: Experience from the lab and field. In L. Jessup & J. Valacich (Eds.), *Group support systems*. New York: Macmillan.

Phillips, L. (1984). A theory of requisite decision modeling. *Acta Psychologica, 56*, 29–48.

Reagan-Cirincione, P. (1992). Combining group facilitation, decision modeling, and information technology to improve the accuracy of group judgment. In J. Nunamaker, & R. Sprague, (Eds.), *Proceedings of the Hawaii international conference on system sciences* (Vol. IV), Los Alamitos, CA: IEEE Computer Society Press.

Watson, S., & Buede, D. (1987). *Decision synthesis*. Cambridge, England: Cambridge University Press.

Group Decision Making

Fatemeh Mariam Zahedi
University of Wisconsin-Milwaukee, Milwaukee, WI, USA

Introduction

Group decision making focuses on problems in which there is more than one decision maker and more than one choice. The choices or alternatives have multiple attributes. In other words, the decision makers must consider more than one objective or criterion in their decision. Hence, group decisions involve multiple criteria and multiple decision makers. Since preferences and objectives of individual decision makers vary and may be in conflict, arriving at a decision is far more complex in a group setting than in individual cases.

Group decision covers a wide range of collective decision processes and encompasses numerous methods designed under various assumptions and for different circumstances. One can divide the group decision approaches into the following categories: utility theory, group consensus, group analytic hierarchy process, social choice theory, and game theory.

Group Utility Analysis

Group utility theory is based on the von Neumann-Morgenstern utility function. This method assumes that there is a multicriteria utility function U_i (x_1, x_2,. . ., x_m), where i represents member i, x_m represents the mth attribute, and m is the number of attributes. Based on the assumption that the utilities of members are functionally independent, the group utility function is computed as the aggregation of the member utility functions by one of the following two function types.

The additive form of the function is

$$U = \sum_{i=1}^{n} W_i U_i$$

where n is the number of attributes, and the multiplicative form is

$$wU + 1 = \sum_{i=1}^{n} (ww_i U_i + 1),$$

where w and w_i are scaling constants satisfying $0 < w_i < 1$, and $w > -1$, and $w \neq 0$. The estimation of member utility functions follows the assumptions and procedures used in estimating the individual multicriteria utility functions. The important question in the group utility estimation is the determination of the scaling constants. Keeney and Kirkwood (1975) suggested that these constants could be determined either by a benevolent dictator or internally by group members.

To resolve the problem of assigning weights to the utility functions of group members, two methods have been proposed. First, Bodily (1979) suggested the delegation process. This method is an iterative process for combining the utility functions of group members. The idea is that each member should assign weights to other members. This process assumes that each member is adequately familiar with the views and utilities of other members. Each member replaces his or her utility by linearly combining other members'

utilities. Members will not know the weights assigned to them by others. The method consists of the following steps:

Step 1. A delegation subcommittee for member i is formed consisting of the remaining $n-1$ members. Member i assigns a weight w_{ij} (a value between 0 and 1) to member j, and repeats the weight assignment for all $n-1$ members. The $n-1$ assigned weights should sum to 1. The weight for member i is 0; that is, $w_{ii}=0$.

Step 2. The combined utilities of the delegation subcommittee are computed as

$$u_i^1 = \sum_{j=1}^{n} w_{ij} u_j,$$

and replaces member i's utility. This process is repeated for all members.

Step 3. Step 2 is iterated for a second time as

$$u_i^2 = \sum_{j=1}^{n} w_{ij} u_j^1,$$

and for the rth time as

$$u_i^r = \sum_{j=1}^{n} w_{ij} u_j^{r-1}.$$

In matrix form, the iteration can be represented as

$$U^r = PU^{r-1},$$

where

$$U^r = \left[u_1^r, u_2^r, \ldots, u_n^r\right].$$

If the process is repeated adequately, one can show from a theorem in Markov processes that under certain conditions, $\boldsymbol{U}^r$ converges and represents the group utility function.

Brock Method

Brock (1980) developed a method for estimating the weights for aggregating the utilities. The Brock Method is based on the assumptions that the solution for the group decision should be Pareto optimal, obtained from the additive combination of member utilities, and that utility gains should be distributed based on the needs of the affected parties. The needs are defined as the intensity of desire, computed as

$$\frac{u_i - d_i}{u_j - d_j} = -\frac{du_i}{du_j} \quad \forall i, j.$$

Brock showed that the relative weights of members' utility functions are the reciprocals of the above coefficients.

Group Consensus

Group consensus methods combine the observed preferences of members to create consensus points. These points are used to estimate the consensus function for the group. Group consensus methods do not require explicit estimation of member utility functions and may not necessarily lead to the estimation of a function for the group. This approach is in contrast with the utility approach, in which the utility functions of members are estimated, then combined to arrive at a group utility function.

Krzysztofowicz Method — The Krzysztofowicz method (1979) is based on the following assumptions:

1. The group utility function can be decomposed into functions (W_i) of its attributes (x_i) and these functions could be combined via another function (H) such that:

$$W(x_1, x_2, \ldots, x_n) = H(W_1(x_1), W_2(x_2), \ldots, W_n(x_m))$$

2. where $W_i(x_i)$ is the group marginal utility of attribute x_i and m is the number of attributes relevant to the group decision.
3. The group's observed preference is the result of combining members' observed preferences by the decision rule d.
4. The group members are divided into subgroups of experts. Each subgroup has expertise in a subset of attributes, and each subgroup is responsible for estimating $W_i(x_i)$.

5. Each member and subgroup behave according to the axioms of utility theory.

In this method, the group is divided into sub-groups. Each subgroup estimates $W_i(x_i)$ based on its expertise. The group's marginal utility functions of attributes are then combined by H, which is either an additive or multiplicative function, similar to those in the group utility theory.

In the subgroup estimation of $W_i(x_i)$, the expressed preferences of members are combined by the decision rule d. This leads to a series of consensus points from which the function $W_i(x_i)$ is estimated.

Zahedi Group Consensus Method — This method (1986a) is based on the following assumptions:

1. Preferences of individual members are uncertain.
2. The relative weight (or importance) of a member is inversely proportional to his or her degree of uncertainty in his or her response.
3. Standard deviation is the measure of uncertainty.
4. A member's preference response has a normal probability distribution.
5. Correlations among members remain constant over various alternatives.
6. A consensus point is generated by combining the members' expressed preference responses such that the combined point has the minimum variance or uncertainty.
7. The consensus function is estimated based on the generated consensus points.

Based on the above assumptions, the following steps lead to the estimation of the group consensus function:

Step 1. For each multicriteria alternative a, member i assigns an interval score $[x_{ai}, y_{ai}]$.

Step 2. Estimate the mean and standard deviation of the interval by

$$\begin{cases} \hat{U}_{ai} = \frac{y_{ai}+x_{ai}}{2} \\ \hat{\sigma}_{ai} = \frac{y_{ai}-x_{ai}}{6} \end{cases}$$

Step 3. Compute group member correlations by

$$\hat{\rho}_{ik} = \frac{\mathrm{Cov}\left(\hat{U}_i, \hat{U}_k\right)}{\sqrt{\mathrm{Var}\left(\hat{U}_i\right) \cdot \mathrm{Var}\left(\hat{U}_k\right)}},$$

where i and k are members. Form a covariance matrix among group members for alternative a by using the standard deviations obtained in Step 2 and pairwise covariance obtained in Step 3. This matrix is symmetric of size n, where n is the number of group members. The main diagonal elements are the variances of n members. The off-diagonal element of row i and column j is $\hat{\sigma}_{ai}\,\hat{\rho}_{ik}$.

Step 4. Compute member i's weight for alternative $a(w_{ia})$ by

$$w_{ia} = \frac{\sum_{k=1}^{n} \alpha_{ika}}{\sum_{h=1}^{n}\sum_{k=1}^{n} \alpha_{hka}},$$

where α_{hka} is the element of the inverse of the covariance matrix computed at Step 3.

Step 5. Compute the consensus point for alternative a by using the results of Steps 2 and 4 in

$$\hat{U}_a = \sum_{i=1}^{n} w_{ia}\hat{U}_{ia}.$$

Step 6. The consensus point could be used directly for selecting the alternative with the highest consensus value. Furthermore, one can estimate the group consensus function by using the consensus points as the dependent variable in a regression analysis in which the independent variables are attribute values.

In the Zahedi method, consensus values and the consensus function are obtained directly from the preference responses of members. It does not assume the existence of utility axioms and does not require members' utility estimation.

Nominal Group Technique — The nominal group technique was first proposed by Delbecq and Van de Ven (1971). The idea of nominal group technique became one of the methods of consensus generation in total quality management (TQM). In this technique, the ideas are generated in silence and recorded, then they are discussed in group, their importance is voted upon, and the final vote is taken. It has the following steps:

Step 1. The team leader presents the group with the description of the problem and each member records his or her idea or solution individually in silence.

Step 2. The leader asks members to express their ideas one at a time and records them on a chart.

Step 3. Members discuss the recorded ideas so that all members understand each idea.

Step 4. Each idea is voted upon and ranked by members and the average ranking for each idea is computed.

Step 5. Another round of discussions clarifies the position of various members.

Step 6. The final vote is taken by a procedure similar to that of Step 4.

Delphi Method — The Delphi method, developed by Dalkey (1967), is used for generating consensus among members who are not in the same location. It involves written questionnaires and written answers. In this method, the group leader identifies the problem or the question, identifies group members, and contacts them. A sample of group members is selected by the group leader. The method goes through iterations of the following steps:

Step 1. Design the questionnaire to be answered by the selected group members.

Step 2. Have members complete the questionnaire.

Step 3. Analyze responses, make changes in the questionnaire, and include the aggregated responses from the previous round. Ask the members to react to the results of the previous round.

After a number of iterations, the final results are computed, and alternatives are ranked accordingly.

Iterative Open Planning Process — Ortolano (1974) suggested the open planning process. In this method, the activities are divided into four stages: problem identification, plan formulation, impact assessment, and evaluation. There are two sets of decision makers: planners and the affected public. Planners and the public interact at each stage.

- At the problem identification stage, planners determine and evaluate factors from many perspectives and the public articulates problems and concerns.
- At the plan formulation stage, planners delineate alternatives and the affected public suggests alternatives.
- At the impact assessment stage, planners forecast and describe the impacts, while the affected public assists them in describing the impacts.
- At the evaluation stage, planners organize and display information on alternatives and impacts, and the affected public evaluates impacts, makes tradeoffs, and expresses preferences.

These stages take place concurrently and the planners and the affected public repeat the process a number of times.

Group Analytic Hierarchy Process

The analytic hierarchy process (AHP) method was developed by Saaty (1977) and extended to group decision making by Aczel and Saaty (1983). In this method, the alternatives receive a score computed via AHP. This method does not require estimation of utility functions and does not assume axioms of utility analysis. The group AHP method consists of the following steps (Zahedi 1986b).

Step 1. A decision hierarchy is created based on the nature of the decision problem. This hierarchy has multiple levels. At the upper-most level, the decision goal is specified as selecting the best alternative. The next level consists of categories of the attributes that are of importance in the group decision. The next level details each category of attributes into finer and more tangible features. The lowest level of the hierarchy contains the decision alternatives.

For example, for selecting the best car, the highest level of the hierarchy has the selection of the best car as its only element. The second level of the hierarchy includes cost, safety, and design attributes. At the third level of the hierarchy, these attributes are divided into more specific attributes. For example, the cost attribute may be divided into purchase price, preventive maintenance costs, and repair costs at the third level. The safety attribute may be divided into accident outcomes and frequency of breakdowns. The design attribute may be divided into esthetics, driver comfort, and space. The fourth level of the hierarchy includes the decision alternatives — cars to be selected — say, Toyota, Ford, and GM.

Step 2. At each level of the hierarchy, elements are compared pairwise for their role or importance in each element on the level immediately above. The input matrix for the pairwise comparisons has the following form:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix}$$

where $a_{ij} = 1/a_{ji}$ for all $i, j = 1, 2, \ldots, n$, $a_{ii} = 1$, and n is the number of elements of one level being compared

pairwise for their role in accomplishing one of the elements on the upper level.

For example, Toyota, Ford, and GM cars could be compared pairwise in their purchase price to get a pairwise comparison matrix A. The size of this matrix will be 3. One such matrix is needed for each of the elements of the prior level — purchase price, preventive maintenance costs, repair costs, accident outcomes, frequency of breakdowns, esthetics, driver comfort, and space.

Step 3. At this step, a computational method is used to reduce the matrix of pairwise comparisons into a vector of local relative weights. The best known and most widely used computational method is the eigenvalue method, in which the local relative weights are computed as $AW = \lambda W$, were W is the vector of local relative weights, which is the largest eigenvector of $\boldsymbol{A}$, and λ is the largest eigenvalue of matrix $\boldsymbol{A}$.

Step 4. At this step, the local relative weights are combined to arrive at one vector of global relative weights for alternatives at the lowest level of the hierarchy in accomplishing the goal specified at the highest level of the hierarchy. The alternative with the highest global relative weight is the best choice, according to the AHP.

Applied to the group decision setting, the group must reach a consensus as to the structure of the hierarchy at Step 1. At Step 2, there will be a matrix of pairwise comparison elicited for each member of the group. The group pairwise matrix is computed from combining the member matrices. Each element of this matrix is the geometric average of the corresponding elements of the member matrices.

For example, assume that there are four decision makers involved in the decision to purchase a car. When three cars are compared pairwise for purchase price, one matrix of pairwise comparison is created when there is only one decision maker. When there are four decision makers, there will be four such matrices. To compute the group matrix for comparing cars, g_{ij}, the ijth elements of the four matrices are multiplied and then raised to one-fourth power, so that $g_{ij} = (a_{ij}^1\ a_{ij}^2\ a_{ij}^3\ a_{ij}^4)^{1/4}$, where the superscript on a_{ij} represents the decision maker and g_{ij} is the geometric mean of the four decision makers' pairwise values. Steps 3 and 4 of the group AHP are the same as those of the single-decision-maker AHP.

Extensions of The Group Analytic Hierarchy Process

The group AHP has been extended in a number of ways by synthesizing it with other methods and approaches, such as entropy optimization, Bayesian estimation procedure, and data envelopment analysis.

1. Gass and Rapcsák (1998) have extended the group AHP method to the case where the decision makers do not have equal voting power. In their formulation, there is one relative-weight vector $\boldsymbol{W}_i$ for each decision maker $i = 1, 2, \ldots, m$. They formulate an optimization problem that minimizes one of the Hölder-Young distances of the weighted sum of the relative weight vectors from the unknown vector $x = (x_1, x_2, \ldots, x_n)$. The solution to this entropy optimization problem is

$$x_j = \left[\sum_{i=1}^{m} \left(\frac{v_i}{v}\right) w_{ij}^{\alpha}\right]^{1/\alpha}, \quad j = 1, 2, \ldots, n,$$

where v_i is the voting power of decision maker i, and

$$v = \sum_{i=1}^{m} v_i.$$

In this set of solutions, α varies from 0 to 1. In a special case, the above solution set yields an explicit form that corresponds to the geometric mean in the following form:

$$x_j = \prod_{i=1}^{m} w_{ij}^{v_i/v}, \quad j = 1, 2, \ldots, n,$$

where x_j is the relative weight of alternative j for the group.

2. Gargallo et al. (2007) have developed another extension of the group AHP in which a Bayesian estimation procedure has been used to compute the priorities in the group AHP analysis. In this approach, the individual priorities are assumed to have a mixture of normal distributions. Using a hierarchical cluster algorithm, this method identifies opinion subgroups within the group.
3. Ramanathan (2006) *and* Wang and Chin (2009) have extended the group AHP and synthesized it

with the data envelopment analysis in order to address inconsistencies in input matrices in the group. Wang and Chin (2009) have suggested a data envelopment method to compute the best local priorities regardless of the extent of inconsistencies in the pairwise comparison matrices.

4. Dong et al. (2010) have combined the group AHP with the Chiclana et al. (2008)'s consensus framework to develop the AHP group consensus model. They define a consensus index to represent the extent of agreement among individual decision makers' input matrices, and propose two AHP consensus models that could result in an improved consensus index.
5. Monroe-Jimènez et al. (2008) have also proposed a method for the identification of consensus among a group of decision makers. They present the consistency consensus matrix for the purpose of consensus building within the decision group, particularly in large groups, such as democratic voting.
6. Although not an extension of AHP, fuzzy group decision making has emerged as yet another method in handling group decisions (see, for example, Gabriella and Yager 2006 for a review). Li (2010) suggests a fuzzy multi-attribute group decision making in which input information (equivalent to the matrices of pairwise comparison in AHP) are non-homogeneous, in that the preferences for attributes could be expressed in four different formats: qualitatively in linguistic terms, fuzzy numbers, interval values, and real numbers. These formats of preference expression and elicitation are similar to those available in the AHP approach. Li (2010) has proposed a method to transform the non-homogeneous inputs to a fuzzy group decision making formulation. In this approach, decision makers are viewed as attributes. The solution to this approach is called a compromise solution, which maximizes the group utility of the majority and minimizes individual regret for the minority (p. 99).

Social Choice Theory

The group decision problem was of interest as early as the eighteenth century, when Borda studied voting problems in the 1770s, and Marquis de Condorcet noticed the paradoxes and problems of majority rule in the 1780s. One example of such problems is that of three alternatives a, b, and c, where a is preferred to b, b is preferred to c, and c is preferred to a. Attention to methods for making a social choice continued in nineteenth century and greatly intensified in this century.

One way to arrive at a group decision is through voting, which also falls under the heading of social choice theory. Social choice theory investigates the process of arriving at a group decision in democratic societies through the expression of the majority's will. Voting involves selecting an alternative or candidate based multiple criteria. It involves two processes: voting and the aggregation method for determining the winner, that is, voting and counting the vote. There are a number of methods for voting, such as bivalue (yes, no), rating, and ranking the alternatives. Counting could be a simple counting of yes or no votes, averaging the rates, or a more complex aggregation method using ranks.

Social Welfare Function — In the social welfare function, the voting and counting processes are given a formal mathematical structure. Each member has a utility function based on which he or she determined the ordinal ranking of all alternatives. A member's ordering of alternatives is called the preference profile. The social welfare function is a rule for structuring group preference orderings of alternatives from the members' preference profiles. Obviously, there are numerous ways to arrive at group ordering alternatives. Arrow sought to limit the number of possible group orderings of alternatives, which led to his famous impossibility theorem.

Arrow's Impossibility Theorem — Arrow (1951) observed that by imposing rational conditions, one can reduce the number of solutions in the social welfare function. He postulated that:

1. All possible choices are already included in the problem set.
2. If one alternative is dropped such that the preference relations remain unchanged, the group preference will not change.
3. For any given two alternatives, the group can express its preference of one over the other.
4. There is no individual in the group whose preference represents the group preference.

5. Assume the group prefers alternative 1 to 2. If one member's preference for alternative 1 increases without affecting the pairwise preference ordering of other alternatives, the group will continue to prefer alternative 1 to 2.

Arrow's famous impossibility theorem states that there is no social welfare function that satisfies all of the above five properties.

Note that in voting and the social welfare function, the strength of members' preferences is not taken into account, whereas in utility theory, consensus generation methods, and AHP, the strength of members' preferences is incorporated into the method. In utility theory and consensus generation methods, members' preferences are assigned relative weights, relative to their importance in the group decision.

Boiney's Envy-Based Fair Group Decision — Boiney (1995) developed a model for fair choice among limited options under uncertainty when preferences are heterogeneous. The measure of fairness in this model is built on the degree of envy. Pairwise envy is defined as

$$e_{ij}(x) = \max\left(0, u_i\left(x_j\right) - u_i(x_i)\right),$$

Where e_{ij} is i's envy of j, x_i is i's bundle of goods, x_j 's bundel of goods, and u_j is i's utility. Individual i's envy is

$$e_i(x) = \frac{1}{n-1}\sum_j e_{ij}(x),$$

where the constant $(n - 1)$ is used to normalize the extent of envy between 0 (no envy) to 1 (maximum envy). Under certain conditions of linearity, independence, monotonicity, and anonymity, group envy could be computed as the (weighted) average of individual envies, where weights, if used, are the relative importance of individuals. Using this measure of envy, Boiney defined ex post and ex ante measures of unfairness, combining them into an overall measure of fairness F. This measure is, in turn, combined with the group utility function to determine social preference function, which could be used in making a fair and efficient choice among limited options.

Game Theory

Game theory has been developed in the context of decision makers (or players) who are in conflict. Game theory, however, has been extended to include a cooperative n-person game, in which the players cooperate with one another in order to maximize their own gain or payoff. The Nash-Harsanyi and Shapley methods are among the game-theoretic methods of group decision.

Nash-Harsanyi Bargaining Method — Nash (1950, 1953) developed the two-person cooperative game, which was generalized to the n-person cooperative game by Harsanyi (1963). In this model, one can find a unique solution to the n-person cooperative game problem by solving the following:

$$\begin{aligned} &\max_{x_i} \prod_{i=1}^{n} (x_i - d_i) \\ &\text{s.t.} \quad x_i \geq d_i \quad x \in P; D = (d_1, d_2, \ldots, d_n), \end{aligned}$$

where P is the set of payoff vectors, and D is the payoff when disagreement exists.

The above formulation is based on the following assumptions:

1. No payoff is better than the solution of the above formulation.
2. The players' payoffs are the same.
3. The linear transformation of all payoffs does not change the solution to the above problem.
4. Assume there are two games 1 and 2 with the same payoffs for disagreement, and the payoff vector of game 1 is a subset of the payoff vector of game 2. If the solution of game 2 is in the payoff vector of game 1, then it is also the solution to game 1. This assumption ensures that adding nonoptimal payoffs does not change the optimal solution.

Harsanyi (1977) has shown that the above formulation can also be derived from Zeuthen's principle that the player who has the highest risk-aversion toward conflict always makes the next concession.

The Shapley Value — If the utility of the players is transferable — that is, one player can transfer money, goods, or services to another player such that the sum of the two players' utilities remain the same — then the Nash-Harsanyi solution does not hold. This is due to the fact that there would not be a unique payoff vector

for disagreement. Shapley (1953) provides a solution for an n-person cooperative game with a transferable utility function.

A *coalition* is defined as a subgroup of members. The grand coalition consists of all the group members. If a member i joins a coalition, his or her marginal contribution to the coalition C is specified as $V(C) - V(C - i)$. The payoff to member i should be the average marginal contribution of the player to the grand coalition. Assume that the grand coalition is formed by members gradually joining the coalition and the order of members joining the coalition is equally likely, then the payoff of each player, (P_i, $i = 1, 2,\ldots, n$) or the Shapley value, is

$$P_i = \sum_{C \subseteq N} \frac{(c-1)!(n-c)!}{n!} V(C) - V(C - i),$$

where c is the number of players in C, n is the number of players, and N is the set of players.

The above solution is based on the following assumptions:

1. The value of the entire game is the sum of the payoffs to members.
2. All members receive an equal payoff.
3. If a game consists of two subgames, the payoff of the game is the sum of the payoffs of the two subgames.

Computer-Based Group Decision Process

A number of computer-based methods have been developed to facilitate the group decision process in various circumstances. These systems can divided into two groups: intelligent systems and group decision support systems.

Intelligent Systems for Group Decisions — In a sequential group process, there is more than one party involved in a negotiation process that takes place sequentially through time. One can use the artificial intelligence and expert system techniques to facilitate the process. A number of such systems have been developed.

Sycara developed PERSUADER, which simulates the labor-management negotiation process (Sycara 1991). This system uses frame-based knowledge representation and case-based reasoning of artificial intelligence with graph search and multi-attribute utilities to propose problem restructuring for simulated negotiations. The system restructures the problem by (1) introducing new goals, (2) substituting goals, and (3) abandoning goals.

The logical representation of the negotiation process using the framework of mathematical logic is another way to model the group negotiation process. Kersten et al. (1991) showed how one can model negotiation and restructure it for arriving at a negotiated solution.

Group Decision Support Systems — Group decision support systems refer to computer-based systems and methods developed to facilitate group decision making. One category of such systems is the electronic meeting system (EMS), which consists of a collection of hardware, software, audio and video equipments, and group procedures to create a supportive environment for the group decision process (Dennis et al. 1988).

These systems are designed for various purposes:

- Generating group options and brainstorming.
- Supporting and improving communication among the group members.
- Increasing participation.
- Providing computational and procedural support for the group process.

There are different conclusions regarding the existence and extent of positive contributions of such systems (Jessup and Valacich 1993; Fjermestad and Hiltz 1999). However, group decision support systems have gained increasing acceptance within industry as tools that increase the efficiency of group decision processes.

Intelligent Agents — The use of group decision methods has found an unexpected application in creating multiagent intelligent systems. Although there is no consensus regarding the definition of software agents, one can describe it as components that take actions on behalf of users (ACM 1999). Capturing users' preferences is one of the important aspect of software agents developed for electronic commerce (Maes et al. 1999). If multiagents act on behalf of one entity and need to come to an agreement regarding an action, then they need to apply group decision methods for arriving at a consensus.

Issues in Decision Making in Interactive Groups

The use of computer-based group decision making has generated interest in factors that may impact the outcome and quality of group decisions. These factors include group facilitation, anonymity of group members, group size, nature of decision, decision process, and groupthink (Bostrom et al. 1993; El-Shinnawy and Vinze 1998; Esser 1998; Fjermestad and Hiltz 1999).

An important property of the group decision is the number of new ideas and information initiations during the course of decision making. Silver (1995) posited that the number of ideas and negative evaluation of them are the most important determinant factors of group decisions. He formulated a two-stage optimization heuristics for the dual motives of group members: maintaining their relative status in the group and contributing to group objectives. At the first stage, the members maintain their status by contributing ideas that minimize the probability of receiving negative evaluations. At the second stage, members contribute ideas that incrementally increase the probability of receiving negative evaluations in proportion to their relative status, in order to contribute to the group decision quality.

See

- ▶ Analytic Hierarchy Process
- ▶ Decision Analysis
- ▶ Delphi Method
- ▶ Electronic Commerce
- ▶ Game Theory
- ▶ Group Decision Computer Technology
- ▶ Markov Processes
- ▶ Multi-attribute Utility Theory
- ▶ Multiple Criteria Decision Making
- ▶ Total Quality Management

References

ACM. (1999). Multiagent systems on the Net and agents in E-commerce. *Special Issue of Communications of the ACM, 42*(3), 39–114.

Aczel, J., & Saaty, T. L. (1983). Procedures for synthesizing rational judgements. *Journal of Mathematical Psychology, 27*, 93–102.

Arrow, K. J. (1951). Social choice and individual values. *Cowles Commission Monograph* 12. New York: Wiley.

Bodily, S. E. (1979). A delegation process for combining individual utility function. *Management Science, 25*, 1035–1041.

Boiney, L. G. (1995). When efficient is inefficient: Fairness in decisions affecting a group. *Management Science, 41*, 1523–1537.

Bostrom, R. P., Anson, R., & Clawson, V. K. (1993). Group facilitation and group support systems. In L. M. Jessup & J. S. Valacich (Eds.), *Group support systems: New perspectives* (pp. 146–168). New York: Macmillan.

Brock, H. W. (1980). The problem of utility weights in group preference aggregation. *Operations Research, 28*, 176–187.

Chiclana, F., Herrara, F., Herrara-Viedma, E., & Alonso, S. (2008). Integration of a consistency control module within a consensus decision making model. *International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems, 16*, 35–53.

Dalkey, N. C. (1967). Delphi, Rand Corporation.

Delbecq, A. L., & Van de Ven, A. H. (1971). A group process model for problem identification and program planning. *Journal of Applied Behavior Sciences, 7*, 466–492.

Dennis, A. R., George, J. F., Jessup, L. M., Nunamaker, J. F., Jr., & Vogel, D. R. (1988). Information technology to support electronic meetings. *MIS Quarterly, 12*, 591–624.

Dong, Y., Zhang, G., Hong, W.-C., & Xu, Y. (2010). Consensus models for AHP group decision making under Row geometric mean prioritization method. *Decision Support Systems, 49*, 281–289.

El-Shinnawy, M., & Vinze, A. S. (1998). Polarization and persuasive argumentation: A study of decision making in group meetings. *MIS Quarterly, 22*, 165–193.

Esser, J. K. (1998). Alive and well after 25 years: A review of groupthink research. *Organizational Behavior and Human Decision Processes, 73*, 116–141.

Fjermestad, J., & Hiltz, S. R. (1999). An assessment of group support systems experiment research: Methodology and results. *Journal of Management Information Systems, 15*(3), 7–149.

Gabriella, P., & Yager, R. R. (2006). Modeling the concept of majority opinion in group decision making. *The Information of the Science, 176*, 390–414.

Gargallo, P., Jimènez-Moreno, J. M., & Salvador, M. (2007). AHP group decision making: A Bayesian approach based on mixtures for group pattern identification. *Group Decision and Negotiation, 16*, 485–506.

Gass, S. I., & Rapcsák, T. (1998). A note on synthesizing group decisions. *Decision Support Systems, 22*, 59–63.

Harsanyi, J. C. (1963). A simplified bargaining model for the n-person cooperative game. *International Economic Review, 4*, 194–220.

Harsanyi, J. C. (1977). *Rational behavior and bargaining equilibrium in games and social situations*. England: Cambridge University Press.

Jessup, L. M., & Valacich, J. S. (1993). *Group support systems: New perspectives*. New York: Macmillan.

Keeney, R. L., & Kirkwood, C. W. (1975). Group decision making using cardinal social welfare functions. *Management Science, 22*, 430–437.

Kersten, G., Michalowski, W., Szpakowicz, S., & Koperczak, Z. (1991). Restructurable representations of negotiation. *Management Science, 37*, 1269–1290.
Krzysztofowicz, R. (1979). *Group utility assessment through a nominal-interacting process*. Unpublished working paper, Department of Civil Engineering, MIT, Cambridge, MA.
Li, D.-F. (2010). A new methodology for fuzzy multi-attribute group decision making with multi-granularity and non-homogeneous information. *Fuzzy Optimal Decision Making, 9*, 83–103.
Maes, P., Guttman, R. H., & Moukas, A. G. (1999). Agents that buy and sell. *Communications of the ACM, 42*(3), 81–91.
Monroe-Jimènez, J. M., Aguaròn, J., & Escobar, M. T. (2008). The core of consistency in AHP group decision making. *Group Decisions and Negotiation, 17*, 249–265.
Nash, J. (1950). The bargaining problem. *Econometrica, 18*, 155–162.
Nash, J. (1953). Two-person cooperative games. *Econometrica, 21*, 128.
Ortolano, L. (1974). A process for federal water planning at the field level. *Water Resources Bulletin, 10*(4), 776–778.
Ramanathan, R. (2006). Data envelopment analysis for weight derivation and aggregation in the analytic hierarchy process. *Computers and Operations Research, 33*, 1280–1307.
Saaty, T. L. (1977). A scaling method for priorities in hierarchical process. *Journal of Mathematical Psychology, 15*, 234–281.
Shapley, L. S. (1953). A value for n-person games. In Kuhn, H. W., & Tucker, A. W. (Eds.), *Contributions to the theory of games* (pp. 307–317). Princeton University Press.
Silver, S. D. (1995). A dual-motive heuristic for member information initiation in group decision making: Managing risk and commitment. *Decision Support Systems, 15*, 83–97.
Sycara, K. P. (1991). Problem restructuring in negotiation. *Management Science, 37*, 1248–1268.
Wang, Y.-M., & Chin, K.-S. (2009). A New data envelopment analysis method for priority determination and group decision making in the analytic hierarchy process. *European Journal of Operations Research, 195*, 239–250.
Zahedi, F. (1986a). Group consensus function estimation when preferences are uncertain. *Operations Research, 34*, 883–894.
Zahedi, F. (1986b). The analytic hierarchy process — a survey of the method and its applications. *Interfaces, 16*(4), 96–108.

GUB

▶ Generalized Upper-Bounded (GUB) Problem

GUI

Graphical user interface. Means by which users interact with electronic devices via images, e.g., computers using pull-down or touch-screen menus, in contrast with typed text commands.

See

▶ Computer Science and Operations Research Interfaces
▶ Visualization
▶ WIMP

H

Half Space

► Linear Inequality

Hamiltonian Tour

In an undirected connected graph, a Hamiltonian tour is a sequence of edges that passes through each node of the graph exactly once.

See

► Graph Theory
► Traveling Salesman Problem

Hamilton-Jacobi-Bellman Equation

Condition specifying a partial differential equation that the (optimal) value function must satisfy in an optimal control problem, analogous to the Bellman optimality equation in dynamic programming.

See

► Bellman Optimality Equation
► Dynamic Programming
► Optimal Control

Hazard Rate

► Distribution Selection for Stochastic Modeling
► Failure-Rate Function
► Reliability of Stochastic Systems

Health Care Management

Yasar A. Ozcan
Virginia Commonwealth University, Richmond, VA, USA

Introduction

The techniques of operations research have found their way into health care management, not just in the logistical and managerial support of clinical services, but in the central decision processes of disease screening, diagnosis and therapy, and in medical education. Hundreds of citations to operations research and its associated analytical techniques are to be found in the medical literature and are accessible in the U.S. National Library of Medicine's on-line MEDLARS system. The early applications spawned new professional organizations and journals, now thriving in the medical arena. Many operations research applications are indexed to the near synonymous term, Medical Informatics, the application of computers and information technology to the broad field of health care. Operations research techniques that most frequently applied to solve managerial issues in health care organizations can be

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

found in health care management text books (Ozcan 2009; Shiver and Eitel 2009). To understand the position of operations research in the medical literature, a simple rule helps: medical informatics relates to medicine and health care as operations research relates to the work of business and industry. Both place a heavy emphasis on exploitation of the potentials of computer and information technologies. In addition to these, in conjunction with bioengineering and medical physics, more recent advances in computational biology and medical applications had used OR in treatment design and in genomics. These applications range from optimizing the dose and location of radiation treatments to optimization and simulation models to identify correct diagnosis and treatment for genetic variants of some diseases (Greenberg et al. 2009); similarly, using optimization and extensive classification systems to predict immunity to a vaccine without exposing individuals to infection (Lee 2010); abnormal brain activity (Chaovalitwongse et al. 2008). In short, to address contemporary issues in health care, diverse disciplines including electrical engineering, biomedical engineering, industrial engineering, and medicine attempt to bridge a vital gap between operations research and medical research with the help from data mining, signal processing techniques can be used to tackle the most challenging problems in modern medicine (Chaovalitwongse et al. 2010). Major traditional operations research methods that predominate in health care applications include stochastic models, computer simulation, mathematical programming, and decision analysis.

Simulation and Stochastic Models

Computer simulation plays an important role in teaching, research, and development of medical practice, expanding beyond its early role as a mimic of complex stochastic processes. A review of over hundred documents indexed both to simulation and medicine reveals a range of applications from the traditional Monte Carlo representation of physiological processes to three-dimensional imaging. An example of use of computer simulation to develop a protocol for burn care is given by Roa and Gomez-Cia (1994). Many applications are devoted to medical education by simulation of clinical problems, either through random sequences of events in diagnosis and therapy or the responsive behavior of images or a manikin, a move toward virtual reality. Examples of simulation as an instructional aid in a clinical setting are: in cardiac care (Lipner et al. 2010; Kobayashi et al. 2010), prostate care (Wang et al. 2010), training and teaching in obstetrics and gynecology (Dayal et al. 2009), training in anesthesiology (Waisel et al. 2009).

Serving as a decision support tool, computer simulation offers management detailed information about the processes of health care, enabling management to make better decisions through performance data on a variety of issues of in terest before any change was introduced within the process. Thus, use of computer simulation makes information better and faster without requiring intensive financial and resources, especially when there are a number of alternatives under consideration. Computer simulation has found its application particularly in both estimating and forecasting of several issues involved in hospital industry, with the primary objective of guiding hospital management and policy makers about evaluating alternatives involved in allocation of scarce resources or anticipating results from certain changes that can be made to solve the problems identified. Some examples of simulation and stochastic programming as a management aid in a hospital setting:

- Application of stochastic programming and simulation of nurse assignments (Punnakitikashem et al. 2008; Sundaramoorthi et al. 2010);
- Analysing management policies for operating room planning using simulation (Persson and Persson 2010);
- Reducing patient wait times and improving resource utilization using simulation at ambulatory cancer care units (Santibáñez et al. 2009);
- Redisign of hospital based pharmacy delivery processes using simulation and oprimization (Augusto and Xie 2009);
- Improving patient flow at outpatient settings (Chand et al. 2009);
- Simulation of strategies for cervical cancer screening (McLay et al. 2010).

The field of epidemiology, lying within the domain of medicine, has been attractive to stochastic model building and simulation. Understanding the origin, spread, and decline of epidemics is essential to recognition of causal agents and vectors of transmittal and the development and evaluation of prevention

measures. The complexity of epidemics and interventions to control them often defies a purely mathematical analysis. In early studies of mathematical epidemiology, Bailey (1967) used computer simulation to predict both temporal and spatial progress of epidemics, expressed in stochastic models. Examples of stochastic models and simulation are found in epidemic models of HIV/AIDS (Rossi 1999; Rauner 2002). The problem of allocation of resources for this disease is addressed by Brandeau et al. (2005).

Mathematical Programming

Increasing health care costs, in particular hospital costs, have provided incentive for health care researchers to analyze performance of hospitals through mathematical programming techniques. Growing interest in performance has led to increasing applications of various efficiency measurement techniques, as a means of evaluating performance. Proposed methods to measure efficiency can be classified into two broad categories: parametric and non-parametric methods. The former assumes a particular functional form, for example, Cobb-Douglas method, while the latter does not assume a functional form, for example, Data Envelopment Analysis (DEA) method.

DEA is a mathematical programming tool that is designed to evaluate how efficiently an organizational entity, called the decision-making unit or DMU, produces a mixture of outputs with an available mixture of inputs. The power of DEA methodology as a multifactor productivity tool in contrast to unidimensional ratio analysis and central-tendency-based regression analysis has been demonstrated in the health care literature (Sherman 1984; Sexton 1986; Huang 1990; Sherman and Zhu 2006, chapter 5; Ozcan 2008). In particular, DEA is recognized as a superior method to identify the sources and amounts of inefficiency in the use of inputs. Pioneering uses of DEA by Charnes et al. (1978, 1981) were followed in the operations research field by many applications across service and program-oriented industries. In health care, Sherman (1984) applied DEA to hospital multifactor productivity studies. Some studies in the health care organization or program efficiency arena followed the Sherman study for benchmarking hospitals (Clement et al. 2008; Nayar and Ozcan 2008; Sikka et al. 2009; Kazley and Ozcan 2009; Ozcan and Luke 2010; Sahin et al. 2010; Ozcan et al. 2010; Lobo et al. 2010), nursing units (Mark et al. 2009), ambulatory surgery centers (Iyengar and Ozcan 2009), and nursing homes (Ferrier and Valdmanis 1996; Björkgren et al. 2001; Knox et al. 2003).

Decision Analysis in Medical Decision Making

The notion of tradeoff among benefits and losses carries through to decisions made under certainty in screening and clinical diagnosis, where the costs of missing a case — a false negative — are balanced against the costs of interpreting as present a condition not there — a false positive. The expression in the normal form of search for a Bayesian solution, that is, as a minimization of expected loss, bears strong resemblance to linear programming, but the evolution of medical decision theories has yielded a number of new approaches. Nearly parallel in time to the early applications of operations research to the logistical and organizational problems of health services, examples of the techniques appeared directly in the procedures of diagnosis and therapy. The availability of large databases linking patient signs, symptoms, and other descriptors to disease states has led applications of statistical analysis and value theory to diagnostic processes and choice of therapy. A review by Barnoon and Wolfe (1972) cited early work on the logistical foundations of diagnosis (Ledley and Lusted 1959) and on disease screening (Flagle 1967), demonstrating that the optimal screening level of a test is a specific function of prevalence of undetected disease and their relevant costs, or regrets, of false negative and false positive determinations. The applications of linear programming, pattern recognition and decision support systems for breast cancer diagnosis are described in Mangasarian et al. (1990), Wolberg and Mangasarian (1993), and Mangasarian et al. (1995).

Improvement in screening through increases in sensitivity and specificity of test still remains an objective, and is aided by multivariate analyses made possible by large databases and clinical trials. Emergence of the term, "Computer-Aided Diagnosis," has accompanied many efforts to sharpen the statistical relationship between symptoms and disease (Chan et al. 1999; Nawano et al. 1999; Lowe and Harrison 1999). Turning attention to treatment,

knowledge gained from statistical analysis and systematic compilation of outcomes has led beyond estimation of diagnostic probabilities to a large enterprise in expert systems, in which the process of successful therapy are expressed in algorithmic form built around a patient database interfaced to a knowledge base (Wang et al. 1998). Beyond expert systems to diffuse and implement protocols, efforts have been made to understand and emulate the decision process itself — an approach to artificial intelligence (AI). Early development in AI related to specific diseases is reviewed by Szolovits and Pauker (1978). There are also applications of AI processes in medical care using neural networks and applying them to computer-aided decision processes (Chaudhry 2008; Eken et al. 2009).

Concluding Remarks

Two major patterns are discernible in the evolution of medical practice aided by growing databases, improving analytic techniques, and new communication technologies. First is the formalization of decision processes in the multi-disciplinary protocols or practice guidelines based on outcomes of research and technology assessment. The major challenge in developing guidelines is that of creating flexible guidelines that are applicable to broad patient populations and practice settings that differ substantially (Dawson, 1997). Therefore, development of practice guidelines is an important area of interest for physicians and operations researchers.

The format of guidelines, which often contain a prescriptive algorithm familiar to operations researchers, also quite frequently contain a version for patients. This marks the enlightened involvement of patients in decisions about choice of therapeutic strategies, such as optimal and personalized treatment design thanks to new biological findings and imaging technologies (Lee 2010). Other developments include the incorporation of medical decision processes in chronic disease management, as well as optimizing the capability and efficiency of the delivery systems through supply–demand alignment while reducing variability in delivery (Carter 2002; Greenberg et al. 2009; Lee 2010). A collaboration of a physician and OR analyst has remained, while the concepts have become internalized in the medical decision process.

See

- ▶ Artificial Intelligence
- ▶ Data Envelopment Analysis
- ▶ Decision Analysis
- ▶ Decision Support Systems (DSS)
- ▶ Emergency Services
- ▶ Expert Systems
- ▶ Health Care Strategic Decision Making
- ▶ Hospitals
- ▶ Information Systems and Database Design in OR/MS
- ▶ Linear Programming
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Augusto, V., & Xie, X. (2009). Redesigning pharmacy delivery processes of a health care complex. *Health Care Management Science, 12*(2), 166–178.

Bailey, N. T. J. (1967). *The mathematical approach to biology and medicine*. London: John Wiley.

Barnoon, S., & Wolfe, H. (1972). *Measuring effectiveness of medical decisions: An operations research approach*. Springfield, IL: Clarke C Thomas.

Björkgren, M. A., Häkkinen, U., & Linna, M. (2001). Measuring efficiency of long-term care units in Finland. *Health Care Management Science, 4*(3), 193–200.

Brandeau, M. L., Zaric, G. S., & De Angelis, V. (2005). Improved allocation of HIV prevention resources: Using information about prevention program production functions. *Health Care Management Science, 8*(1), 19–28.

Carter, M. (2002). Health care management. Diagnosis: Mismanagement of resources. *OR/MS Today, 29*(2), 26–32.

Chan, H. P., et al. (1999). Improvement of radiologists' characterization of mammographic masses by using computer-aided diagnosis: An ROC study. *Radiology, 212*, 817–827.

Chand, S., Moskowitz, H., Norris, J. B., Shade, S., & Willis, D. R. (2009). Improving patient flow at an outpatient clinic: Study of sources of variability and improvement factors. *Health Care Management Science, 12*(3), 325–342.

Chaovalitwongse, W., Fan, Y. J., & Sachdeo, R. C. (2008). Novel optimization models for abnormal brain activity classification. *Operations Research, 56*(6), 1450–1460.

Chaovalitwongse, W., Pardalos, P. M., & Xanthopoulos, P. (Eds.). (2010). *Computational neuroscience* (Springer optimization and its applications, Vol. 38). New York: Springer.

Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operational Research, 2*, 429–444.

Charnes, A., Cooper, W. W., & Rhodes, E. (1981). Evaluating program and managerial efficiency: An application of data envelopment analysis to program follow through. *Management Science, 27*, 668–697.

Chaudhry, B. (2008). Computerized clinical decision support: Will it transform healthcare? *Journal of General Internal Medicine, 23*(Supplement 1), 85–87.

Clement, J. P., Valdmanis, V. G., Bazzoli, G. J., Zhao, M., & Chukmaitov, A. (2008). Is more better? An analysis of hospital outcomes and efficiency with a DEA model of output congestion. *Health Care Management Science, 11*(1), 66–77.

Dawson, N. V. (1997). Physician judgements of uncertainty. In G. B. Chapman & F. A. Sonneberg (Eds.), *Theory, Psychology, and Application*, Cambridge, UK: Cambridge University Press.

Dayal, A. K., Fisher, N., Magrane, D., Goffman, D., Bernstein, P. S., & Katz, N. T. (2009). Simulation training improves medical students' learning experiences when performing real vaginal deliveries. *Simulation in Healthcare: The Journal of the Society for Simulation in Healthcare., 4*(3), 155–159.

Eken, C., Bilge, U., Kartal, M., & Eray, O. (2009). Artificial neural network, genetic algorithm, and logistic regression applications for predicting renal colic in emergency settings. *International Journal of Emergency Medicine, 2*(2), 99–105.

Ferrier, G. D., & Valdmanis, V. (1996). Rural hospital performance and its correlates. *Journal of Productivity Analysis, 7*(1), 63–80.

Flagle, C. D. (1967). *A decision theoretical comparison of three procedures of screening for single disease. Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Berkeley: University of California Press.

Greenberg, H. J., Holder, A. G., Leung, M.-Y., & Schwartz, R. (2009). Computational biology and medical applications. *OR/MS Today, 36*(3), 34–39.

Huang, Y.-G. L. (1990). An application of data envelopment analysis: Measuring the relative performance of Florida general hospitals. *Journal of Medical Systems, 14*(4), 191–196.

Iyengar, R. I., & Ozcan, Y. A. (2009). Performance evaluation of ambulatory surgery centers: An efficiency approach. *Health Services Management Research, 22*(4), 184–190.

Kazley, A. S., & Ozcan, Y. A. (2009). Electronic medical record (EMR) use and efficiency: A data envelopment analysis of hospitals. *Socio-Economic Planning Sciences, 43*(3), 209–216.

Knox, K. J., Blankmeyer, E. C., & Stutzman, J. R. (2003). Organizational efficiency and quality in Texas nursing facilities. *Health Care Management Science, 6*(3), 175–188.

Kobayashi, L., Dunbar-Viveiros, J. A., Sheahan, B. A., Rezendes, M. H., Devine, J., Cooper, M. R., et al. (2010). In situ simulation comparing in-hospital first responder sudden cardiac arrest resuscitation using semiautomated defibrillators and automated external defibrillators. *Simulation in Healthcare: The Journal of the Society for Simulation in Healthcare, 5*(2), 82–90.

Ledley, R. S., & Lusted, L. B. (1959). Reasoning foundation of medical diagnosis. *Science, 130*, 9–29.

Lee, E. (2010). Advancing health care on multiple fronts. *OR-MS Today, 37*(3), 20–29.

Lipner, R. S., Messenger, J. C., Kangilaski, R., Baim, D. S., Holmes, D. R., Williams, D. O., et al. (2010). A technical and cognitive skills evaluation of performance in interventional cardiology procedures using medical simulation. Simulation in healthcare. *The Journal of the Society for Simulation in Healthcare, 5*(2), 65–74.

Lobo, M. S. C., Ozcan, Y. A., Silva, A. C. M., Lins, M. P. E., & Fiszman, R. (2010). Financing reform and productivity change in Brazilian teaching hospitals: Malmquist approach. *Central European Journal of Operations Research, 18*(2), 141–152.

Lowe, A., & Harrison, M. J. (1999). Computer-enhanced diagnosis of malignant hyperpyrexia. *Anaesthesia and Intensive Care, 27*(1), 41–44.

Mangasarian, O. L., Street, W. N., & Wolberg, W. H. (1995). Breast cancer diagnosis and prognosis via linear programming. *Operations Research, 43*, 570–577.

Mangasarian, O. L., et al. (1990). Pattern recognition via linear programming: Theory and application to medical diagnosis. In T. F. Coleman & L. Yuying (Eds.), *Large-scale numerical optimization* (pp. 22–30). Philadelphia: SIAM.

Mark, B. A., Jones, C. B., Lindley, L., & Ozcan, Y. A. (2009). An examination of technical efficiency, quality and patient safety on acute care nursing units. *Policy, Politics & Nursing Practice, 10*(3), 180–186.

McLay, L. A., Foufoulides, C., & Merrick, J. R. W. (2010). Using simulation-optimization to construct screening strategies for cervical cancer. *Health Care Management Science, 13*(4). doi: 10.1007/s10729-010-9131-x.

Nawano, S., et al. (1999). Computer-aided diagnosis in full digital mammography. *Investigative Radiology, 34*(4), 310–316.

Nayar, P., & Ozcan, Y. A. (2008). Data envelopment analysis comparison of hospital efficiency and quality. *Journal of Medical Systems, 32*(3), 193–199.

Ozcan, Y. A. (2008). *Health care benchmarking and performance evaluation: An assessment using data envelopment analysis (DEA)*. New York: Springer.

Ozcan, Y. A. (2009). *Quantitative methods in health care management: Techniques and applications* (2nd ed.). San Francisco: Jossey-Bass/Wiley.

Ozcan, Y. A., Lins, M. E., de Castro Lobo, M. S., da Silva, A. C. M., Fiszman, R., & Pereira, B. B. (2010). Evaluating the performance of Brazilian university hospitals. *Annals of Operations Research, 178*(1), 247–261.

Ozcan, Y. A., & Luke, R. D. (2010). Healthcare delivery restructuring and productivity change: Assessing the Veterans Integrated Service Networks (VISNs) using Malmquist approach. Medical Care Research and Review (in press). doi: 10.1177/1077558710369912.

Persson, M. J., & Persson, J. A. (2010). Analysing management policies for operating room planning using simulation. *Health Care Management Science, 13*(2), 182–191.

Punnakitikashem, P., Rosenberger, J. M., & Buckley-Behan, D. F. (2008). Stochastic programming for nurse assignment. *Computational Optimization and Applications, 40*(3), 321–349.

Rauner, M. S. (2002). Using simulation for AIDS policy modeling: Benefits for HIV/AIDS prevention policy makers in Vienna, Austria. *Health Care Management Science, 5*(2), 121–134.

Roa, L., & Gomez-Cia, T. (1994). *A burn patient resuscitation therapy designed by computer simulation. Yearbook of medical informatics*. Stuttgart: Schattauer Verlagsgesellchaft.

Rossi, C. (1999). Estimating the prevalence of injecting drug users on the basis of Markov models of the HIV/AIDS epidemic: Applications to Italian data. *Health Care Management Science, 2*(3), 173–179.

Sahin, I., Ozcan, Y. A., & Ozgen, H. (2010). Assessment of hospital efficiency under health transformation program in turkey. *Central European Journal of Operations Research*. doi: 10.1007/s10100-009-0121-3.

Santibáñez, P., Chow, V. S., French, J., Puterman, M. L., & Tyldesley, S. (2009). Reducing patient wait times and improving resource utilization at British Columbia Cancer Agency's ambulatory care unit through simulation. *Health Care Management Science, 12*(4), 392–407.

Sexton, T. R. (1986). The methodology of data envelopment analysis. In R. H. Silkman (Ed.), *Measuring efficiency: An analysis of data envelopment analysis*. San Francisco: Jossey-Bass.

Sherman, D. H. (1984). Hospital efficiency measurement and evaluation. *Medical Care, 22*(10), 922–928.

Sherman, D. H., & Zhu, J. (2006). *Service productivity management: Improving service performance using data envelopment analysis*. New York: Springer.

Shiver, J. M., & Eitel, D. (2009). *Optimizing emergency department trhroughput: Operation management solutions for health care decision makers*. New York: Taylor & Francis.

Sikka, V., Luke, R. D., & Ozcan, Y. A. (2009). The efficiency of hospital-based clusters. *Health Care Management Review, 34*(3), 251–261.

Sundaramoorthi, D., Chen, V. C. P., Rosenberger, J. M., Kim, S. B., & Buckley-Behan, D. F. (2010). A data-integrated simulation-based optimization for assigning nurses to patient admissions. *Health Care Management Science, 13*(3), 210–221.

Szolovits, P., & Pauker, S. C. (1978). Categorical and probabilistic reasoning in medical diagnosis. *Artificial Intelligence, 11*, 115–144.

Waisel, D. B., Simon, R., Truog, R. D., Baboolal, H., & Raemer, D. B. (2009). Anesthesiologist management of perioperative do-not-resuscitate orders: A simulation-based experiment. *Simulation in Healthcare: The Journal of the Society for Simulation in Healthcare., 4*(2), 70–76.

Wang, N., Gerling, G. J., Krupski, T. L., Childress, R. M., & Martin, M. L. (2010). Using a prostate exam simulator to decipher palpation techniques that facilitate the detection of abnormalities near clinical limits. *Simulation in Healthcare: The Journal of the Society for Simulation in Healthcare., 5*(3), 152–160.

Wang, S., et al. (1998). TACHY: An expert system for the management of supra ventricular tachycardia in the elderly. *American Heart Journal, 1359*(1), 82–87.

Wolberg, W. H., & Mangasarian, O. L. (1993). Computer-designed expert systems for breast cytology diagnosis. *Analytical and Quantitative Cytology and Histology, 15*, 67–74.

Health Care Strategic Decision Making

Roice D. Luke and Yasar A. Ozcan
Virginia Commonwealth University, Richmond, VA, USA

Introduction

The joining of operations research/management science (OR/MS) and strategy may, at first, seem a bit incongruous, as they often address issues at very different levels within organizations. OR/MS typically deals with the internal workings and performance of organizations, strategy with external market conduct and consequent gains (or losses) in competitive advantage. The presumed bright line dividing operations from strategic decision making, however, does not always hold. Indeed, the field of strategy has long recognized the role internal systems management can play in driving advantage.

Internal considerations have always been part of strategy analysis. It is notable that the rather well-known analytical framework of strengths, weaknesses, opportunities, and threats (SWOT) addresses internal organizational strengths and weaknesses, as well as external opportunities and threats (Andrews 1971). And, the so-called resource-based view (Barney and Clark 2007), takes a decidedly internal perspective in assessing strategy by suggesting that competitive advantage is dependent on an organization acquiring (and sustaining the distinctiveness of) superior resources and capabilities.

The predominant view in the field of strategy, the market structural perspective, focuses mostly on external concerns within markets, most particularly on two sources of advantage: distinctive positioning relative to consumer preferences and the buildup of market power as might be accomplished through acquisitions and mergers and/or integration of existing business units. But, internal considerations emerge even within this perspective. For instance, Michael Porter – the primary proponent of the market structural perspective – has highlighted the need for organizations to align internal activities, which collectively comprise their so-called value chains (Porter 1985).

Thus, while OR/MS tends to focus on the internal and strategy analysis on the external, these two

perspectives can be and often are overlapping and complimentary. It follows that the key distinction is not internal versus external, but the degree to which solving an organizational problem is important strategically, that is, the degree to which it may contribute to an organization's market advantage. For example, a decision about staffing may be operational or strategic, depending on such factors as the resource scarcity, impact on the bottom line (and, on gains/losses in competitive advantage), disruptiveness of the decision outcome to an organization, and so on. Hospitals commonly fight over physicians who specialize in clinical areas that are of critical importance to them, including, in particular, physicians in the areas of cardiac surgery, orthopedics, obstetrics, oncology, and a number of other areas. Thus, the characterization of those decisions, as well as the management level at which they are made, is likely determined by such factors as complexity, uncertainty, and impact.

This article argues that internal decisions have recently become much more important strategically than was the case in the not too distant past. This is especially true for acute care, a sector to which OR/MS techniques have been extensively applied in addressing a great diversity of management problems. The increased relevance of such applications, however, is attributable to the significant restructuring that has occurred over the last several decades, the result of which was to produce a great many complex, highly intra-dependent provider clusters that dominate markets nationally. Within just a few years, health care delivery systems in the United States (and in the many other countries around the world) grew significantly in scale, product scope, and inter-organizational complexity; and, importantly, this occurred mostly at local and regional levels (Luke 2010). The growth in common ownership, combined with high spatial proximities. introduced very important inter-organizational interdependencies that have increased exponentially the need for multi-unit coordination and rationalization and, consequently, the need for the combined application of system optimization and OR/MS techniques.

Regionalized health care systems must rationalize costly resources and capabilities across system members, a task that heretofore could only be accomplished by informal coordination or as the product of government regulatory effort, neither of which, over the years, did much improve inter-organizational coordination (Bice 1984). The complex organizations that have formed, in other words, have become true laboratories within which OR/MS and strategy analysis can be joined to transform system spatial coordination into competitive advantage.

This article shows that the wave of mergers and acquisitions, especially those that occurred in the 1990s, produced a large number of provider clusters in this country, which, as a result, has created important opportunities for OR/MS to address truly strategic issues affecting health care organizations. These clusters combine hospitals and other providers within local markets, which clusters now must make a large number of cluster-level operational decisions, many of which could have great impact on competitive strategy.

What follows is divided into three sections. The first summarizes why clusters have formed, thereby providing the basic rationale for why application of OR/MS tools and techniques to these systems is needed. The second describes patterns of cluster formation across the country for the purpose of identifying the configurational diversity that complicates the application OR/MS tools to solving cluster problems. And, the third concludes by summarizing strategically relevant organizational issues that arise when one addresses system problems within local and regional health systems.

Clusters: An Emerging Concept in Health Care

The idea that geographic clustering could lead to improved performance is neither new in health care nor is it limited to the health care industry. Economists have long argued that geographic proximities and associated inter-organizational coordination can have a major impact on organizational and competitive performance. For example, Marshall (1920), focused on the economics of geographical proximities that could lead competitors to locate near one another. His work on industrial districts in England created the theoretical foundations upon which a number of fields, including economic geography, location theory, and the study of clustering have been built.

In the early 1990s, Michael Porter (1996), the leading expert in the field of strategy, argued that

spatial clustering should become a central focus of scholarly investigation, not only for the study of individual companies, but also of whole industries and even nations. All of these, he reasoned, could experience significant gains in competitive advantage were they more actively to promote the formation of local clusters. As he pointed out, geographic proximities "amplify many of the productivity and innovation benefits" attributable to multi-organizational production, largely by reducing transaction costs, improving communication flows, and increasing opportunities to share and innovate as collectives (Porter 1996, p. 222).

Porter identified a number of specific economies that interdependent organizations derive from shared geographic proximities. These included: productivity improvements (e.g., due to shared access to specialized inputs and employees), sharing in technical and market information, improved coordination with complementary businesses and support institutions, innovations through cross-company and cross-industry collaboration, new business formation, collaboration in quality improvement, shared technology. Porter even suggested that the study of geographic clusters could produce greater understanding of the dynamics of production and competition than would be possible were one to focus exclusively on individual firm or industry-level behaviors. This is because clusters often represent natural and relatively complete production arrangements, within which the full range of vertical and horizontal interactions among organizations occur.

Porter also identified two structural dimensions that could be key to improving cluster performance: configuration (how and where cluster production activities are distributed in geographic space) and coordination (activities that integrate production and management systems across cluster members). Note that these two dimensions are equivalent to Lawrence and Lorsch's (1967) concepts of differentiation (equivalent to configuration) and integration (equivalent to coordination). The difference is that Porter applied these to organizations located within the same general geographic space, whereas Lawrence and Lorsch did not consider geography in their analyses. Porter's argument is that geographic proximities greatly enhance the potential for exchanging and centralizing functions, facilitating communications across clustered members, and resolving inter-organizational conflicts and the need for compromise. Both Porter (and Lawrence and Lorsch) recognized that configuration (or inter-organizational differentiation) comes first, after which coordination is needed to ensure that the interdependent, spatially proximate and increasingly specialized organizational units collectively achieve a unity of effort.

From the perspective of this article, it should be clear that OR/MS applications can contribute to both configuration and coordination, by enabling better analysis and planning for optimal configurations and designing systems that help multi-organizational configurations become integrated/holistic systems. The health care clusters are described below and some of the configurational diversity that distinguishes them are illustrated. Then some specific issues that uniquely arise when OR/MS techniques are applied to the clusters are identified. While coordinative arrangements per se are not addressed here, it is suggested that these techniques apply to both the configurational and coordinative problems health care clusters face.

The idea that clusters might improve performance is also not new in health care. Fox (1986) argued that the regional model of delivery organization has been at the core of public policy in the U.S. and England for nearly a century. In the early decades of the last century, the industry focused on fragmentation, a problem that plagued this industry from the beginning of modern medicine (Starr 1982; Stevens 1989). In the first third of the last century, the Committee on the Costs for Medical Care produced a number of reports recommending major reforms in health care, many of which focused on ways by which coordination between physicians and hospitals could be improved (Falk 1958). In the 1940s and through the 1970s the focus shifted to a concern with costs and duplicated capacity, especially within the increasingly complex and costly acute care sector (Stevens 1989). In this period, a series of federal and state-sponsored planning efforts – from the Hill Burton Act in 1946 to the National Health Planning and Resources Development Act of 1974 – addressed primarily the rationalization of capacity locally and regionally. These efforts, however, had little impact on system structures, let alone on patterns of coordination, for many reasons, but mostly because the industry successfully defended historic expectations for professional and organizational autonomy (Bice 1984).

With passage of Medicare and Medicaid in 1965, the policy focus began to shift away from system strategies and coordination toward the use of regulation and incentives to control rapidly rising costs. Inter-organizational coordination reemerged in the Health Maintenance Organization Act of 1973, but as it turned out, this produced little change at the level of health services delivery. Twenty years later, in the context of rising pressures for health care reform, the failed Clinton Health Security Act of 1993 sparked enormous system change, especially within the hospital sector (Sisk and Glied 1994). In those years, industry advocates promoted their own version of a system model, which they labeled integrated delivery networks (IDNs). Thus, in the early 1990s, almost as if in anticipation of a successful effort to reform the industry, the hospital sector proposed that informal local and regional provider collectives form (the IDNs; AHA, Section for Health Care Systems 1990). They argued that structures of inter-organizational coordination would help to establish the clinical and management systems needed to ensure that patients and information would flow smoothly through the complex latticework of local provider systems. The industry proposals also acknowledged the need to restructure capacity (configuration changes), but choose to play these down, emphasizing patient flows over system reconfiguration.

As it turned out, the hospital sector gave mostly lip service to clinical integration while it engaged in one of the most significant periods system restructuring ever to have occurred in the industry's history. As discussed below, not only did hospitals rush into multi-hospital systems, but they formed clusters in an effort to consolidate local markets. And, those clusters have since discovered the power of inter-organizational strategies for incorporating physician practices, creating ambulatory surgery centers, expanding into other clinical areas (e.g., long-term care), restructuring clinical capacities and functionality across cluster members, and coordinating infrastructure.

It is notable that even the Veterans Health Administration (VHA) became involved in the movement toward regional models. In 1995, the VHA restructured its network of historically highly independent hospitals and clinics forming regionally coordinated Veterans Integrated Service Networks (VISNs; Perlin 2006). And, the VHA combined this innovative strategy with an integrated information system and a shift in emphasis toward ambulatory care—all of which would be coordinated by the VISNs. These changes, in combination, appear to have had a major impact on VHA health system performance (Ozcan and Luke 2011).

It is also significant that many of the advanced nations of the world have been adopting the regional model as an explicit policy strategy (Luke 2010). Canada, Australia, New Zealand, Scotland, Wales, most of the Scandinavian nations, a number of central European nations (e.g., Spain, Italy, and Portugal) are all heavily invested in regional strategies (Lewis and Kouri 2004; Gauld 2003; Healy et al. 2006). While it has not organized regional provider systems as such, England created 10 regional entities, called Strategic Health Alliances (SHAs), that are responsible regionally for overseeing and evaluating care, ensuring government policy is implemented, and recommending changes at the provider and institutional levels. England also plans and implements its IT strategies and operates its supply channel through regionalized configurations.

It is important to point out that the business clusters examined by Porter represent geographic combinations of otherwise independent companies, many of which are even direct competitors. By contrast, the health care clusters are more formally structured arrangements that, as a result, should have the advantage in overcoming the many sources of resistance that complicate inter-organizational reconfiguration and coordination among otherwise independent entities. On the other hand, formal structures do not guarantee that the clusters will be able to control their hospitals, let alone the physicians who join them. All providers groups historically have resolutely defended their autonomies and many even today are unwilling to abandon those autonomies despite having become members of systems. As a result, the clusters often face significant challenges in reforming and restructuring. It thus should be no surprise that the health care literature has thus far reported only limited performance improvements attributable to cluster formation (Bazzoli et al. 2004). Of relevant interest, however, is the use of Data Envelopment Analysis (DEA) in the study of the efficiency of hospital-based clusters (Sikka et al. 2009).

Nevertheless, as the systems mature and the environment changes, the clusters are likely to find

ways to overcome historic and future constraints to system rationalization. External pressures will likely spur changes and refinements within the clusters over time. Such pressures include growing technological complexities, the continuing probability of an extended downturn in the economy, the broad effects of health care reform, continued restrictions on payment, continued demographic change, increasing consolidation in the markets and consequent increases in competitiveness (non-price competition), continued substitution of ambulatory for acute care and the consequent need to integrate acute and other delivery modalities, and other important system and environmental changes. Within this powerful mix of forces and pressures for change, OR/MS analysts are likely to find many opportunities to apply their tools and techniques to help systems improve performance and gain competitive advantage within their markets.

Patterns of System Formation in the US

This section describes the configurational diversity that exists among the clusters. To illustrate this diversity, the focus will specifically be on the configuration of acute facilities, the lead, largest, and most important entities in most clusters. In so doing, it is recognized that many of the clusters are heavily invested in other related businesses, many of which are highly interdependent with the hospitals in terms of operations, production, and strategy.

The focus is specifically on three configurational dimensions: 1) number of hospitals per cluster, 2) the geographic dispersion of the hospitals, and 3) the hierarchical diversity among cluster hospital members. Using a national database on hospital systems, these three configurational characteristics have been reported, updated for the year 2009. (The hospital data are based on 25 years of monitoring system memberships and markets, the most recent full update was completed in 2009. This was supplemented by 2007 American Hospital Association Annual Survey data for individual hospitals).

Before examining the clusters, however, it is essential that they be distinguished from their parent organizations, the multi-hospital systems (MHSs). The MHSs are companies that own, lease or manage two or more hospitals, whereas the clusters are subunits of the MHSs; specifically, they are combinations of two or more same MHS hospitals that are located in the same geographic areas.

It is noted that some MHSs are themselves clusters (e.g., the 5 hospital INOVA health system located within the Northern Virginia part of the Washington D.C. metropolitan area). Other MHSs are dispersed much more widely in space, although many (but not all) of these operate one or more clusters. The Hospital Corporation of America, for instance, operates multiple clusters that are located in markets across the country. Alternatively, the for-profit SunLink Healthcare Corporation operates seven hospitals and nursing homes that are located in rural areas in the Midwest and South, no one of which is close enough to any of the others to constitute a cluster.

As of 2009, a total of 56 percent or 2,688 of 4,767 acute care hospitals nationally were members of 418 MHSs. The companies that operate these hospitals differ importantly by ownership. A total of 55 percent of MHS hospitals are in not-for-profit systems, 23 percent in for profit, and 22 percent Catholic systems. This compares to the distribution of MHSs by ownership, which breaks down to 79 percent not-for-profit, 9 percent for profit, and 11 percent Catholic. When examined together, these two distributions show that most for profit and Catholic MHSs are much larger on average, whereas the more numerous not-for-profit MHSs are far smaller as systems.

While the number of hospitals in MHSs grew significantly over the past 20 years, and especially so within the mid-1990s when the industry experienced a spike in mergers and acquisitions, the growth overwhelmingly favored not-for-profit hospitals and systems. Virtually all of the net growth in this period can be attributed to the not-for-profit sector (Luke 2010). This is very significant, for a number of reasons. First, and most importantly, a number of large, often referral not-for-profit hospitals took the initiative in this period to form systems and these hospitals mostly chose as partners other hospitals that were located within and/or around their same metropolitan areas. This development, therefore, not only generated a significant increase in the number of clustered system hospitals, but also produced a number of large and powerful hierarchical model types, which join together referral with community hospitals (as well as with other provider entities) into highly interdependent, hub-spoke cluster configurations.

Second, most of the restructuring affecting for profit and Catholic systems involved mergers among systems as opposed to new hospitals joining systems. But, these mergers also produced a number of important clusters, as the combined systems often brought same market hospitals together into clustered configurations. The key point is that clusters grew rapidly in this period, which dramatically altered the landscape of health care. And, this development has created a large number of very important entities that now need to be refined, rationalized, and optimized—a task, if properly performed, could produce very consequential strategic outcomes for these new and important organizational entities.

The Clusters

The central problem in defining clusters is specifying the outer geographic limit within which hospitals would appropriately be considered members of clusters. The most common approach is to define them as combinations of two or more same system hospitals located within the same CBSAs (core based statistical areas); METSAs–metropolitan statistical areas or MICSA–micropolitan statistical areas; e.g., see Cuellar and Gertler 2003 and 2005; Luke et al. 2003. While this is a highly reliable approach, it is also true that many same system hospitals are located outside the urban boundaries, in nearby rural or other urban areas. These often interact operationally and strategically with their urban partners, and thus they should be joined with their nearby urban partners in designating cluster memberships. On the other hand, relaxing the boundaries to include such other hospitals introduces error at the outer limit – how far out should one extend the boundaries to find the same cluster members?

A number of techniques for identifying the regional clusters were explored, including by setting distance breaks (e.g., 30 miles, 60 miles, 90 miles, etc.; Wong et al. 2005). However, such proved highly arbitrary, given that regionalized configurations vary dramatically from one to the other. Among the hospitals operated by the 15-hospital East Texas Medical Center Regional Healthcare System (ETMC), which operates out of 200,000 population Tyler, Texas, are two regional hospitals each of which is located about 110 miles from the system center (the location of the major ETMC referral center) in Tyler. One is north in Clarkesville and the other is south in Trinity. Both are operated as regional spokes in the hub-spoke ETMC system. By comparison, HCA operates two hospitals that are about the same distance apart – HCA's 194-bed St Lucie Medical Center, located in the Port St. Lucie-Fort Pierce, FL Metro area, and 235-bed Osceola Regional Medical Center, located in Kissimmee in south Orlando. HCA incorporates the former within its Miami cluster (within its East Florida Division) and the latter within a much smaller Orlando cluster (and more generally within its West Florida Division). Clearly, distance is not the only consideration required for designating cluster boundaries. Many other factors come into play, such as market density (both urban and rural), system ownership type, system organizational strategies, system size and complexity, and so on.

Fortunately, most clusters are relatively easy to identify, especially in the many cases in which their facilities are grouped exclusively within or just surrounding single metropolitan areas (such as was the case for the ETMC system). A small number of the regional clusters, however, were more difficult to identify, again, mostly those that combine hospitals run by the more dispersed, multi-hospital systems (which are mostly for profit and Catholic systems). Many of the clusters in these MHSs were easily discerned (e.g., the HCA clusters in Denver, Kansas City, Richmond, and in most other large metropolitan areas), but others were not. For the latter, a process of hospital inclusion, beginning with the largest markets per MHS, looking for same market and nearby same state combinations, and then working through the remaining hospitals to identify additional spatially-proximate combinations was used. Those few systems that specialize in small urban and non-urban markets (e.g., Community Health Systems and Lifepoint) presented the greatest challenge in identifying clusters. Accordingly, some judgment was required for a small number of clusters and cluster members belonging to systems such as these.

A total of 505 urban clusters and 638 regional clusters among 418 MHS chains operating in the U.S. have been identified. These represent 59 and 91 percent, respectively, of all MHS hospitals. Once the boundaries are relaxed and nearby hospitals (outside primary urban and/or rural centers) are

included, the number of hospitals and clusters obviously rises considerably – by 880 hospitals (a 56 percent increase) and by 134 clusters (a 26 percent increase). Clearly, a choice to use urban boundaries only would result in a very significant underrepresentation of clusters. Such a choice also would affect the configurational characteristics of the clusters, such as their relative dispersion and the combinations of large and small hospitals and other providers within the clusters. In what follows, therefore, the emphasis is on the regional (as opposed to the urban) approach to defining and measuring clusters.

Cluster Configurations. The important point in this article is that the clusters come in many different forms, the characteristics of which should be important when applying OR/MS techniques to solving system problems. Three configurational dimensions were identified earlier in the paper (focusing on the acute care facilities within the clusters) – cluster size (numbers of hospitals per cluster), geographic dispersion, and hierarchical diversity. It is probable that the size of the cluster itself will affect the other configurational dimensions. Larger clusters will likely have larger central hospitals, more dispersion, and more diversity in business units. It is also likely that the size of the market will affect the patterns as well. For instance, the dispersion of facilities beyond the urban boundaries is very much greater in smaller than in larger markets. Also, the size of the central or lead hospital facility per cluster tends to be much larger in the larger markets. Further, the size of the smaller hospitals in each cluster are likely to be much smaller in the clusters that emanate out of smaller urban areas. Thus, given the likely importance of both cluster and market size, control for these two dimensions is emphasized in the analyses below.

Cluster size. The average number of hospitals per cluster is just under four hospitals per cluster. Two-hospital clusters represent 39 percent of the total, which means that just over 60 percent of clusters have three or more hospitals in them (25 percent are five or more). This suggests that many of the clusters represent much more than mere hospital/hospital mergers, but rather mature (in terms of size and scope) multi-organizational groupings. As might be expected, the larger clusters tend to be found in the larger markets of one million population and over – 4.4 hospitals per cluster – compared to those in the smaller markets – 3.4 hospitals per cluster. While this relationship is statistically significant, there is much variation across market size categories (as reflected in the very large ETMC system described above).

Hospital dispersion. Within clusters, the average miles per cluster from the urban center (defined by the location of the largest hospital in the cluster) to each cluster member is about 21 miles. As would be expected, the averages increase by cluster size and decline with market size. The relationship with market size reflects a common pattern in which clusters located in larger markets tend to combine urban members, whereas those located in smaller markets often spill over to nearby rural and other urban areas. The average distances are 16 miles for clusters centered in markets of one million and over and 24 miles for those centered in markets fewer than one million. Clusters centered in the smaller markets, in other words, are very much more likely to incorporate within them more distant hospitals located outside their urban boundaries. The distances rise with cluster size, as follows: 2 hospitals – 14 miles, 3 to 4 hospitals – 22 miles, 5 to 6 hospitals – 28 miles, and 7 & over hospitals – 33 miles. (Means in the market and cluster size comparisons are significantly different). This reflects the need for clusters, as they grow, to reach further out in distance to find additional cluster partners.

Hierarchical configuration. This reflects the degree to which clusters combine large and small hospitals (the hub-spoke model). The range in bed size averages 278 beds, which suggests considerable variation across individual cluster members in the size of their hospitals. The significance of this is that differentiation among the cluster members indicates possibilities for within cluster rationalization of function and service capacity. And, the range is positively associated with cluster and market size. The ranges by cluster size are: 170, 288, 383, and 481 beds for clusters with 2, 3–4, 5–6 and seven and over hospitals per cluster. For 1 million and over markets, the average range in beds is 326 beds and for those centered in markets under one million, it is 240 beds. (All of these means are statistically significantly different from one another.) This suggests that the larger clusters located in the larger markets represent

those clusters that are most likely to exhibit significant functional differentiation among cluster members.

Considering the three dimensions together (size, dispersion, and hierarchy), it is clear that the clusters come in very different configurational combinations. And, these also vary directly with market size. The larger market clusters tend to be larger, less widely dispersed, more contained within the urban boundaries, and more hierarchical, by comparison to those in smaller markets.

Analytical Challenges in Studying Clusters

The most important challenge that arises when studying multi-unit business organizations (as opposed to studying free-standing facilities) is the analytical problems become exponentially more complicated. This is all the more true for spatially interdependent entities, such as the health care clusters. To help make this point clear, four issues are discussed that highlight some of the analytical challenges inherent in the study of geographically configured, multi-organizational forms: boundary specification, intensity of interdependencies, business unit diversity, and aggregation.

Boundary specification. This issue was raised above when the specification of the spatial boundaries of clusters were discussed. But, this is important for the OR/MS analyst as much as it is for the health services researcher. A too narrow definition of cluster boundaries risks under representing the total resources involved in the delivery of services, which could significantly affect optimization solutions involving multi-organizational systems. The reverse would be true for a too broad definition of cluster boundaries. Obviously, the boundary specification problem is less of a concern for clusters that are limited to urban areas or even are located within single counties. However, as discussed, many clusters spread well outside such boundaries, including, in particular, clusters that are centered within smaller urban markets. The boundary specification problem makes it necessary for the OR/MS researcher to grapple with the distinctive geographies, the logic of individual system design, and, importantly, the diversity and complexity of the management problems being addressed.

Intensity of interdependencies. Differences in the intensity of interdependencies can greatly compound the organizational boundary issue. Interdependencies will vary by location, functional capabilities, facility size and power, tightness of legal ownership arrangements, distinctive characteristics of parent organizations, and many other organizational characteristics. For example, a small, relatively distant hospital might have a tight and intense relationship with an urban centered, same system, referral hospital member, whereas a large, suburb-based (and much more geographically proximate) hospital might be very much more self sufficient with regard to its nearby referral partner. Given its self sufficiency, the suburb member might even be a fierce competitor with its same system partner. On the other hand, the more geographically proximate same system facilities might have more opportunities to engage in other coordinative activities, such as integrating purchasing and supply distribution, physician recruitment, IT strategies, the establishment of other related businesses within their metropolitan areas (e.g., ambulatory surgery centers), marketing activities, laboratory and other clinical and technological support activities, and so on. Each cluster, in other words is likely to be somewhat unique in how it captures the synergies of same system membership and geographic proximities.

Facility interdependencies will vary not only across clusters, but also across markets. The Wellstar health system, for instance, that operates five hospitals in the northwest quadrant of Atlanta, is likely to view system interrelationships among its hospitals far differently than will the New York Presbyterian Health System that geographically concentrated throughout the densely populated New York metropolitan area. Facilities in the latter tend to be larger and much more independent, given history and other factors.

It should be clear that OR/MS analysts must take particular care to ensure that they understand the particular configurational, structural, integrative and competitive features of the clusters they study.

Business unit diversity. The clusters also vary greatly in their diversity of business units. Indeed, the U.S. health care clusters are becoming far more than acute care agglomerations; many are expanding more rapidly into clinical areas well beyond acute care. This has important implications for how one conceptualizes

the clusters as production systems. Some of these other businesses are highly interdependent with cluster acute care activities, others are only tangentially related. Ambulatory care surgery is perhaps the most interconnected, because every case that is performed in the ambulatory setting represents a lost admission (and, lost revenues) to a hospital. However, this is not just a relationship of financial substitution. Hospitals provide much needed back up to their free-standing ambulatory surgery centers, by covering them in the event complications arise, providing ancillary services, coordinating site of surgery, recruiting physicians and staff, offering supply channel and other management support, and so on.

The largest private hospital chain in the country, HCA, has invested heavily in ambulatory surgery centers. HCA owns and operates 105 freestanding surgery centers and about 165 hospitals nationally, as well as in London. Significantly, all of HCA's surgery centers are located in markets in which HCA already has at least one hospital. This suggests that HCA is not investing in ambulatory businesses per se, but rather is investing in this rapidly growing business as an extension of the acute care business and to strengthen market positions within the markets in which the company already has facilities. Ambulatory surgery centers thus have become major components of the clusters' overall production and market strategies and, therefore, should be incorporated into the overall analyses of cluster operations and performance.

Many other system and production interrelationships also need to be understood when assessing cluster performance. The urban/rural clusters offer a particularly interesting example of high interdependencies among spatially distant cluster members. The Tyler, Texas-based ETMC health system, is actively engaged in integrative and support activities between its hub facility and the smaller, highly dependent rural facilities. ETMC, for example, operates emergency transport systems (both helicopter and ground systems), mobile imaging and other clinical services vehicles, and a network of clinics and specialty service centers that support their geographically sprawling system. Notably, there is no single template for how the clusters are configured or what particular services and business units they offer. There are general patterns, but every cluster is comprised of a unique mix of resources, capabilities, competitors, geographic constraints, and environmental conditions that determine their mix of providers and patterns of interrelationships. The key point is that distinctive combinations of facilities and businesses in geographic space greatly increases analytical complexity, making it essential that the systems are analyzed as integrated production systems (Ozcan 2009).

The foregoing is further complicated by the regulatory and legal contexts within which multi-organizational coordination takes place, especially within shared geographies. For instance, health systems are subject to Federal policies promulgated by CMS (Federal Centers for Medicare and Medicaid Services) that greatly increase the strategic importance and the urgency with which systems engage in coordination to achieve system improvements. As an example, the government's use of "meaningful use" criteria (require providers to meet minimum standards in the use of certified electronic health record technologies) to support bonus payments to health care providers makes it all the more important that their investments in costly and strategically significant health information technologies be examined at the cluster level (Blumenthal and Tavenner 2010; Ford et al. 2010; and Hikmet et al. 2007).

Aggregation. Related to the above is the problem of aggregation. This issue may not be as important when examining single-facility organizations. But, when multiple, geographically configured units are studied as if they were single production systems, it becomes necessary to address issues of standardization and weighting. Often this is a simple matter of summing across the units as if they were mere extensions of a production system. But, the diversity that exists across hospitals, let alone, between hospitals and other provider units, makes it necessary to consider how to count inputs and outputs. How, for instance, does one weight hospital-based surgeries versus ambulatory care surgeries? How does one weight outpatient visits versus emergency department visits. These problems exist even when examining single organizations, but they are magnified when evaluating multi-organizational systems that are as large and complex as are the clusters.

See

- ▶ Decision Analysis
- ▶ Health Care Management
- ▶ Hospitals

References

AHA, Section for Health Care Systems. (1990). *Renewing the U.S. health care system*. Washington, DC: AHA.

Andrews, K. R. (1971). *The concept of corporate strategy*. Homewood, IL: Irwin.

Barney, J. B., & Clark, D. N. (2007). *Resource-based theory creating and sustaining competitive advantage*. Oxford, England: Oxford University Press.

Bazzoli, G. J., Dynan, L., Burns, L. R., & Yap, C. (2004). Two decades of organizational change in health care: What have we learned? *Medical Care Research and Review, 61*, 247–331.

Bice, T. W. (1984). Health services planning and regulation. In S. J. Williams & P. R. Torrens (Eds.), *Introduction to health services* (2nd ed.). New York: Wiley.

Blumenthal, D., & Tavenner, M. (2010). The "meaningful use" regulation for electronic health records. *The New England Journal of Medicine, 363*, 501–504.

Cuellar, A. E., & Gertler, P. J. (2003). Trends in hospital consolidation: The formation of local systems. *Health Affairs, 22*, 77–87.

Cuellar, A. E., & Gertler, P. J. (2005). How the expansion of hospital systems has affected consumers. *Health Affairs, 24*, 213–219.

Falk, I. S. (1958). The committee on the costs of medical care – 25 years of progress. *American Journal of Public Health, 48*, 979–982.

Ford, E. W., Menachemi, N., Huerta, T. R., Yu, F., & Moore, R. A. (2010). Hospital IT adoption strategies associated with implementation success: Implications for achieving meaningful use/practitioner application. *Journal of Healthcare Management, 55*(3), 175–190.

Fox, D. M. (1986). *Health policies, health politics: The British and American experience, 1911–1965*. Princeton, NJ: Princeton University Press.

Gauld, R. (2003). One country, four systems: Comparing changing health policies in New Zealand. *International Political Science Review, 24*, 199–218.

Healy, J., Sharman, E., & Lokuge, B. (2006). Australia: Health system review. *Health Systems in Transition, 8*(5), 1–158.

Hikmet, N., Bhattacherjee, A., Menachemi, N., Kayhan, V. O., & Brooks, R. G. (2008). The role of organizational factors in the adoption of healthcare information technology in Florida hospitals. *Health Care Management Science, 11*(1), 1–9.

Lawrence, P., & Lorsch, J. (1967). Differentiation and integration in complex organizations. *Administrative Science Quarterly, 12*, 1–30.

Lewis, S., & Kouri, D. (2004). Regionalization: Making sense of the Canadian experience. *Healthcare Papers, 5*, 12–31.

Luke, R. D. (2010). System transformation: USA and international strategies in healthcare organization and policy. *International Journal of Public Policy, 5*(2/3), 190–203.

Luke, R. D., Walston, S. L., & Plummer, P. M. (2003). *Healthcare strategy: In pursuit of competitive advantage*. Chicago: Health Administration Press.

Marshall, A. (1920). *Principles of economics*. London: Macmillan.

Ozcan, Y. A. (2009). *Quantitative methods in health care management: Techniques and applications* (2nd ed.). San Francisco: Jossey-Bass/Wiley.

Ozcan, Y. A., & Luke, R. D. (2011). Healthcare delivery restructuring and productivity change: Assessing the veterans integrated service networks (VISNs) using the malmquist approach. *Medical Care Research and Review, 68*, 20S–35S.

Perlin, J. B. (2006). Transformation of the US veterans health administration. *Health Economics, Policy and Law, 1*, 1–7.

Porter, M. E. (1985). *Competitive advantage: Creating and sustaining superior performance*. New York: The Free Press.

Porter, M. E. (1996). *On competition*. Boston: Harvard Business Review Book.

Rais, A., & Vianna, A. (2011). OR in healthcare: A survey. *International Transactions in Operational Research, 18*(1), 1–31.

Sikka, V., Luke, R. D., & Ozcan, Y. A. (2009). The efficiency of hospital-based clusters. *Health Care Management Review, 34*(3), 251–261.

Sisk, J. E., & Glied, S. A. (1994). Innovation under federal health care reform. *Health Affairs (Summer), 13*, 82–97.

Starr, P. (1982). *The social transformation of American medicine*. New York: Basic Books.

Stevens, R. (1989). *In sickness and in wealth*. Baltimore: The Johns Hopkins University Press.

Wong, H., Zhan, C., & Mutter, R. (2005). Do different measures of hospital competition matter in empirical investigations of hospital behavior. *Review of Industrial Organization, 26*(February), 27–60.

Heavy-Tailed Distribution

A probability distribution that has more probability in its tail than exponentially decaying densities such as the normal (Gaussian) and exponential distributions. Sometimes also called fat-tailed distribution, particularly in the finance community. The precise technical definition may differ in the literature, but a commonly used one is the following: The cumulative distribution function F is heavy-tailed if there exists a $\gamma > 0$ such that

$$\lim_{x \to \infty} e^{\gamma x} F^c(x) = \infty,$$

where the superscript c denotes the complement.

Special cases include long-tailed distributions and subexponential distributions. The most commonly known heavy-tailed distributions are also long-tailed and subexponential, including the lognormal, Pareto,

and Weibull with certain shape parameter values (one-tailed), and the student-t, Caucy, and family of stable distributions (two-tailed).

See

- ▶ Light-Tailed Distribution
- ▶ Rare Event Simulation

Heavy-Traffic Approximation

As the traffic intensity of a queueing problem approaches 1 (from below), the measures of effectiveness for the system often take on patterns which become essentially insensitive to the exact form of the input and service processes defining the system and, for example, may depend only on expectations and variances. As an illustration, the distribution for line delay of the general G/G/1 queue with utilization rate $\rho = 1 - \varepsilon$ can be well approximated by $W_q(t) = 1 - \exp(-at)$, where $a = (1/2)$(interarrival time variance + service-time variance)/(mean interarrival time − mean service time).

See

- ▶ Queueing Theory

Hedging

In finance, a trading strategy that leaves one indifferent to market outcomes. For example, a delta hedge of a stock option is intended to make the portfolio indifferent to whether the underlying stock price increases or decreases.

See

- ▶ Financial Engineering
- ▶ Financial Markets
- ▶ Risk Management for Software Engineering

Hessenberg Matrix

A matrix that would be upper triangular except for having nonzero elements immediately below the main diagonal. Such matrices arise when trying to preserve sparsity in computing a matrix inverse.

See

- ▶ Matrices and Matrix Algebra

Hessian Matrix

For a function $f(\boldsymbol{x})$ of the n-dimensional vector variable $\boldsymbol{x}$, the Hessian, denoted by $\nabla^2 f(\boldsymbol{x})$, is an $n \times n$ square matrix of second-order partial derivatives (assuming they exist) evaluated at a specific point $\boldsymbol{x}$, with the $(i, j)th$ element given by

$$\nabla^2 f(\mathbf{x})_{ij} = \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j}.$$

If the second partial derivatives are continuous at $\boldsymbol{x}$, then the Hessian is a symmetric matrix.

See

- ▶ Nonlinear Programming
- ▶ Quadratic Programming

Heterogeneous Lanchester Equations

Differential (of difference) equations equating force size changes for each of several weapons systems (components) on each side to sums of the products of coefficients and component force sizes. The concept is that each component is attrited to some degree by each component of the opposing side; however, the killing mechanism and rate depend on the pairing. Hence, each term defines the mechanism (such as square law or linear law) and rate (the coefficient) and the sum of

the terms defines the total attrition for the system. Therefore, rather than the two equations of a homogeneous Lanchester law, there is one equation for each component of each side.

See

- ▶ Battle Modeling
- ▶ Lanchester's Equations

Heuristic Procedure

For a given problem, a collection of rules or steps that guide one to a solution that may or may not be optimal. The rules are usually based on the problem's characteristics, intuition, hunches, good ideas, or reasonable processes for searching.

See

- ▶ Greedy Algorithm
- ▶ Heuristics
- ▶ Metaheuristics
- ▶ Simulated Annealing
- ▶ Tabu Search

Heuristics

Manuel Laguna[1] and Rafael Martí[2]
[1]University of Colorado Boulder, Boulder, CO, USA
[2]University of Valencia, Valencia, Spain

Introduction

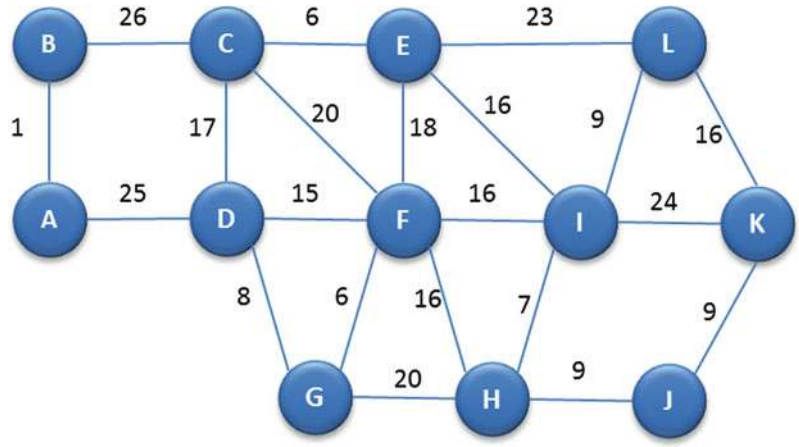
The word heuristics derives from the Greek *heurisken*, which means to find or to discover. In general, for a given problem, a heuristic procedure is a collection of rules or steps that guide one to a solution that may or may not be the best (optimal) solution. The rules are usually based on the problem's characteristics, reasonable processes for searching, plus one's intuition, hunches, or good ideas.

In general, heuristics describes a class of procedures for finding acceptable solutions to a variety of difficult decision problems, that is, procedures for searching for the best solutions to optimization problems. The solution set of most real world optimization problems often include a large or even an infinite number of possible solutions, as well as a criterion or a set of criteria to evaluate the merit of a solution. These problems may be stated as finding the values for a set of decision variables for which one or more objective functions reach a minimum or a maximum value. Restrictions may be placed on the values of individual variables or combination of variables. In what follows, important operations research models are discussed first to help illuminate the main theme of heuristics.

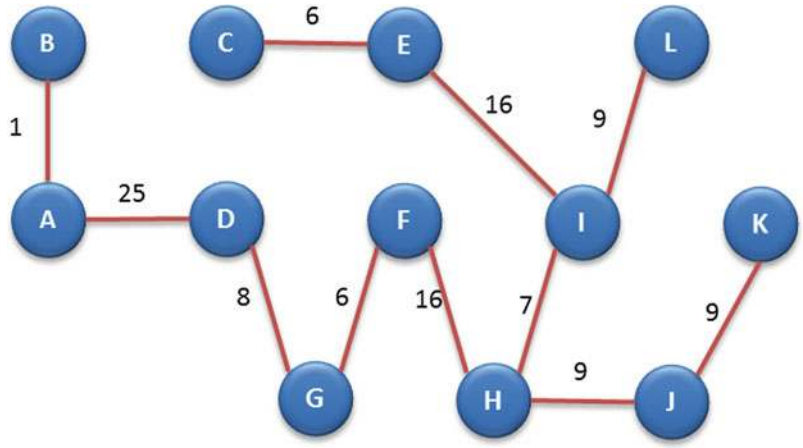
Illustrative Examples

Optimization problems are found in many areas of business, engineering, and science. Some problem classes are relatively easy to solve. For instance, consider a simplified version of a problem that arises in telecommunications in which it is desired to connect a number of customers in a network using the least amount of cable. Figure 1 shows a network of 22 possible cable links joining 12 customers (labeled A to L) and the costs of the potential links. Note, that with 22 links, there are many ways a subset can be selected that would connect all customers. The problem is to find the subset with the least amount of total cable. That is, for each customer, one of the possible links that connects the customer has to be chosen. For the network in Fig. 1, can 11 such links (decision variables) be found?

It is well know that the optimal solution (i.e., the connection that guarantees the minimum cable cost) can be found by a simple procedure that starts with choosing the link with the smallest cost in the network. Similarly, the remaining links are chosen successively to minimize the increase in total cost at each step, where the links considered meet exactly one customer from those that are endpoints of links previously chosen. The resulting solution is called a tree, which is defined as a set of links that contains no cycles, i.e., the tree contains no paths that start and end at the same customer (without retracing any links). In Fig. 1, the first link to be added to the solution is A-B, with a cost of 1. To minimize the additional cost of connecting a new customer, the A-D link must be



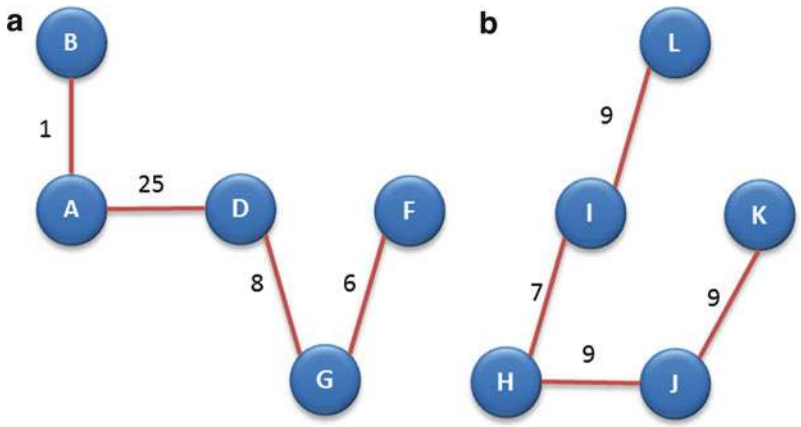
Heuristics, Fig. 1 Illustrative network with 12 customers



Heuristics, Fig. 2 Minimum spanning tree

chosen. That link has a cost of 25, resulting in a total cost of 26 to connect A, B, and D. Now, it is possible to connect either customer C, F or G. Cost minimization dictates that link D-G should be added, resulting in customer G to be connected to the partial solution with a total cost of $1 + 25 + 8 = 34$. The next candidates to be connected to the current tree are customers C, F, and H. The link with the minimum cost between the end points of the current tree and the candidate customers is G-F with a cost of 6. After adding this link, the total cost is 40. Subsequent links are added in a similar fashion, resulting in the tree shown in Fig. 2 that has a total connection cost of 112. The above process can be interpreted as a heuristic approach to solving the connection problem by someone uninitiated in any formal knowledge of trees or related matters.

The tree shown in Fig. 2 is called a minimum spanning tree. It can be shown to be a least cost optimal tree for the problem. The procedure avoids cycles because it is easy to verify that no optimal solution of the problem includes a cycle, since a cycle adds an unnecessary link and therefore additional cost (assuming that all costs are strictly positive). Note that an alternative optimal solution is available in which the F-H link is replaced with the F-I link. Since ties are arbitrarily broken, either solution is acceptable and optimal. The procedure is exact, that is, it guarantees an optimal solution; it is simple and can be applied to very large networks. Unfortunately, the great majority of optimization problems are not as easy to solve. In fact, sometimes what seems to be a simple change to a problem definition may turn the problem from easy to hard. For example, if the problem of



Heuristics, Fig. 3 (**a**) Heuristic solution versus (**b**) optimal solution

finding the minimum-cost connection of all customers is changed to finding the minimum-cost connection of a subset of customers, the problem becomes extremely difficult. This is known as the minimum k-tree problem and consists of finding a tree with k links so that the sum of the costs is minimized. Figure 3a shows the solution that is obtained from the application of the procedure described above for the case of $k = 4$. However, this solution (with a cost of 40) is not optimal for the minimum k-tree problem. The optimal solution is shown in Fig. 3b and has a total cost of 34. A solution procedure that is exact (i.e., optimal) for the minimum spanning tree problem becomes a heuristic for the minimum k-tree problem.

Linear programming is one of the most valuable mathematical modeling techniques in operations research. As an example that reflects the main heuristics theme of this article, consider a relatively simple linear-programming model known as the continuous knapsack problem. Here, there are 10 items available to be placed in a knapsack whose associated weights are given in the subject to inequality; the utility or value of each item to the person with the knapsack is given in the maximize objective function, e.g., item 1 has a weight of 5 and a utility of 67. Note that this example allows a fraction of an item to be placed in the knapsack.

$$\begin{aligned}\text{Maximize}\quad & 67x_1 + 500x_2 + 98x_3 + 200x_4 + 120x_5 \\ & + 312x_6 + 100x_7 + 200x_8 + 180x_9 + 100x_{10} \\ \text{Subject to}\quad & 5x_1 + 45x_2 + 9x_3 + 19x_4 + 12x_5 + 32x_6 \\ & + 11x_7 + 23x_8 + 21x_9 + 14x_{10} \leq 100 \\ & 0 \leq x_j \leq 1 \quad \text{for j} = 1, \ldots, 10\end{aligned}$$

Linear-programming problems are typically solved using standard techniques such as the simplex algorithm. The problem above, however, has a special structure that makes it easy to find the optimal solution by applying what can be considered to be an obvious heuristic approach. Specifically, an optimal solution procedure to this problem can be found by performing three simple steps:

1. Order the variables by their decreasing "bang for the buck" ratio. The ratio is calculated by dividing the objective function coefficient (utility) by the corresponding constraint coefficient (weight).
2. Consider each variable in order, one at a time, and set its value as large as possible without violating its upper bound of 1 or the 100 capacity restriction.
3. When capacity is reached, set the remaining variables to zero.

For convenience, the variables in this example have been ordered by their bang for the buck ratio. The procedure may then be applied by simply considering the variables in the order as specified by their index value. The application of the three steps requires setting the first 5 variables to their maximum possible value of 1 with an associated total utility of 67 + 500 + 98 + 200 + 120 = 985; the total weight is 5 + 45 + 9 + 19 + 12 = 90. Since variable 6 has a weight of 32 and the remaining capacity is 10, then this variable cannot take on its maximum value of 1. Thus, variable 6 is set to a value of 10/32 = 0.3125. The additional utility is (0.3125 × 312) = 97.5 for a total objective function value of 985 + 97.5 = 1,082.5, the optimal value to the example. The three-step procedure is an exact method for this problem because it guarantees to find an optimal solution.

The problem may be transformed into an integer-programming model, by requiring the variables take on integer values, in this case 0 or 1. This interpretation requires each variable to be either wholly selected for inclusion in the knapsack, or not selected at all, until the knapsack reaches its limited capacity or falls below it as the selection of any other item will exceed the capacity. Again, the objective function coefficients represent the utility of the item, the constraint coefficients represent the weight of the items, and the right-hand-side of the equation indicates the total weight limit (or capacity of the knapsack). One logical solution approach to finding a solution is to apply the bang for buck method and then stop when it is not possible to select another item without violating the capacity limit. This may be achieved by simply assigning a value of zero to the variable that has a fractional value in the solution to the continuous problem (i.e., the one that does not force the variables to take on integer values). In particular, the rounded solution to the example above is such that the first five variables (x_1 to x_5) take on values of 1 and the last 5 variables (x_6 to x_{10}) take on values of zero. The total utility of this solution is 985 with a total weight of 90. It turns out, however, that this solution is not optimal. The method that is exact for the continuous case becomes a heuristic for the integer case. Finding the optimal solution would require an additional search that would replace item 5 with item 9. That is, variable x_5 should be set to 0 and x_9 to 1. The net gain in utility is 60 for a total utility of 1,045 and the total weight becomes 99.

The Role of Heuristics

The preceding examples illustrate the notion of problem difficulty. It is often possible to readily find optimal solution to some problems, while what may be perceived as a simple change to the problem may increase considerably the difficulty of finding an optimal solution. The meaning of a difficult problem is captured by the computer-science term NP-hard, which is commonly applied in the context of algorithmic complexity. A difficult optimization problem is one for which it is not possible to guarantee that the optimal solution will be found within a reasonable computational time. The existence of a great variety of difficult problems that arise in practice motivated the development of efficient procedures capable of finding good (or acceptable) solutions even when these solutions could not be proven optimal, that is heuristic methods. The development of a heuristic method is usually concerned with both solution speed and solution quality. A definition of heuristics in the context of optimization is the following:

A heuristic is a well-defined intelligent procedure — based on intuition, problem context and structure — designed to find an approximate solution to an optimization problem.

In contrast with exact methods that are designed to find optimal solutions, heuristic methods find solutions that are not necessarily optimal. The time that exact methods require for finding and proving the optimality of a solution is typically orders of magnitude larger than the time required by a heuristic. The effectiveness of a heuristics depends on the quality of the approximations that it produces. Heuristics have not always been accepted as an elegant and perhaps even valid form of optimization, as noted by Fred Glover (1977, p. 156):

> Algorithms are conceived in analytic purity in the high citadels of academic research, heuristics are midwifed by expediency in the dark corners of the practitioner's lair . . . and are accorded lower status.

The effectiveness of heuristics, particularly when applied to difficult practical problems (such as those in the area of combinatorial optimization), has made them popular among practitioners and academics, as reflected by the increased number of articles and publications devoted to them. Combinatorial optimization is a fertile area of application for heuristics because it includes a large number of practical problems that are difficult to solve (within reasonable computer time) by means of exact procedures. The objective of these problems is to maximize or minimize a function over a finite set of solutions. No conditions or properties are placed on the form of the objective function and the set of feasible solutions tends to be so large that evaluating them all to search for the best is impractical. The minimum k-tree and the knapsack problems fall within the area of combinatorial optimization. There are several reasons for employing heuristics when facing an optimization problem:

- The problem is such that no exact solution method is known for it.
- Known exact solution methods are computationally expensive and, therefore, they are able to solve only small instances of the problem.

- The flexibility of the heuristic approach enables the incorporation of realistic problem features that otherwise would be difficult to model.
- A heuristic method is used within an exact procedure to generate an initial solution or to guide the search.

Typically, heuristic methods are developed for a particular class of problems. That is, most heuristics are context dependent. For example, there are many heuristics for scheduling jobs in production settings that take the form of dispatching rules: FIFO (first in first out) specifies that jobs should be processed in the order in which they arrived, SPT (shortest processing time) suggests to process the smallest (in terms of estimated duration) jobs first, and EDD (earliest due date) gives preference to those jobs that are more immediately due. Some of these dispatching rules are indeed optimal under certain limited circumstances, but for the most part they are applied as heuristic procedures.

Heuristics are also found in more general settings for which the context is provided by the modeling framework, e.g., the class of problems that may be modeled as integer programs that are solved with exact methods based on branch and bound. These methods are such that the search for the optimal solution results in the examination of many partial solutions. Within this structure, it is desirable to eliminate search directions that can be ruled out because it is possible to determine that they lead to suboptimal solutions. This can be accomplished by improving the quality of the best-known solution, also referred to as the incumbent solution. If a partial solution consists of a subset of variables for which their integer values have been fixed by a sequence of systematic decisions, with the remaining variables holding fractional values, a rounding heuristic may be applied to convert the fractional values into integers to obtain a complete solution to the problem. If the rounded heuristic solution becomes the incumbent solution, some branches of the search tree may be eliminated. The optimization software Cplex allows the use of heuristics during the branch and bound process. These heuristics may be used to find feasible solutions quickly, avoid exploring unproductive subtrees, and diversify the search. Within this framework, the combination of branch and bound and local search heuristics has been suggested and applied to tackle difficult integer-programming problems. Furthermore, a branch-and-bound process that is terminated prematurely and hence not allowed to confirm the optimality of an incumbent solution is considered a heuristic method.

Classification of Heuristics

Although most heuristics are designed for specific problems, it is possible to classify them into five general categories:

- **Decomposition**: These procedures decompose the original problem into subproblems that are simpler to solve. The solutions to the subproblems are merged to provide a solution to the original problem.
- **Induction**: These procedures are based on the notion that solution strategies learned from small or simplified instances of the original problem may be applied to the original problem.
- **Reduction**: This technique consists of identifying properties that are common to good or optimal solutions and introducing them as problem constraints. The goal is to reduce the solution space and hence simplifying the original problem. The risk in doing so is that the optimal solution may be unintentionally left out.
- **Construction**: Many heuristics fall into this category, which includes all those procedures designed to build solutions by applying a sequence of selection steps. The selections are typically deterministic and are based on a measure of merit for choosing elements that are not yet in the solution.
- **Local Search**: These procedures operate on an existing solution to the problem with the goal of improving it. The neighborhood (as defined by a move mechanism) is explored at each step and the process continues as long as it is possible to move to a neighbor with a better objective function value.

The merit of heuristic procedures is judged by their efficiency (i.e., computational effort), the average quality of the solutions that they produce, and their robustness (i.e., their ability to avoid extremely inferior solutions). To assess the performance of a heuristic procedure, researchers and practitioners usually rely on the following five methods:

- **Comparison against optimal solutions**: Occasionally, it is possible to find optimal

solutions to a limited number of instances of the problem of interest. The solutions obtained by the heuristic may be compared to the optimal solutions to measure, for example, the percent deviation from optimality.

- **Comparison against bounds**: When optimal solutions are not available, it is possible, in some situations, to calculate bounds that can be used to assess solution quality. Of course, the quality of the assessment depends on the quality of the bound, as measured by its proximity to the optimal solution.
- **Comparison with an incumbent solution from a truncated search of an exact method**: A common practice when attempting to solve difficult problems (e.g., those formulated as integer programs) consists of applying an exact procedure, such as branch and bound, for a limited computational time. If successful, the search yields a feasible incumbent solution and an optimality gap. These values may be used to assess the quality of a heuristic solution.
- **Comparison against other heuristics**: If alternative heuristics for the problem of interest are available, their solutions can be used as benchmarks for comparison purposes.
- **Worst case analysis**: This assessment method is quite popular and consists of identifying a bound for the worst possible performance of the heuristic procedure. The advantage of this analysis is that it provides a guarantee that the performance of a procedure (typically measured as a deviation from the optimal solution) will never fall below a certain value. The disadvantage, however, is that knowledge of this worst case does not give information on average or best performance. Also, the worst-case analysis is not trivial for most heuristics.

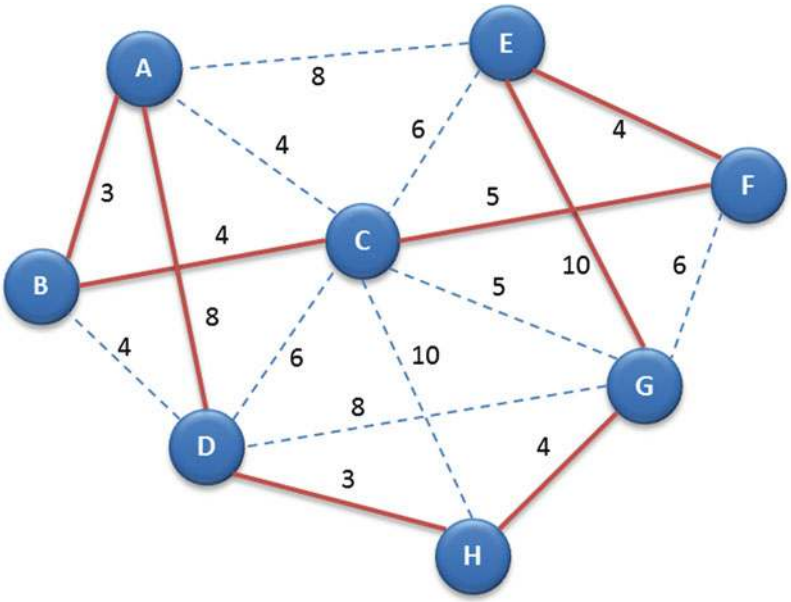
Construction Search Heuristics

Construction and local search heuristics have become the cornerstones of the development of many metaheuristics that include strategies to search the solution space beyond local optimality (Glover 1986; Glover and Kochenberger 2003). The attitude towards the development and application of heuristics and metaheuristics as tools for optimization has changed over the years, gaining their recognition as valuable methods of analysis. In some application areas, such as scheduling, heuristics and metaheuristics are indeed the main solution approaches. Morton and Pentico (1993) make an excellent case for heuristics in scheduling production systems and project management. Related to scheduling and sequencing is the traveling salesman problem (TSP). Its objective is to find a tour (cyclic permutation) that visits a set of cities such that the total distance traveled is minimized. The TSP represents an appropriate context to illustrate the principles associated with construction and local search heuristics.

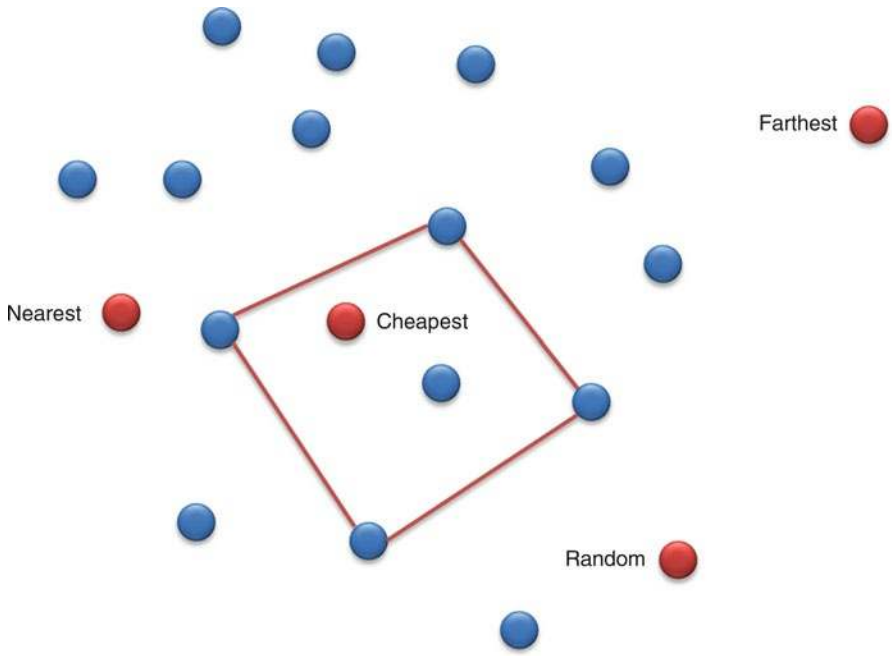
The TSP is typically stated in terms of a problem on graphs. Given a complete graph (i.e., a set of nodes and edges that connect each node to all others) and a distance matrix, the TSP consists of finding the shortest Hamiltonian cycle (tour) in the graph. The symmetric variant of the TSP, used for illustrative purposes below, assumes that the distance from one node to any other is the same in both directions. Figure 4 shows a partial representation of a graph with 8 vertices, in which, for the sake of clarity, not all possible edges (dotted lines) are included and a tour is shown by solid lines.

The nearest neighbor procedure is probably the most intuitive heuristic for obtaining a tour. It starts by randomly selecting a node in the graph. Then, it adds the edge that connects the node that is closets to the last node added to the partial tour. The procedure terminates after all nodes have been included and the edge that connects the last node to the first one is added to the tour. This procedure is an example of a so-called greedy heuristic because at each step it selects the most attractive option without considering that this can lead to inferior choices in future steps. That is, the selection of the next element to be added to the solution is myopic.

An alternative approach to obtaining a relatively good tour is the insertion procedure. In this heuristic, nodes are inserted in the best position of the current partial tour. That is, the node is inserted in the position that results in a new subtour of minimum length. As shown in Fig. 5, four different criteria are typically considered for the selection of the node to be inserted: the nearest (closest to the nodes in the subtour), the cheapest (closest to the edges in the subtour) the farthest and one at random. The selection of the farthest may seem counterintuitive, but since this node will become part of the tour at



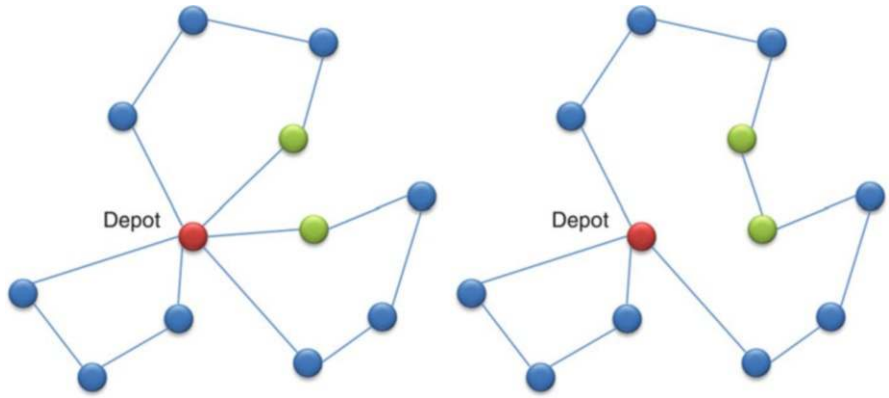
Heuristics, Fig. 4 Graph with 8 nodes and TSP tour



Heuristics, Fig. 5 Subtour and insertion candidates

some point, selecting it earlier in the construction may in fact be beneficial.

The savings heuristic is a specialization of a more general heuristic proposed by Clarke and Wright (1964). An arbitrary node is selected to play the role of the central node or depot. The method starts by creating subtours from the depot to each other node. One edge is added to go from the depot to a node and one to come back from the node to the depot. This initially creates a number of subtours that is



Heuristics, Fig. 6 Step of Clarke and Wright savings heuristic

equal to the number of nodes in the graph minus one (the depot). Then, at each step, two subtours are merged as shown in Fig. 6 (by replacing a length-2 path from one non-central node to another by a direct link), thus reducing the number of subtours by one unit. The method finishes when all the sub-tours have been successively merged into a single tour.

The Christofides (1976) procedure is another heuristic for constructing TSP tours. It is based on graph theory. It starts with a minimum spanning tree to which the number of edges is doubled to obtain an Eulerian graph (where every node has even degree). Then an Eulerian tour is found that is turn converted into a traveling salesman tour by using shortcuts. The graphical representation of this procedure is more complex than the previous ones discussed.

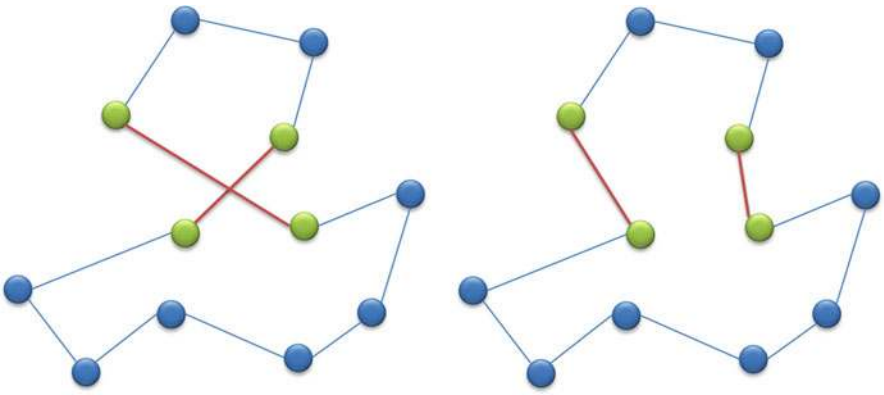
The above TSP heuristics have been applied to a set of TSPs for which the optimal answers are known. The results show that the savings and the insertion heuristics obtain the best results, with an average deviation from optimality of 9.6% and 9.9%, respectively. The nearest neighbor and Christofides heuristics have similar performance with deviations of 18.6% and 19.5%, respectively. The computational effort is similarly small for all four heuristics, making them attractive as the basis for developing more complex search procedures. While in some settings, deviations of almost 10% from optimality may be acceptable, a more precise solution may be required in others. One way of discovering solutions of higher quality is through the combination of construction and local search heuritics.

Local Search Heuristics

Local search methods are based on the notion of neighbor structures that generate changes to move from one solution to another in the solution space. Local searches perform moves as long as the current solution improves and terminate when no further improvement is possible. The resulting solution is said to be locally optimal (i.e., the solution cannot be improved within the neighbor structure under consideration). In the context of the TSP, the most popular moves are the so-called k-opt.

The 2-*opt* procedure consists of replacing two non-adjacent edges by two others that create a tour after the first two are removed. Figure 7 illustrates this move and shows that once the two edges are removed there is only one way to reconnect the two sub-paths in order to create a tour. The move value is the change in the objective function produced by the move. In the case of the TSP, it is the difference between the distances of the added edges minus the distance of the dropped ones.

A natural extension of the 2-opt heuristic is to consider three edges to drop and then relink the resulting sub-paths in the best possible way. The 3-opt heuristic results in seven possibilities for reconnecting the sub-paths, which makes it computationally more expensive than the 2-opt procedure. Several strategies have been proposed to reduce this computational effort. The most successful one consists of building, offline, a candidate subgraph that contains a reduced set of promising edges that will be considered for exchange. There are $\binom{n}{k}$ possible ways to remove k edges in



Heuristics,
Fig. 7 Illustration of a 2-opt move

a tour and $(k-1)!2^{k-1}$ ways to reconnect the tour. This is why only small values of k are usually considered in k-opt heuristics with $k = 2$ and $k = 3$ the ones most commonly used. However, in special situations of large scale TSP instances, $k = 4$ and $k = 5$ have been employed.

The TSP examples illustrate the creativity involved in the development of heuristic procedures that are designed to tackle a specific class of problems. They also show the effectiveness of combining heuristics (e.g., construction and local search), even without the addition of search strategies (e.g., memory or mechanisms to combine solutions) that are typical to metaheuristic methodologies. The combination of heuristics is the notion behind the framework known as hyperheuristics (Burke et al. 2003). While a metaheuristic searches in the solution space, a hyperheuristic is designed to search in the heuristics space. Acknowledging that all heuristics have strengths and weaknesses, the goal of a hyperheuristic is to select the right heuristic in any given situation. Hyperheuristics attempt to automate (through machine learning techniques) what a human would do when faced with a challenging problem and with the knowledge of several applicable heuristics. Additional details on heuristics and their role in solving operational research problems are found in (Silver, et al. 1980) and Silver (2004).

See

- ▶ Algorithm
- ▶ Branch and Bound
- ▶ Greedy Algorithm
- ▶ Hamiltonian Tour
- ▶ Integer and Combinatorial Optimization
- ▶ Knapsack Problem
- ▶ Kruskal's Algorithm
- ▶ Metaheuristics
- ▶ Minimum Spanning Tree Problem
- ▶ *NP*, *NP*-Complete, *NP*-Hard
- ▶ Prim's Algorithm
- ▶ Traveling Salesman Problem

References

Burke, E., Kendall, G., & Newall, J. (2003). Hyper-heuristics: An emerging direction in modern search technology. In F. Glover & G. Kochenberger (Eds.), *Handbook of metaheuristics*. Boston: Kluwer Academic.

Christofides, N. (1976). Worst-case analysis of a new heuristic for the travelling salesman problem. *Report 388*, Graduate School of Industrial Administration, CMU.

Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research, 12*, 568–581.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. *Decision Sciences, 8*, 156–166.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research, 13*, 533–549.

Glover, F., & Kochenberger, G. (Eds.). (2003). *Handbook of metaheuristics*. New York: Kluwer Academic.

Gutin, G., & Punnen, P. (2002). *The traveling salesman problem and its variations*. Boston: Kluwer Academic.

Lawler, E. L., Lenstra, J. K., Rinnoy Kan, A. H. G., & Shmoys, D. B. (1985). *The traveling salesman problem: A guided tour of combinatorial optimization*. New York: Wiley.

Morton, T. E., & Pentico, D. W. (1993). *Heuristic scheduling systems: With applications to production systems and project management*. New York: Wiley.

Silver, E. A. (2004). An overview of heuristic solution methods. *Journal of the Operational Research Society, 55*, 936–956.

Silver, E. A., Victor, R., Vidal, V., & de Werra, D. (1980). A tutorial on heuristic methods. *European Journal of Operational Research, 5*(3), 153–162.

Hidden Markov Models

Yariv Ephraim
George Mason University, Fairfax, VA, USA

Introduction

Hidden Markov models (HMMs) constitute a family of versatile statistical models that have proven useful in many applications. HMMs were introduced in their full generality in 1966 by Baum and Petrie (1966). Baum, Petrie and other colleagues (Baum et al. 1970) at the Institute for Defense Analysis also developed and analyzed a maximum likelihood (ML) procedure for efficient estimation of the HMM parameters from a training sequence. This procedure turned out to be an instance of the now well-known EM (Expectation-Maximization) algorithm of Dempster, Laird and Rubin (1977). A form of HMM, referred to as a Markov Source, was introduced as early as 1948 by Shannon in developing a model for the English language (Shannon 1948).

Baum et al. (1970) referred to HMMs as probabilistic functions of Markov chains. Indeed, an HMM process comprises a Markov chain whose states are associated with some probability distributions. For example, the Markov states may be associated with Poisson probability distributions that differ in their means. At each time instant, a random variable with probability distribution that depends on the state in which the Markov chain lies is generated. An HMM process is thus comprised of a Markov state sequence and an associated sequence of random variables. Normally, only that sequence of random variables is observed while the corresponding sequence of Markov states is not; hence, the term hidden Markov models.

The significance of the states of an HMM varies with the application. For example, in automatic speech recognition, states may represent phonemes of the language. The probability distributions associated with the states may represent statistical variations of the acoustic signals corresponding to the different phonemes. In this application, only the acoustic signal is observed while the phonemes are estimated from the given signal during the recognition process.

Since their introduction in 1966, HMMs have been extensively studied and applied primarily to modeling of speech signals in automatic speech recognition applications. Jelinek and his group at IBM Research Labs proposed a purely statistical HMM-based speech recognition system in the early 1970s (Jelinek 1974). The models were popularized in the early 1980s primarily by Ferguson and his colleagues at the Institute for Defense Analysis (Ferguson 1980) and by Rabiner and his group at AT&T Bell Laboratories (Rabiner 1989). Since then, many advances in the theory and application of HMMs have been introduced. HMMs have been successfully used in Image Recognition, Sonar Signal Processing, Automatic Fault Detection and Monitoring, Speech Enhancement, Communication and Control, Epidemiology and Biometrics and in various Biomedical applications. An excellent survey of the field is contained in the 1996 dissertation of Couvreur. The structure and basic concepts of HMMs are first reviewed and then the Baum algorithm is presented.

Hidden Markov Models

Consider a homogeneous Markov chain of M states. Let $\boldsymbol{\pi} = (\boldsymbol{\pi}_m, m = 1, \ldots, M\}$ denote the vector of initial state probabilities and $\boldsymbol{A} = \{a_{ij}, i, j = 1, \ldots, M\}$ denote the stochastic matrix of state transition probabilities. Let $S_0^N \stackrel{\Delta}{=} \{S_0, S_1, \ldots, S_N\}$, denote the sequence of random variables representing the states of the HMM process at time instants $n = 0, \ldots, N$. A realization of the sequence S_0^N is given by $s_0^N = \{s_0, s_1, \ldots, s_N\}$, where $s_n \in \{1, 2, \ldots, M\}$ for $n = 0, \ldots, N$. The probability that the Markov chain visits state s_n at time n, given that it was in state s_{n-1} at time $n - 1$ is denoted by $a_{s_{n-1}s_n}$. For $n = 0$, $a_{s_{n-1}s_0} = \pi_{s_0}$. Thus, for example, if $s_{n-1} = i$ and $s_n = j$, $0 \le i, j \le M$, then $a_{s_{n-1}s_n} = a_{ij}$ for $n > 0$ and $a_{s_{n-1}s_n} = \pi_j$ for $n = 0$.

Let $Y_0^N \stackrel{\Delta}{=} \{Y_0, Y_1, \ldots, Y_N\}$, denote a sequence of scalar random variables observed at time instants $n = 0, \ldots, N$. The probability distribution of the random variable Y_n given that the HMM is in state S_n is denoted by $P_{Y_n|s_n}(\boldsymbol{y}_n|s_n)$ for $s_n = 1, \ldots, M$. The observable random variables $\{Y_n\}$ may be discrete, continuous or a mixture of both. Moreover, each scalar random variable Y_n may in fact be a vector $\boldsymbol{Y}_n$

of K random variables. In that case, a vector of random variables with conditional probability distribution $P_{\boldsymbol{Y}_n|s_n}(\boldsymbol{y}_n|s_n)$ is emitted whenever state s_n is visited. The vector notation will be used to cover both scalar and vector observable random variables.

The probability density function (PDF), or the probability mass function (PMF), of a sequence of observable random vectors $\boldsymbol{Y}_0^N$ from an HMM process is given next. This PDF (or PMF) is conveniently written using the sequence of states S_0^N, i.e.,

$$\begin{aligned} p_{\boldsymbol{Y}_0^N}(\boldsymbol{y}_0^N) &= \sum_{s_0^N} p_{S_0^N \boldsymbol{Y}_0^N}(S_0^N,\ \boldsymbol{y}_0^N) \\ &= \sum_{s_0^N} p_{S_0^N}(S_0^N)\, p_{\boldsymbol{Y}_0^N|S_0^N}(\boldsymbol{y}_0^N|S_0^N) \\ &= \sum_{S_0^N} \prod_{n=0}^{N} a_{S_{n-1}S_n} p_{\boldsymbol{Y}_n|S_n}(\boldsymbol{y}_n|s_n) \end{aligned} \tag{1}$$

where the Markov property has been invoked in assuming that given the state $S_n = s_n$, the observed random vector $\boldsymbol{Y}_n$ is independent of past and future states and observable random variables.

An important example for the conditional PDF $p_{\boldsymbol{Y}_n|s_n}(\boldsymbol{y}_n|s_n)$ in (1) results when a K-dimensional Gaussian observable random vector $\boldsymbol{Y}_n$ is emitted from a visited state, say $s_n = j$. In that case, $p_{\boldsymbol{Y}_n|s_n}(\boldsymbol{y}_n|s_n = j) = N(\boldsymbol{\mu}_j, \boldsymbol{R}_j)$, where $\boldsymbol{\mu}_j$ and $\boldsymbol{R}_j$ are, respectively, the mean vector and covariance matrix associated with state j. When a single PDF is not sufficient to describe the data associated with a given state, a mixture of probability distributions is useful. For example, a mixture of Gaussian PDFs may be associated with each state of the HMM.

Statistical properties of an HMM process are inherited from properties of its Markov chain (Grimmett and Stirzaker 1995). An irreducible Markov chain is said to be ergodic if all states are positive recurrent and aperiodic. An irreducible ergodic Markov chain is strongly stationary if the initial state probability distribution $\boldsymbol{\pi}$ equals the unique stationary probability distribution of the chain. This stationary probability is obtained from the unique nonnegative solution of the matrix equation $\boldsymbol{\pi} = \boldsymbol{\pi}\boldsymbol{A}$. Stationarity of the Markov chain $\{S_n\}$ implies stationarity of the observable sequence of random variables $\{\mathbf{Y}_n\}$; see Theorem 2.2 in Couvreur (1996). If $\{S_n\}$ is stationary and ergodic, then the observable sequence of random variables $\{\boldsymbol{Y}_n\}$ is ergodic; see Lemma 1 in Leroux (1992).

HMM Parameter Estimation

Modeling of a random process by an HMM involves estimation of the parameters set of the HMM from training data generated by the process. To specify the HMM, one must choose the number of states for the Markov chain, the allowable state transitions and the type of conditional probability distributions of the HMM that best fit the nature of the random process being modeled. When a Markov chain of M states is used, the parameter set of the HMM is given by $\lambda = (\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B})$, where $\boldsymbol{\pi}$ is the M-dimensional vector of initial state probabilities, $\boldsymbol{A}$ is the $M \times M$ matrix of state transition probabilities, and $\boldsymbol{B}$ is the set of parameters of the conditional probability distributions for the various states. If these conditional distributions are Gaussian, $\boldsymbol{B}$ consists of the set of M mean vectors and M covariance matrices.

A random process from which a training sequence $\boldsymbol{Y}_0^N$ is available by an M-state HMM with parameter set λ is shown next. The modeling is commonly performed by

$$\max_{\lambda} \frac{1}{N} \log\ p\left(\boldsymbol{y}_0^N|\lambda\right), \tag{2}$$

where $p(\boldsymbol{y}_0^N|\lambda)$, is the PDF or PMF in (1) evaluated for the training sequence $\boldsymbol{Y}_0^N$, and $\tilde{N} \triangleq K(N+1)$ is the total number of samples in the training data. Note that the subscript $\boldsymbol{Y}_0^N$ from (1) has been dropped, with the dependence of this PDF or PMF on the parameter set λ shown explicitly now. This notation will be adopted henceforth. $\tilde{N}^{-1} \log p(\boldsymbol{y}_0^N|\lambda)$ in (2) is referred to as the normalized log-likelihood function or simply the likelihood function. The estimation approach outlined by (2) is motivated by the Maximum Likelihood (ML) parameter estimation approach. Under certain conditions, the ML estimation approach has optimal asymptotic properties when the training sequence is generated by the HMM whose parameters are being estimated.

Maximization of the likelihood function in (2) is not trivial because the problem is inherently nonlinear. Furthermore, it is easy to see from (1) that evaluation

of the likelihood function requires $(2N + 1)M^{N+1}$ multiplications, since there are M^{N+1} possible state sequences $\{s_0, s_1, \ldots, s_N\}$, and summation is over products of $2(N + 1)$ terms along each such state sequence. This constitutes an exponentially growing number of multiplications as a function of the length of the training sequence. Since N is required to be large to achieve statistical consistency of the estimate of λ, any iterative maximization procedure that requires evaluation of the likelihood cost function will be prohibitively expensive. Fortunately, Baum et al. (1970) developed an efficient estimation procedure for iterative maximization of the likelihood cost function, and an efficient approach for calculating the likelihood function.

The Baum Algorithm

Suppose that an estimate λ_m of the parameter set λ is available at the end of the mth iteration of the Baum algorithm. Let $\tilde{\lambda}$ denote any other estimate of the parameter set λ. Let $L(\lambda_m) \stackrel{\Delta}{=} \tilde{N}^1 \log p\left(y_0^N|\lambda\right)$ denote the likelihood function of the training sequence under the HMM with parameter set λ_m. Then, using Jensen's inequality, Baum et al. (1970) showed that

$$L\left(\tilde{\lambda}\right) - L(\lambda_m) \geq \frac{1}{\tilde{N}}\left[Q\left(\tilde{\lambda},\ \lambda_m\right) - Q(\lambda_m,\ \lambda_m)\right], \quad (3)$$

where

$$Q\left(\tilde{\lambda},\ \lambda_m\right) \stackrel{\Delta}{=} E\left\{\log\ p\left(S_0^N,\ \mathbf{y_0^N}|\tilde{\lambda}\right)|\mathbf{y_0^N},\ \lambda_m\right\} \quad (4)$$

is called the auxiliary function. Clearly, if λ_{m+1} is chosen to be

$$\lambda_{m+1} = \arg\ \max_{\tilde{\lambda}}\ Q\left(\tilde{\lambda},\ \lambda_m\right), \quad (5)$$

then from (3), $L(\lambda_{m+1}) \geq L(\lambda_m)$, since $\tilde{\lambda}$ can always be chosen to be equal to λ_m. Thus, starting with an initial estimate λ_0, and alternating between (4) and (5) results in a sequence of estimates λ_m for the parameter set λ with non-decreasing likelihood values $L(\lambda_m)$. The iterations may be terminated if a fixed point of the algorithm is reached, that is, when $\lambda_{m+1} = \lambda_m$. In that case, $L(\lambda_{m+1}) = L(\lambda_m)$. Equations (4) and (5) constitute the Baum algorithm, or the E-step and M-step, respectively, of the EM algorithm.

It is easy to see that the ML estimate of λ is a fixed point of the algorithm. The algorithm, however, may have many other fixed points that may not even be stationary points of the likelihood function. Convergence of the sequence of estimates λ_m can be established by applying the Global Convergence Theorem to the EM algorithm (Wu 1983).

So far the discussion has focused on modeling of one random process, which is not necessarily an HMM process, by another random process in the form of an HMM. Thus, the training sequence available for estimating the parameter set of the HMM may not be generated by an HMM. In that case, no true HMM parameter set exists, and the quality of the estimate of the HMM parameter set is judged by the performance of subsequent applications. For example, if HMMs are estimated for acoustic signals from various words in a vocabulary, then the performance of a speech recognition system which relies on the estimated HMMs is measured. If, however, a training sequence generated by an HMM is available, then the goal of the estimation procedure is to provide an accurate estimate of the true parameter set of the HMM in some given sense. Thus, if the length of the training sequence is N, and the ML estimate of the parameter set λ is given by $\tilde{\lambda}(N)$, then it is desirable that $\tilde{\lambda}(N) \to \lambda$ as $N \to \infty$ with probability one. An estimator $\tilde{\lambda}(N)$ satisfying this property is called a strongly consistent estimator. One may also be interested, among other properties of the estimator, in the asymptotic distribution of $\tilde{\lambda}(N)$ as $N \to \infty$. Note that these convergence properties of the estimator $\tilde{\lambda}(N)$ are substantially different from those of an instance λ_m of the Baum algorithm. While the first case studied stochastic convergence of the estimator as more and more data become available, the second case studied deterministic convergence of an iterative algorithm for given fixed training data. Baum and colleagues have studied both aspects of convergence; Baum et al. (1970) established similar convergence properties of λ_m to those developed by Wu (1983) for two special models. Baum and Petrie (1966) verified the strong consistency and asymptotic normality of the ML sequence of estimates $\tilde{\lambda}(N)$ for HMMs with discrete observable random variables and stationary and ergodic Markov chains. Strong consistency holds for HMMs with continuous observable random variables under some additional regularity conditions (Leroux 1992). Note that ML

estimation may not be achieved by the Baum algorithm, for example, when the likelihood function has multiple local maxima.

The Re-Estimation Formulas

Maximization of the auxiliary function $Q(\tilde{\lambda}, \lambda_m)$ in (5) results in the so-called re-estimation formulas for the parameter set of the HMM, since they provide a new estimate of the parameter set in term of an old estimate of that set. These re-estimation formulas can be conveniently described using the posterior probabilities $q(s_{n-1}, s_n | \boldsymbol{y}_0^N, \lambda_m)$ of the Markov state pair (s_{n-1}, s_n) calculated from an HMM with parameter set λ_m and the training sequence $\boldsymbol{y}_0^N$. Specifically, maximization of (4) over $\tilde{\pi}$ and $\tilde{a}_{ij}$ for a given λ_m results in the following re-estimation formulas:

$$\pi_j(m+1) = q\big(s_0 = j | \boldsymbol{y}_0^N, \lambda_m\big), \tag{6}$$

$$a_{ij}(m+1) = \frac{\sum_{n=1}^{N} q\big(s_{n-1} = i,\ s_n = j | \boldsymbol{y}_0^N, \lambda_m\big)}{\sum_{j=1}^{M}\sum_{n=1}^{N} q\big(s_{n-1} = i,\ s_n = j | \boldsymbol{y}_0^N, \lambda_m\big)}. \tag{7}$$

For HMMs with Gaussian conditional PDFs, the re-estimation formulas for the mean vector $\boldsymbol{\mu}_j$ and the covariance matrix $\boldsymbol{R}_j$, respectively, are given by

$$\boldsymbol{\mu}_j(m+1) = \frac{\sum_{n=0}^{N} q\big(s_n = j | \boldsymbol{y}_0^N, \lambda_m\big)\boldsymbol{y}_n}{\sum_{n=0}^{N} q\big(s_n = j | \boldsymbol{y}_0^N, \lambda_m\big)} \tag{8}$$

and

$$\boldsymbol{R}_j(m+1) = \frac{\sum_{n=0}^{N} q\big(s_n = j | \boldsymbol{y}_0^N, \lambda_m\big)\Big(\boldsymbol{y}_n - \boldsymbol{\mu}_j(m)\Big)\big(\boldsymbol{y}_n - \boldsymbol{\mu}_j(m)\big)^T}{\sum_{n=0}^{N} q\big(s_n = j | \boldsymbol{y}_0^N, \lambda_m\big)} \tag{9}$$

Note that the posterior probability $q(s_n | \boldsymbol{y}_0^N, \lambda)$ is obtained from summing $q(s_{n-1}, s_n | \boldsymbol{y}_0^N, \lambda)$ over all possible values of s_{n-1}. Efficient recursive calculation of $q(s_{n-1}, s_n | \boldsymbol{y}_0^N, \lambda)$ can be performed by using the forward-backward formulas as shown next.

The Forward-Backward Formulas

Let $F(s_n, \boldsymbol{y}_0^N)$ and $B(\boldsymbol{y}_{n-1}^N | s_n)$, $n = 0, 1, \ldots, N$, denote, respectively, the forward and backward probability functions. The definitions of these functions and their recursive calculation are as follows:

$$F(s_0,\ \boldsymbol{y}_0) \stackrel{\Delta}{=} p(s_0,\ \boldsymbol{y}_0 | \lambda) = \pi_{s_0} p(\boldsymbol{y}_0 | s_0) \tag{10}$$

$$\begin{aligned} F\big(s_n,\ \boldsymbol{y}_0^n\big) &\stackrel{\Delta}{=} p\big(s_n,\ \boldsymbol{y}_0^n | \lambda\big) \\ &= \sum_{s_{n-1}} F\big(s_{n-1},\ \boldsymbol{y}_0^{n-1}\big) a_{s_{n-1}s_n} p(\boldsymbol{y}_n | s_n) \\ &0 < n \le N; \end{aligned} \tag{11}$$

$$\begin{aligned} B\big(\boldsymbol{y}_{N-1}^N | s_N\big) &\stackrel{\Delta}{=} 1, \\ B\big(\boldsymbol{y}_{n-1}^N | s_n\big) &\stackrel{\Delta}{=} p\big(\boldsymbol{y}_{n-1}^N | s_n\big) \\ &= \sum_{s_{n-1}} B\big(\boldsymbol{y}_{n-2}^N | s_{n-1}\big) \alpha_{s_n s_{n-1}} p(\boldsymbol{y}_{n-1} | s_{n-1}), \\ &0 \le n < N. \end{aligned} \tag{12}$$

These relations straightforwardly result by considering the PDF or PMF $p(\boldsymbol{y}_0^N | \lambda)$ given in (1). Note that $F(s_n, \boldsymbol{y}_0^N)$ and $B(\boldsymbol{y}_{n-1}^N | s_n)$ are referred to as the forward and backward probability functions, since strictly speaking they are neither PDFs nor PMFs.

Using the forward and backward probability functions, it can be shown that the desired state posterior probabilities are given by

$$q\big(s_{n-1},\ s_n | \boldsymbol{y}_0^N,\ \lambda\big) = \frac{F\big(s_{n-1},\ \boldsymbol{y}_0^{n-1}\big) B\big(\boldsymbol{y}_{n-1}^N | s_n\big) \alpha_{s_{n-1}s_n} p(\boldsymbol{y}_n | s_n)}{\sum_{s_{n-1},\ s_n} F\big(s_{n-1},\ \boldsymbol{y}_0^{n-1}\big) B\big(\boldsymbol{y}_{n-1}^N | s_n\big) \alpha_{s_{n-1}s_n} p(\boldsymbol{y}_n | s_n)} \tag{13}$$

for $0 < n \le N$, and by

$$q(s_n | N,\ \lambda) = \frac{F\big(s_n,\ \boldsymbol{y}_0^n\big) B\big(\boldsymbol{y}_{n-1}^N | s_n\big)}{\sum_{s_n} F\big(s_n,\ \boldsymbol{y}_0^n\big) B\big(\boldsymbol{y}_{n-1}^N | s_n\big)} \tag{14}$$

for $0 \le n \le N$.

Equations (10)–(14) provide an efficient recursion for calculating the state posterior probabilities required by the re-estimation formulas (6)–(9). Moreover, by definition of the forward probability function,

$$p\left(\mathbf{y}_{\mathbf{0}}^{\boldsymbol{N}}|\lambda\right) = \sum_{s_N} F\left(s_N,\ \mathbf{y}_0^N\right). \tag{15}$$

Thus evaluation of the likelihood function using the forward formula (10) can be performed using only $2NM^2 + M$ multiplications, in contrast with the $(2N - 1)M^{N+1}$ multiplications required by direct calculation of $p(\mathbf{y}_{\boldsymbol{0}}^{\boldsymbol{N}}|\tilde{\lambda})$. Thus, the forward formula enables calculation of the likelihood function with linear rather than exponential complexity as a function of N.

See

- Markov Chains
- Markov Decision Processes
- Markov Processes

References

Baum, L. E., & Petrie, T. (1966). Statistical inference for probabilistic functions of finite state Markov chains. *The Annals of Mathematical Statistics, 37*, 1554–1563.

Baum, L. E., Petrie, T., Soules, G., & Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics, 41*, 164–171.

Couvreur, C. (1996). *Hidden Markov models and their mixtures*, Department of Mathematics, Université Catholique de Louvain, Belgium.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B, 39*, 1–38.

Ferguson, J. D., (Ed.) (1980). In *Proceedings of the symposium on the applications of hidden Markov models to text and speech*. IDA-CRD, Princeton, NJ.

Grimmett, G. R., & Stirzaker, D. R. (1995). *Probability and random processes*. Oxford, UK: Oxford Science Publications.

Jelinek, F. (1974). Continuous speech recognition by statistical methods. *Proceedings of the IEEE, 64*, 532–556.

Leroux, B. G. (1992). Maximum likelihood estimation for hidden Markov models. *Stochastic Processes and Their Applications, 40*, 127–143.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE, 77*, 257–286.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal, 27* (379–423), 623–656.

Wu, C. F. J. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics, 11*, 95–103.

Hierarchical Production Planning

Arnoldo C. Hax
Massachusetts Institute of Technology, Cambridge, MA, USA

Introduction

Production management encompasses a large number decisions that affect several organizational echelons. These decisions can be grouped into three broad categories:

1. Strategic decisions, involving policy formulation, capital investment decisions, and design of physical facilities.
2. Tactical decisions, dealing primarily with aggregate production planning.
3. Operational decisions, concerning detailed production scheduling issues.

These three categories of decisions differ markedly in terms of level of management responsibility and interaction, scope of the decision, level of detail of the required information, length of the planning horizon needed to assess the consequences of each decision, and degree of uncertainties and risks inherent in each decision. These considerations have led to the favoring of a hierarchical planning system to support production management decisions, which guarantees an appropriate coordination of the overall decision-making process but, at the same time, recognizes the intrinsic characteristics of each decision level.

Hierarchical Production Planning

The basic design of a hierarchical planning system includes the partitioning of the overall planning problem, and the linkage of the resulting subproblems.

An important input is the number of levels recognized in the product structure. Three different levels are identified:

1. *Items* are the final products to be delivered to the customers. They represent the highest degree of specificity regarding the manufactured products.

   A given product may generate a large number of items differing in characteristics such as color, packaging, labels, accessories, size, and so on.
2. *Families* are groups of items which share a common manufacturing setup cost. Economies of scale are accomplished by jointly replenishing items belonging to the same family.
3. *Types* are groups of families whose production quantities are to be determined by an aggregate production plan. Families belonging to a type normally have similar costs per unit of production time, and similar seasonal demand patterns.

These three levels are required to characterize the product structure in many batch-processing manufacturing environments. In this section, a hierarchical planning system is proposed based on these three levels of item aggregation.

The first step in the hierarchical planning approach is to allocate production capacity among product types by means of an aggregate planning model. The planning horizon of this model normally covers a full year in order to take into proper consideration the fluctuation demand requirements for the products. The use of a linear- programming model is advocated at this level.

The second step in the planning process is to allocate the production quantities for each product type among the families belonging to that type by disaggregating the results of the aggregate planning model only for the first period of the planning horizon. Thus, the required amount of data collection and data processing is reduced substantially. The disaggregation assures consistency and feasibility among the type and family production decisions and, at the same time, attempts to minimize the total setup costs incurred in the production of families. It is only at this stage that setup costs are explicitly considered.

Finally, the family production allocation is divided among the items belonging to each family. The objective of this decision is to maintain all items with inventory levels that maximize the time between family setups. Again, consistency and feasibility are the driving constraints of the disaggregation process.

Figure 1 shows the overall conceptualization of the hierarchical planning effort.

Aggregate Production Planning for Product Types

Aggregate production planning is the highest level of planning in the production system, addressed at the product-type level. Any aggregate production planning model can be used as long as it adequately represents the practical problem under consideration. The following simplified linear program is considered at this level:

Problem P

Minimize

$$\sum_{i=l}^{I}\sum_{i=l}^{T}\left(c_{it}X_{it}+h_{i,t+L}I_{i,t+L}\right)+\sum_{i=l}^{T}\left(r_lR_l+o_lO_l\right)$$

subject to

$$X_{it}-I_{i,l+L}+I_{i,l+L-1}=d_{i,t+L}\quad i=1,\ \ldots,\ I;\ t=1,\ \ldots,\ T$$

$$\sum_{i=l}^{I}m_tX_{it}=O_t+R_t\qquad t=1,\ \ldots,\ T$$

$$R_t\le(rm)_t\qquad t=1,\ \ldots,\ T$$

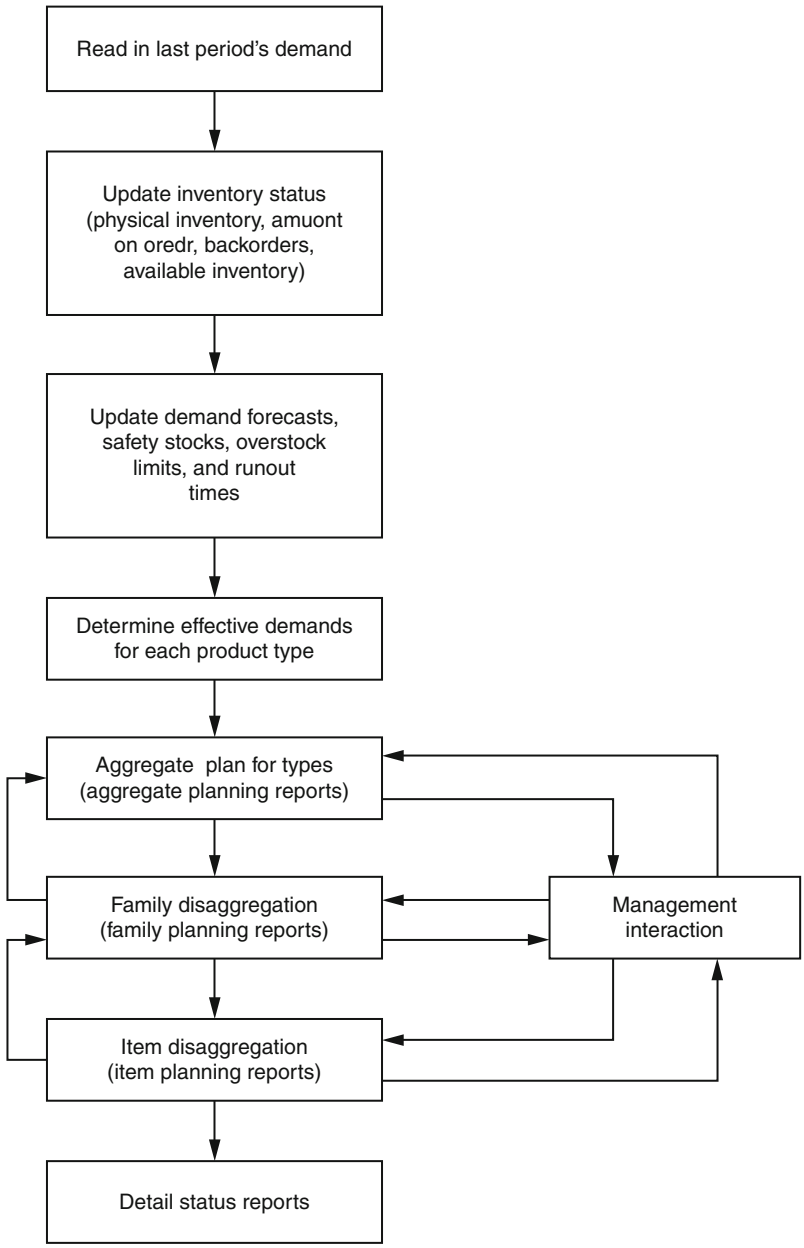
$$O_t\le(om)_t\qquad t=1,\ \ldots,\ T$$

$$X_{it}I_{i,t+L}\ge 0\qquad i=1,\ \ldots,I;\ t=1,\ \ldots,\ T$$

$$R_t,\ O_t\ge 0\qquad t=1,\ \ldots,\ T.$$

The decision variables of the model are: X_{it}, the number of units to be produced of type i during t; $I_{i,t+L}$, the number of units of inventory of type i left over at the end of period $t + L$: and R_t and O_t, the regular hours and the overtime hours used during period t, respectively.

The parameters of the model are: I, the total number of product types; T, the length of the planning horizon; L, the length of the production lead time; c_{it}, the unit production cost (excluding labor); h_{it}, the inventory carrying cost per unit per period; r_t and o_t, the cost per man-hour of regular labor and of overtime labor; $(rm)_t$ and $(om)_t$, the total availability of regular hours and of overtime hours in period t, respectively; and m_i, the inverse of the productivity rate for type i in hours/unit; $d_{i,t+L}$ is the effective demand for type i during period $t + L$.



Hierarchical Production Planning, Fig. 1 Conceptual overview of hierarchical planning system

Because of the uncertainties present in the planning process, only the first time period results of the aggregate model are implemented. At the end of every time period, new information becomes available that is used to update the model with a rolling planning horizon of length T. Therefore, the data transmitted from the type level to the family level are the resulting production and inventory quantities for the first period of the aggregate model. These quantities will be disaggregated among the families belonging to each corresponding type.

The Family Disaggregation Model

The central condition to be satisfied at this level for a coherent disaggregation is the equality between the sum of the productions of the families in a product type

and the amount dictated by the higher level for this type. This equality will assure consistency between the aggregate production plan and the family disaggregation process. This consistency is achieved by determining run quantities for each family that minimize the total setup cost among families.

Bitran and Hax (1977, 1981) proposed the following model for family disaggregation which has to be solved for every product type i and gives rise to a continuous knapsack problem:

Problem P_i

$$\begin{aligned} \text{Minimize} \quad & \sum_{j \in J^0} \left(s_j d_j | Y_j\right) \\ \text{subject to} \quad & \sum_{j \in J^0} Y_j = X_i^* \\ & lb_j \le Y_j \le ub_j \quad \left(j \in J^0\right) \end{aligned}$$

where Y_j is the number of units of family j to be produced; s_j is the setup cost for family j; d_j is the forecast demand (usually annual) for family j; lb_j and ub_j are lower and upper bounds for the quantity Y_j; and X_i^* is the total amount to be allocated among all the families belong to type i. The quantity X_i^* has been determined by the aggregate planning model and corresponds to the optimum value of the variable X_{i1} since only the first-period result of the aggregate model is to be implemented.

The lower bound lb_j, which defines the minimum production quantity for family j, is given by:

$$lb_j = \max\left[0,\ \left(d_{j,1} + d_{j,2} + \cdots + d_{j,L+1}\right) - AI_j + SS_j\right],$$

where $d_{j,1} + d_{j,2} + \ldots + d_{j,L+1}$ is the total forecast demand for family j during the production lead time plus the review period (assumed equal to one); AI_j is the current available inventory for family j (equal to the sum of the physical inventory and the amount on order minus the backorders); and SS_j is the required safety stock. The lower bound lb_j guarantees that any backorder will be caused by forecast errors beyond those absorbed by the safety stock SS_j.

The upper bound ub_j is given by:

$$ub_j = O\ S_j - AI_j,$$

where OS_j is the overstock limit of family j.

The objective function of problem P_i assumes that the family run quantities are proportional to the setup cost and the annual demand for a given family. This assumption, which is the basis of the economic order quantity formulation, tends to minimize the average annual setup cost. Notice that the total inventory carrying cost has already been established in the aggregate planning model; therefore, it does not enter into the current formulation.

The first constraint of problem P_i,

$$\sum_{j \in J^0} Y_j = X_i^*$$

assures the equality between the aggregate model input X^*_i and the sum of the family run quantities.

Initially, J^o contains only those families which trigger during the current planning period. A family is said to trigger whenever its current available inventory cannot absorb the expected demand for the family during the production lead time plus the review period, that is, those families whose current available inventory is such that

$$AI_j < \left(d_{j,1} + d_{j,2} + \cdots + d_{j,1}\right) + SS_j.$$

Equivalently, one can define J^o as containing all those families whose run out times are less than one time period, that is,

$$ROT_j = \frac{AI_j - SS_j}{\sum_{t=1}^{L+1} d_{j,t}} < 1.$$

It is necessary to start production for these families in order to avoid future backorders. All other families are put on a secondary list and will be scheduled only if extra capacity is available. Bitran and Hax (1977) proposed an efficient algorithm to solve problem through a relaxation procedure.

The Item Disaggregation Model

For the period under consideration, all the costs have already been determined in the former two levels, and any feasible disaggregation of a family run quantity has the same total cost. However, the

feasible solution chosen will establish initial conditions for the next period and will affect future costs. To save setups in future periods, one could distribute the family run quantity among its items in such a way that the run out times of the items coincide with the run out time of the family. A direct consequence is that all items of a family will trigger simultaneously. To attain this objective, the use of the following strictly convex knapsack problem is proposed for each family j.

Problem P_j

Minimize

$$\frac{1}{2}\sum_{k\in K^0}\left[\frac{Y_j^*\sum\limits_{k\in K^0}(AI_k - SS_k)}{\sum\limits_{k\in K^0}\sum\limits_{t=1}^{L+1} d_{k,l}} - \frac{Z_k + AI_k - SS_k}{\sum\limits_{t=1}^{L+1} d_{k,l}}\right]^2$$

subject to

$$\sum_{k\in K^0} Z_k = Y_j^*$$

$$Z_k \le OS_k - AI_k$$

$$Z_k \ge \max\left[0,\ \sum_{t=1}^{L+1} d_{k,l} - AI_k + SS_k\right]$$

where Z_k is the number of units to be produced of item k; AI_k , SS_k , and OS_k are, respectively, the available inventory, the safety stock, and the overstock limit of item k; $d_{k,t}$ is the forecast demand for item k in period t; $K^o = \{1, 2, \ldots, j\}$; and, Y_j^* is the total amount to be allocated for all items belong to family j. The quantity Y^* was determined by the family disaggregation model.

The first constraint of problem P_j requires consistency in the disaggregation from family to items. The last two constraints are the upper and lower bounds for the item run quantities. These bounds are similar to those defined for the family disaggregation model in the previous section.

The two terms inside the square bracket of the objective function represent, respectively, the run out time for family j and the run out time for an item k belonging to family j (assuming perfect forecast). The minimization of the square of the differences of the run out times will make those quantities as close as possible. (The term 1/2 in front of the objective function is just a computational convenience.)

For a description of the algorithm recommended to solve this problem, as well as a discussion on performance of the hierarchical production planning model, the reader is referred to Hax and Candea (1984).

See

- ▶ Knapsack Problem
- ▶ Operations Management
- ▶ Production Management

References

Bitran, G. R., Haas, E. A., & Hax, A. C. (1982a). Hierarchical production planning: A single stage system. *Operations Research, 29*, 717–743.

Bitran, G. R., Haas, E. A., & Hax, A. C. (1982b). Hierarchical production planning: A two stage system. *Operations Research, 30*, 232–251.

Bitran, G. R., & Hax, A. C. (1977). On the design of hierarchical production planning systems. *Decision Sciences, 8*, 28–54.

Bitran, G. R., & Hax, A. C. (1981). Disaggregation and resource allocation using convex knapsack problems with bounded variables. *Management Science, 27*, 431–441.

Bitran, G. R., & Tirupati, D. (1993). Hierarchical production planning. In S. C. Graves, A. H. G. Rinnooy Kan, & P. H. Zipkin (Eds.), *Logistics of production and inventory, handbooks in operations research and management science* (Vol. 4, pp. 523–568). New York: Elsevier. Chapter 10.

Hax, A. C., & Candea, D. (1984). *Production and inventory management*. Englewood Cliffs, NJ: Prentice Hall.

Hax, A. C., & Golovin, J. J. (1978a). Hierarchical production planning systems. In A. C. Hax (Ed.), *Studies in operations management*. Amsterdam: North Holland.

Hax, A. C., & Golovin, J. J. (1978b). Computer based operations management system (COMS). In A. C. Hax (Ed.), *Studies in operations management*. Amsterdam: North Holland.

Hax, A. C., & Meal, H. C. (1975). Hierarchical integration of production planning and scheduling. In M. Geisler (Ed.), *TIMS studies in management science* (Logistics, Vol. 1). New York: North Holland/American Elsevier.

Holt, C. C., Modigliani, F., Muth, J. F., & Simon, H. A. (1960). *Planning production inventories and work force*. Englewood Cliffs, NJ: Prentice Hall.

Lasdon, L. S., & Terjung, R. C. (1971). An efficient algorithm for multi-item scheduling. *Operations Research, 19*, 946–969.

Winters, P. R. (1962). Constrained inventory rules for production smoothing. *Management Science, 8*, 470–481.

Higher Education

James C. Hearn[1] and James W. Morrison[2]
[1]University of Georgia, Athens, GA, USA
[2]University of Notre Dame, Notre Dame, IN, USA

Introduction

Until the mid-1960s, rigorous planning techniques and serious attention to campus resource allocations were relatively uncommon in higher education. Expansionist institutions were flush with both students and public support. In the ensuing years, however, academic administrators' interest in better management techniques was heightened by the increasing size and complexity of academic organizations and by persuasive warnings of shrunken public financing and reduced numbers of college-going young people. By the early 1980s, college and university operations were a well-established locus of interest among OR/MS professionals.

Since that time, there has been a continued diffusion of analytic innovations and decision support systems within higher education, and operations research and management science have become increasing important in those settings. Business intelligence software, academic space planning tools, enrollment-management systems, and smart-grid technologies are just a few of the OR/MS applications enabling institutions to streamline operations and improve overall efficiency and effectiveness.

Historical Background

The earliest appearance of modern OR/MS applications in higher education came in the development of formal planning and budgeting models for institutions and systems. The first planning model, CAMPUS (Comprehensive Analytical Methods for Planning University Systems), began in 1964 at the University of Toronto. CAMPUS was part of an attempt to build a computer-based econometric model simulating cost patterns in Canadian universities. Early versions of CAMPUS required extensive data input and placed great demand on computer capacity, making widespread use by a number of colleges and universities virtually impossible. CAMPUS clearly demonstrated, however, the feasibility of developing useful planning and decision-making tools for postsecondary institutions.

Among the first budgeting processes was the planning, programming, and budgeting systems approach (PPBS). PPBS was initially developed by the RAND Corporation for use by the Department of Defense, but was adopted by many higher-education institutions in the early 1960's. The purpose of PPBS was to connect programmatic information to planning and budgetary decision making, and in higher education it represented one of the earliest efforts to tighten couplings among the academy's historically fragmented structures and processes.

In the late 1960s, planning and budgeting models began to proliferate. With funding from the U.S. Office of Education, the National Center for Higher Education Management Systems (NCHEMS) developed RRPM (the Resource Requirements Prediction Model), a computer-simulation model aimed at providing institutions detailed information on costs and resource requirements for establishing and maintaining academic programs. In Europe, two significant planning models emerged: the HIS (Hochschule Information System) in West Germany and the TUSS (Total University Simulation System) in Holland. HIS and TUSS focused mainly on the efficient use of instructional space and certain other aspects of academic operations. As such, they were less ambitious than the North American models of the time, and much less powerful than the models soon to come.

In 1977, Stanford University developed a computer-based financial planning model called TRADES to assist administrators in forecasting income and expenses. True to its name, the TRADES model focused on the trade-offs facing campus leaders. Specifically, it allowed the user to manipulate certain Primary Planning Variables (such as the number of faculty, the number of admitted students, or the level of utility rates), plus approximately 200 other variables to create what-if scenarios for any variety of campus and environmental conditions. Because it was interactive, fast, and extraordinarily comprehensive in scope, the TRADES model represented a distinct advance for the field. Soon, Stanford's Academic

Planning Office produced a generalized version for use in other institutions: GENTRA (GENeralized TRAdes). The TRADES and GENTRA models were the first to handle the large volume of tedious calculations involved in systematic, holistic financial forecasting. As a consequence, the models gave their users greater freedom to focus directly on core issues in institutional planning. In subsequent years, Stanford's pioneering models were both adapted and imitated. Perhaps the best known of the descendant models is EFPM, the EDUCOM Financial Planning Model. Developed in the late 1970s by the EDUCOM consortium of over 350 U.S. colleges and universities, EFPM provided a sophisticated financial projection system.

Beginning in the mid-1990's, developments in computing and online technologies dramatically expanded and diversified OR/MS applications within the higher-education sector (McClure 2005). Institutions' internal management systems were varied and resided on a wide assortment of hardware platforms, operating systems, and databases, however. Responding to the proliferation of incompatible homegrown systems emerging within different administrative units on campuses, universities began to adopt Enterprise Resource Planning (ERP) systems to consolidate and streamline operations and improve end-to-end connectivity (Stevens 2003). As Murphy (2004, p. 29) noted, ERP systems hold potential to "integrate disconnected business operations such as student administration, human resources, and financial systems that have been previously handled by disparate legacy systems, while satisfying the need for real-time, on-demand information.

Although OR/MS applications have been employed for over half a century in higher education, colleges and universities have generally lagged in adoptions of approaches popularized in other organizational sectors. Those lags have both been sensible and restricting. On one hand, fund accounting, systematic asset management, and other innovations have successfully been imported from business to higher education. Yet many applications have achieved only marginal success, including performance-based and zero-based budgeting, total quality management, and process reengineering. Birnbaum (2000) argue persuasively that many of these management and analytic innovations are merely fads borrowed from business or government without full consideration of their limitations within the university's distinctive organizational structures, processes, and cultures. Consequently, imported management approaches often have a limited life cycle and only brief popularity in higher education before abandonment. Inevitably, though, it seems that every failed import is soon followed by the introduction of some other idea novel to the setting.

While imported management processes and systems have had limited success in higher education, universities increasingly rely on OR/MS applications to support core operational functions. In financial and business process management, such applications aid in financial planning and modeling, budget systems, asset management, quality management, enterprise resource planning, business process design, and institutional development and fund raising. OR/MS techniques have been extensively utilized in facilities planning and management in such domains as space planning, energy management, and campus master planning. In institutional effectiveness efforts, OR/MS has been applied in institutional research and assessment, business intelligence and action analytics, strategic planning, enrollment management, and retention management. In risk management, managers have paid growing attention to crisis preparedness and business continuity. And, in research management, regulatory compliance has become a focus of systems development.

The Literature

Literature on the uses of OR/MS approaches in higher education falls into three general categories: 1) general reviews of OR/MS applications in higher education; 2) specialized essays and research reports; and 3) case studies, technical papers, and presentations produced by consultants, vendors, end users, and specialized industry associations. Selected work in each of these arenas merits attention here.

Schroeder (1973) provided an excellent early example of general reviews of OR/MS applications in higher education. The author critically surveyed work on Program Planning Budgeting Systems (PPBS); management information systems (MIS); resource-allocation models; and mathematical models for enrollment planning, faculty staffing, and optimization of resource use. Wilson (1981) provided

a less technical piece in this same vein, with eight short, well-documented articles describing actual and proposed applications. Around the same time, Hussain (1976) and Bleau, (1981b) produced thorough reviews of the packaged planning models being developed then for higher education, including CAMPUS, RRPM, HIS, TUSS, TRADES, and EFPM.

In this period, two classic, comprehensive works on OR/MS applications in higher education appeared: Halstead (1974) and Hopkins and Massy (1981). The Halstead book focused on the planning efforts of state-level postsecondary officials, and paid special attention to forecasting revenues and costs and examining alternative possible uses of scarce state resources. The Hopkins and Massy book, in contrast, addressed central financial issues facing individual institutions. The authors defined financial planning models, outline what these models could reasonably be expected to accomplish, specified how to build the necessary models, and offered historical background on Stanford University's experience in developing and applying planning models. Both of these books became widely cited in the field and are useful references.

Later in the 1980s, improvements in software-based decision-support systems (DSS) made the use of complex planning models more accessible and attractive to administrators not trained in technical applications. Rohrbaugh and McCartt (1986) presented a solid overview of the emerging uses of DSS in higher education at that time, considering such topics as Markov-based decision-support applications, formal decision models, tactical and strategic decision making, system-dynamics simulation models, and alternative approaches for evaluating decision processes. At around the same time, Yancey (1988) provided a comprehensive overview of the statistical methods then being employed by institutional research (IR) offices on campus, and White (1987) provided a useful listing of publicly known OR/MS applications in higher education, classifying extant applications along six dimensions: (1) administrative level, (2) primary purpose of the model, (3) program type, (4) techniques used, (5) resources being allocated, and (6) implementation.

Hoenack and Collins (1990) provided an important contribution focusing theoretically and practically on decision making and planning on campuses. Employing concepts from both economics and OR/MS, the book's contributing authors reviewed thinking on resource allocation, decision processes and priorities, incentive structures, fiscal environments, and cost functions. William Becker, in an especially valuable chapter in the volume, examined the extensive econometric research on students' sensitivity to institutional prices and considered the implications of that research for institutional and system-wide planning efforts. In a less ambitious but similarly framed review, Cheng (1993) explored the impacts of OR techniques on higher-education administration across a variety of functional areas, prominently including resource allocation, budgeting, registration, academic scheduling, and tuition-setting.

As technological developments of the past decade have further lowered costs and expanded information availability on campuses, numerous reviews have focused the uses and analysis of institutional data. Most prominently, McLaughlin et al. (2004) provided a comprehensive overview on the conceptual and theoretical framework for managing institutional data, including the emerging use of innovative technologies to improve data quality, streamline reporting systems, and enhance planning, assessment, and data-driven decision making processes.

Specialized essays and research reports comprise a second category of relevant OR/MS literature. Among the earliest applications of OR/MS in colleges and universities were efforts to improve facilities management, notably including increasing efficiency in buildings and grounds use and maintenance, identifying and analyzing new facility requirements, improving space utilization, examining inventory patterns, projecting demand, and decreasing energy consumption. Facilities planners have to act in the context of an institution's multiple, sometimes conflicting goals of teaching, research, and service, creating a need for sophisticated multi-objective analytic techniques (Ritzman et al. 1979).

While facilities management continues to be a major concern on campuses, it is increasingly incorporated into broader, human-resource modeling, including enrollment management. Large institutions require sophisticated models for scheduling, loading, and controlling students' course enrollments across different physical locations. Similarly, planning for short and long-term faculty and staff needs creates challenges because of ongoing changes in legal and financial conditions and changes surrounding

employment. Before the 1980s, OR/MS approaches to enrollment and employment forecasting largely extrapolated historical data into the near and long-term future (Cox and Jesse 1981), but subsequent approaches have incorporated far more variables and more advanced statistical techniques, including Markov chain methods (Bleau 1981a).

Beyond facilities and human-resource management lie OR/MS-based analyses of many topics relating to quality assessment, assurance, and improvement. Performance issues have long been prominent topics in higher education, and OR/MS work has contributed integrally to quality analysis. Dressell (1976) argued that such techniques as program budgeting, management by objectives (MBO), cost-benefit analysis, MIS, Program Evaluation Review Technique (PERT), and PPBS could provide logical, systematic, comprehensive, and, above all, rational support for evaluation and assessment processes. Although preferred techniques have evolved appreciably over the years, Dressel's early observations proved prescient. In an era in which institutions are being held to increasingly difficult accountability standards, evaluation and assessment have become central important aspects of administration on virtually every campus.

Much of the work on quality assessment, assurance, and improvement in higher education has focused on efficient and effective resource allocation and management, and on the tools needed to achieve those goals. Lee and Van Horn (1983) proposed improving institutional management through joining administration-by-objectives with goal programming, and included useful technical analyses of several resource-allocation modeling approaches. Lewis (1988) reviewed academic-program assessment efforts as of the 1980s and argued that they might be cast more productively in cost-effectiveness terms. His conceptual and empirical work provided directions for later work in that vein, as cost concerns became increasingly pressing.

In the mid-1990s, Geraint and Jill Johnes at Lancaster University's management school focused on funding, pricing, and cost issues in British higher education. G. Johnes (1996) used production theory and multiple-regression and stochastic-frontier techniques to estimate institutions' multi-product cost functions. J. Johnes (1996) explored potential ways to produce comparative indices of institutional performance taking into account substantial differences in inputs. Ryan (2004) pursued similar themes, examining relationships between inputs, in terms of expenditures on varied production factors, and outputs in degree-production terms. His results suggest that relationships between spending and outputs over time, across a wide set of institutions, are positive and merit greater attention in models of student persistence and graduation rates. Lavieri, Puterman and colleagues addressed similar issues in workforce planning: they applied linear programming to Canadian healthcare workforce data to support health human resources (HHR) planning (Lavieri et al. 2008), and used a linear-programming-based hierarchical planning model in pursuit of optimal training, promoting, and recruiting for Canadian nurses over a two-decade period (Lavieri and Puterman 2009).

Work by Welsh and Metcalf (2003) critically examined the institutional effectiveness perspective on quality. Arguing that management efforts to improve quality have consistently failed, despite increased external pressures for change and accountability, the authors maintain that the institutional-effectiveness concept holds the potential to become institutionalized within higher education, rather than another passing fad. Their empirical analysis found that faculty and administrators hold differing perceptions regarding the importance of pursuing institutional effectiveness, owing to ideological and historical factors. They nevertheless contend these resistances may be overcome through careful planning and collaboration, and suggest a variety of OR/MS-related approaches that may prove useful in improving quality assessment and achieving campus-wide support.

Much empirical OR/MS work in higher education has centered on DSS. Often, such work is not published in readily available outlets, but there are exceptions. Stallaert (1997) reported on the design and implementation of a course timetabling system for the management school at UCLA, using sophisticated integer-programming and heuristic algorithms. The author and his dean reported major scheduling efficiency improvements from the initiative (p. 81). Similarly, Darroch and Toleman (2006) explored the implementation of a learning management system in an online education environment for an Australian university. Learning management systems (LMS) are

specialized, integrated software toolsets, developed specifically for the support of online course delivery (Turban et al. 2005). Familiar examples include WebCT and Blackboard. The case-study suggests a number of lessons and recommendations for future LMS implementations. Finally, Maltz et al. (2007) found that a DSS system dramatically improved enrollment management at a small liberal arts college. Both responsiveness and real-time management improved, and knowledgability increased among key administrators, leading to attainment of several strategic enrollment objectives.

Of all the empirical work using OR/MS in higher education since the 1990s, perhaps no approach has consumed greater attention than data envelopment analysis (DEA). This is unsurprising, given the heavy attention to the topic in OR/MS more generally in the preceding years (Seiford 1997; Sarafoglou 1998). DEA is an analytical procedure for measuring the relative efficiency of decision making units performing similar functions with similar goals and objectives. Noting the special characteristics of higher-education systems, institutions, and units, Ahn et al. (1988a) and Ahn et al. (1988b) compared DEA-generated efficiency indexes across sizable samples of public and private U.S. colleges and universities. Somewhat later, Ahn and Seiford (1993) provided an especially useful examination of the history and development of the DEA approach, in the context of further analyses of inter-institutional efficiency.

Tomkins and Green (1988) provided one of the earliest efforts to examine the uses of DEA in academic departments and centers, as opposed to institutions as a whole. Focusing on university accounting departments in the U.K., they cautioned that there are inevitably concerns over data quality in such analyses, and that efforts to incorporate judgments of scholarly performance can raise concerns over subjectivity. Nevertheless, they concluded that comparable efficiency scores could be reliably and productively calculated for academic units.

Shortly after the Tomkins and Green analysis, DEA research on academic departments began to proliferate. Beasley (1990) provided a DEA model for comparing university departments in chemistry and physics departments in the U.K. Sinuany-Stern et al. (1994) produced a useful exploration of a range of academic departments in Israel, and Sarafoglou and Haynes (1996) did the same for business and economic departments in Sweden. Friedman and Sinuany-Stern (1997) added consideration of Canonical Correlation Analysis (CCA) to the use of DEA, aiming to provide full-rank scaling for all units rather than simple classification into efficient and inefficient units alone. Attempting to bridge the gap between the frontier approach of DEA and the mean-tendencies approach of econometrics, the authors noted some problems, but some promise in this approach as well.

Colbert et al. (2000) used DEA to determine the relative efficiency of 24 top ranked U.S. MBA programs. Focusing on output that measured student satisfaction, output that measured recruiter satisfaction, and output that measured both, the authors sought to compare the relative efficiency scores of certain foreign and U.S. MBA programs. The authors note numerous data issues limiting the analysis, but conclude that new rankings based on DEA would "more completely and accurately represent MBA programs [and]. . . make it possible to more fairly compare specific programs" (p. 668).

In an effort to allow effective cross-disciplinary efficiency analyses, Moreno and Tadepalli (2002) used DEA for evaluating the efficiency of a diverse sample of academic departments at a public university in the U.S. Arguing that such analyses are imperative in the increasingly accountability-driven political context of the U.S., the authors suggested that computation of a single summary measure of relative unit efficiency, across different fields, may become central in ongoing debates over resource allocation. They concluded, however, that numerous conceptual and methodological issues require attention before summary measures can be used most effectively on a given campus or across campuses.

Acknowledging such difficulties, Korhonen et al. (2001) focused on the output measurement issues that have constrained earlier work using DEA. These authors sought to combine DEA and decision maker preferences into what they term a Value Efficiency Analysis of academic research performance at universities and research institutes. The analysts suggest that their work defines the efficiency of research units "in the spirit of . . . DEA, complemented with decision maker's . . . preference information" (p. 121). Preferences were obtained by asking leaders to "locate a point on the efficient

frontier having the most preferred combination of input and output values" (p. 121). The authors conclude that this approach has real promise for addressing in difficult issues central to the further development and assessment of disciplinary units in European universities.

J. Johnes (2006) identified and attempted to resolve a methodological problem plaguing earlier DEA analyses at the unit and institutional level. Noting that prior analyses of teaching efficiency have been aggregated across students, which tends to blur individual and organizational contributions, Johnes argued that "analysis at the individual level can give institutions insight into whether it is the students' own efforts or the institution's efficiency which are a constraint on increased efficiency" (p. 443). Her empirical work on a sample of economics graduates and units in U.K. universities supports that conclusion, and casts doubt on the utility of aggregated DEA analyses, which have long been the norm in the field.

The third category of relevant OR/MS literature in higher education is case studies, technical papers, and presentations produced by consultants, vendors, end users, and specialized industry associations. This knowledge base has grown along with the rapid expansion and diversification of commercialized management tools and software. Technology companies and industry experts often produce research to differentiate themselves from their competitors, while institutional administrators increasingly become engaged in specialized learning communities to exchange information and share best practices on emerging topics in the industry. The result has been a growing literature base not reflected in traditional journal and book outlets.

Among the higher-education associations providing relevant OR/MS research and technical reports are the Web-based groups EduCause, the Society for College and University Planning (SCUP), and the Association for Institutional Research (AIR). EduCause is a nonprofit association supporting information-technology professionals in higher education. EduCause's online resource center provides an information repository concerning the use and management of information technology in higher education, including applied research and case studies on OR/MS topics. SCUP and AIR are professional associations for decision makers, planners, and management analysts in higher education. Both provide numerous online and paper resources on OR/MS uses in higher education.

Beyond the associations, individual vendors often produce reports regarding their products' performance, sometimes in conjunction with end users. Such industry-funded efforts must necessarily be evaluated with awareness of the self-interests involved, of course. Unfortunately, there are few independent and objective publicly available analyses of the performance of vendors' products (Birnbaum 2000).

The Complexity of OR/MS in Higher Education

Without question, some institutional leaders perceive OR/MS approaches to be too complex, too user-hostile, or too foreign to elements of their institutions' traditionally decentralized and participatory organizational culture. With that in mind, they resist implementations on the academic side of the business (i.e., curriculum decisions, hiring decisions, admissions decisions). Such views are in keeping with the distinctive organizational characteristics of colleges and universities (Birnbaum 1991). While virtually all large institutions have adopted OR/MS approaches in non-academic areas such as business operations and facilities management, it appears that openness to OR/MS approaches in academic decision making varies widely across institutions. This variation is unsurprising, given the long acknowledged differences among colleges and universities in organizational forms and processes. Earlier work (Baldridge et al. 1978; Baldridge and Tierney 1979) suggests that adoption of business-style techniques in the academic side of campus operations may come most easily to strongly hierarchical institutions, such as community colleges, vocational/technical institutions, and for-profit institutions. Power and control tend to be less centralized in liberal arts colleges and selective universities, and faculty leaders there may resist demands for stepped-up management systems. Pointedly, Birnbaum (2000) has suggested that faculty compliance in such settings may often be more symbolic, virtual, cynical, or self-interested than sincere. And many leaders may simply not pursue implementations of OR/MS techniques in areas touching closely on faculty domains.

But substantial OR/MS work does take place in higher education, and is simply not made widely visible to external audiences. This limited public availability of information regarding OR/MS applications and innovations in campus settings makes definitive conclusions about the state of the field difficult if not impossible. Several factors contribute to this dearth of information.

Notably, the rapid evolution and diversification of commercialized OR/MS applications and the short life-cycle of many management innovations adopted from industry limit the continuity and consistency necessary to collect data and conduct publishable research for traditional professional and academic outlets. Traditional research methodologies often do not align well to these timing constraints. What is more, the public visibility of institutional OR/MS work is constrained by 1) intense competition among vendors, each working to establish competitive advantage in a high-stakes, high-risk environment; 2) managerial and political demands on campuses, including the need to keep certain kinds of analyses in-house for competitive reasons; and 3) the limited professional rewards and outlets for publicly disseminating educationally oriented OR/MS work. The resulting invisibility unquestionably limits broader knowledge regarding the use, and usefulness, of OR/MS approaches.

Locally generated, limited-use techniques have found their way into the administrative portfolios of many institutions. Non-academic institutional units often operate with substantial autonomy under institutions' unique governance and management structures (Birnbaum 2000). Thus, much applied work is done on a small scale (e.g., at the level of a financial-aid office alone, rather than an entire campus), and much of that work remains unknown to those not directly involved. For example, decisions such as choosing among several options for meeting campus heating needs or designing an approach for using residence halls more efficiently are important and are clearly solvable in the OR/MS tradition, but are also constrained in scope. An institution's financial-aid office may choose to deploy business intelligence software as a data management and decision-support system or the university may develop an energy-management policy, but such initiatives may never become widely known on campus or have a significant impact on the institutional core.

Some of the most avid users of sophisticated, albeit localized, OR/MS applications work in offices of institutional research on campuses. Among the tools employed in many such offices are 1) enrollment-management techniques for monitoring and shaping the characteristics of student bodies; 2) models for assessing statistically the relative importance of various student characteristics in predicting whether newly admitted freshmen will register; 3) Markov projections and related predictive models for forecasting enrollments; and 4) student-flow models for analyzing the movement of students into or out of specific programs. Unfortunately, these efforts usually remain unknown to those outside of these specialized offices, unless targeted dissemination efforts are undertaken.

Thus, a variety of factors hamper efforts to estimate overall levels of OR/MS use, and indeed hamper efforts to more fully embed OR/MS in campus life. But, whatever the limitations, there can be no denying that OR/MS techniques and approaches can contribute to the effectiveness and efficiency of colleges and universities as they encounter the multiple opportunities and threats that are ongoing features of their environments. Increasingly, institutions are dependent on good information to frame and support decision making. The emergence of lower-cost computing, improved decision-support systems, expanded telecommunications, and smart-grid technologies all raise expectations that OR/MS applications will continue to play a central role in institutional adaptation and health.

See

- Business Intelligence
- Cost Analysis
- Data Envelopment Analysis
- Decision Support Systems (DSS)
- Total Quality Management

References

Ahn, T., Arnold, V., Charnes, A., & Cooper, W. W. (1988). DEA and ratio efficiency analyses for public institutions of higher learning in Texas. *Research in Governmental and Nonprofit Accounting, 5*, 165–185.

Ahn, T., Charnes, A., & Cooper, W. W. (1988). Some statistical and DEA evaluations of relative efficiencies of public and

private institutions of higher learning. *Socio-Economic Planning Sciences, 22*(6), 259–269.

Ahn, T., & Seiford, L. M. (1993). Sensitivity of DEA to models and variables sets in a hypothesis test setting: the efficiency of university operations. In Y. Ijiri (Ed.), *Creative and innovative approaches to the science of management* (pp. 191–208). New York: Quorum Books.

Baldridge, J. V., Curtis, D. V., Ecker, G., & Riley, G. L. (1978). *Policy making and effective leadership: A national study of academic management*. San Francisco: Jossey-Bass.

Baldridge, J. V., & Tierney, M. L. (1979). *New approaches to management: Creating practical systems of management information and management by objectives*. San Francisco: Jossey-Bass.

Beasley, J. E. (1990). Comparing university departments. *OMEGA The International Journal of Management Science, 18*(2), 171–183.

Birnbaum, R. (1991). *How colleges work*. San Francisco: Jossey-Bass.

Birnbaum, R. (2000). *Management fads in higher education*. San Francisco: Jossey-Bass.

Bleau, B. L. (1981a). The academic flow model: A Markov-chain model for faculty planning. *Decision Sciences, 12*, 294–309.

Bleau, B. L. (1981b). Planning models in higher education: historical review and survey of currently available models. *Higher Education, 10*, 153–168.

Chaffee, E. E. (1985). The concept of strategy: From business to higher education. In J. C. Smart (Ed.), *Higher education: Handbook of theory and research* (pp. 133–172). New York: Agathon Press.

Cheng, T. C. E. (1993). Operations research and higher education administration. *Journal of Educational Administration, 31*(1), 77–90.

Colbert, A., Levary, R. R., & Shaner, M. C. (2000). Determining the relative efficiency of MBA programs using DEA. *European Journal of Operational Research, 125*, 656–669.

Cox, J. F., & Jesse, R. R., Jr. (1981). An application of material requirements planning to higher education. *Decision Sciences, 12*, 240–260.

Darroch, F., & Toleman, M. (2006). Lessons in implementing a learning management system in a university: The academic user perspective. In J. O'Donoghue (Ed.), *Technology supported learning and teaching: A staff perspective* (pp. 261–276). Hershey, PA: Information Science Reference (IGI Global).

Dressel, P. L. (1976). *Handbook of academic evaluation*. Washington: Jossey-Bass 1976.

Friedman, L., & Sinuany-Stern, Z. (1997). Scaling unites via the canonical correlation analysis in the data envelopment analysis context. *European Journal of Operational Research, 100*, 629–637.

Halstead, D. K. (1974). *Statewide planning in higher education*. U.S. Government Printing Office, Washington, DC.

Hoenack, S. A., & Collins, E. L. (Eds.). (1990). *The economics of American universities: Management, operations, and fiscal environment*. Albany, NY: State University of New York Press.

Hopkins, D. S. P., & Massy, W. F. (1981). *Planning models for colleges and universities*. Stanford, CA: Stanford University Press.

Hussain, K. M. (1976). *Institutional resource allocation models in higher education*. Paris: Organization for Economic Cooperation and Development.

Johnes, G. (1996). Multi-product cost functions and the funding of tuition in UK universities. *Applied Economic Letters, 3*(9), 557–561.

Johnes, J. (1996). Performance assessment in higher education in Britain. *European Journal of Operational Research, 89*, 18–33.

Johnes, J. (2006). Measuring teaching efficiency in higher education: An application of data envelopment analysis to economics graduates from UK universities 1993. *European Journal of Operational Research, 174*, 443–456.

Korhonen, P., Tainio, R., & Wallenius, J. (2001). Value efficiency analysis of academic research. *European Journal of Operational Research, 130*, 121–132.

Lavieri, M. S., & Puterman, M. L. (2009). Optimizing nursing human resource planning in British Columbia. *Health Care Management Science, 12*(2), 119–128.

Lavieri, M. S., Regan, S., Puterman, M. L., & Ratner, P. A. (2008). Using operations research to plan the British Columbia registered nurses workforce. *Healthcare Policy, 4*(2), 117–135.

Lee, S. M., & Van Horn, J. C. (1983). *Academic administration: Planning, budgeting, and decision-making with multiple objectives*. Lincoln, NE: University of Nebraska Press.

Lewis, D. R. (1988). Costs and benefits of assessment: A paradigm. In T. W. Banta (ed.), *Implementing outcomes assessment: Promise and perils. New directions for institutional research, 59*, 69–80.

Maltz, E. N., Murphy, K. E., & Hand, M. L. (2007). Decision support for university enrollment management: Implementation and experience. *Decision Support Systems, 44*(1), 106–123.

McClure, P. A. (2005). Managing the complexity of campus information resources. In P. A. McClure (Ed.), *Organizing and managing information resources on your campus* (pp. 1–13). San Francisco: Jossey-Bass.

McLaughlin, G. W., Howard, R. D., Cunningham, L. B., Blythe, E. W., & Payne, E. (2004). *People, processes, and managing data*. Tallahassee, FL: Association for Institutional Research.

Moreno, A. A., & Tadepalli, R. (2002). Assessing academic department efficiency at a public university. *Managerial and Decision Economics, 23*, 385–397.

Murphy, C. (2004). ERP: The once and future king of campus computing. *Syllabus, 17*(7), 29–30. 41.

Neal, J. G., & McClure, P. A. (2003). Organizing information resources for effective management. In P. A. McClure (Ed.), *Organizing and managing information resources on your campus* (pp. 24–44). San Francisco: Jossey-Bass.

Ritzman, L., Bradford, J., & Jacobs, R. (1979). A multiple-objective approach to space planning for academic facilities. *Management Science, 25*, 895–906.

Rohrbaugh, J., & McCartt, A. T., (Eds.). (1986). Applying decision support systems in higher education. *New Directions for Institutional Research, 49*(13).

Ryan, J. F. (2004). The relationship between institutional expenditures and degree attainment. *Research in Higher Education, 45*(2), 97–115.

Sarafoglou, N. (1998). The most influential DEA publications: A comment on Seiford. *Journal of Productivity Analysis, 9*(3), 279–281.
Sarafoglou, N., & Haynes, K. E. (1996). University productivity in Sweden: A demonstration and explanatory analysis for economics and business programs. *The Annals of Regional Science, 30*, 285–304.
Schroeder, R. G. (1973). A survey of management science in university operations. *Management Science, 19*, 895–906.
Seiford, L. M. (1997). A bibliography for data envelopment analysis (1978–1996). *Annals of Operations Research, 73*(1), 393–438.
Sinuany-Stern, Z., Mehrez, A., & Barboy, A. (1994). Academic departments efficiency via DEA. *Computers and Operations Research, 21*, 543–556.
Stallaert, J. (1997). Automated timetabling improves course scheduling at UCLA. *Interfaces, 27*(4), 67–81.
Stevens, C. P. (2003). Enterprise resource planning: A trio of resources. *Information Systems Management, 20*(3), 61–67.
Tomkins, C., & Green, R. (1988). An experiment in the use of data envelopment analysis for evaluating the efficiency of UK university departments of Accounting. *Financial Accountability Management, 4*(2), 147–164.
Turban, E., Aronson, J. E., & Liang, T. P. (2005). *Decision support systems and intelligent systems*. Upper Saddle River, NJ: Prentice-Hall.
Welsh, J. F., & Metcalf, J. (2003). Faculty and administrative support for institutional effectiveness activities: A bridge across the chasm? *Journal of Higher Education, 74*(4), 445–468.
White, G. P. (1987). A survey of recent management science applications in higher education. *Interfaces, 17*(2), 97–108.
Wilson, J. A. (Ed.). (1981). *Management science applications to academic administration* (New directions for higher education, Vol. 35). San Francisco: Jossey-Bass.
Yancey, B. D. (Ed.). (1988). *Applying statistics in institutional research* (New directions for institutional research, Vol. 58). San Francisco: Jossey-Bass.

Hirsch Conjecture

The Hirsch conjecture has a long history in linear programming. For a bounded ($m \times n$) linear-programming problem, the conjecture concerns how many simplex iterations (basis changes) are necessary in going from one extreme point to another. In a 1957 verbal communication with George B. Dantzig, Warren M. Hirsch (a probabilist from New York University who had worked earlier with Dantzig in the Pentagon) asked: "Does there exist a sequence of m or less pivot operations, each generating a new basic feasible solution, which starts with some given basic feasible solution and ends with some other given basic feasible solution, where m is the number of equations?" (Dantzig 1963, p. 160; Dantzig and Thapa 2003, pp. 25, 31, 33, 34). Over the years, there have been many attempts to prove or disprove the Hirsch conjecture; all of them were eventually shown to be false until Francisco Santos, University of Cantabria, Spain, announced and published his paper, "On a counterexample to the Hirsch conjecture, "(Santos 2010; also see De Loera 2011; Ziegler 2011).

In geometric terms, the Hirsch conjecture states that if a polytope (bounded polyhedron) is defined by n linear inequalities in d variables, then the length of the longest shortest path among all possible pairs of vertices (its diameter) should be at most ($n - d$). That is, any two vertices of the polytope may be connected to each other by a path of at most ($n - d$) edges (Santos 2010). Santos showed that the conjecture was false by constructing a 43-dimensional polytope with 86 facets and a diameter greater than 43.

References

Dantzig, G. B. (1963). *Linear programming and extensions*. Princeton, NJ: Princeton University Press.
Dantzig, G. B., & Thapa, M. N. (2003). *Linear programming 2: Theory and extensions*. New York: Springer.
De Loera, J. (2011). New insights into the complexity and geometry of linear optimization. *Optima, 87*(November), 1–13.
Santos, F. (2010). On a counterexample to the Hirsch conjecture. *La Gaceta de la Real Sociedad Matemàtica Española, 13*(3), 525–538.
Ziegler, G. (2011). Comments on new insights into the complexity and geometry of linear optimization. *Optima, 87*(November), 13–14.

Hit-and-Run Methods

Zelda B. Zabinsky[1] and Robert L. Smith[2]
[1]University of Washington, Seattle, WA, USA
[2]University of Michigan, Ann Arbor, MI, USA

Introduction

Hit-and-run is a Markov chain Monte Carlo (MCMC) sampling technique that iteratively generates a sequence of points in a set by taking steps of random length in randomly generated directions. Hit-and-run can be applied to virtually any bounded

region in $\Re^n$, and has nice convergence properties. Hit-and-run can generate a sequence of points that asymptotically approach a uniform distribution on open sets, and modifications of hit-and-run can approximate arbitrary multivariate distributions, including the Boltzmann distribution. The versatility of hit-and-run to approximate an arbitrary distribution makes it useful in a number of settings, including global optimization, identification of redundant constraints, simulation, volume estimation and integration estimation. In addition to converging to a target distribution, a good MCMC sampler will converge quickly from an arbitrary starting point, also known as rapid mixing. The mixing time of the original version of hit-and-run is polynomially bounded for convex sets, as opposed to an exponential mixing time for a ball walk.

Hit-and-run was introduced by Smith (1984) as a way to approximate uniformly distributed points in an open set, but many other uses emerged. Diverse applications of hit-and-run include: identifying non-redundant constraints in linear programs (Berbee et al. 1987); evaluation of multidimensional integrals (Chen and Schmeiser 1996); volume estimation of convex sets (Kannan et al. 1997); statistical model validation; construction of a confidence interval for Bayesian inference; discrete-event simulation (Rubinstein and Kroese 2008), and global optimization (Bertsimas and Vempala 2004; Kalai and Vempala 2006; Mete et al. 2011; Romeijn and Smith 1994; Shen et al. 2007; Zabinsky 2003; Zabinsky et al. 1992, 1993).

Hit-and-run in its simplest form is discussed next, followed by its convergence to a uniform distribution and its mixing time. Then a generalized form of hit-and-run that converges to an arbitrary target distribution is discussed, followed by specific variations and implementation considerations. Next, forms of hit-and-run that operate on discrete or mixed continuous/integer sets are discussed, as the previous algorithms assume that the set to be sampled from is continuous. The final section describes simulated annealing-type algorithms for global optimization that embed hit-and-run as a part of their sampling method.

Definition of Hit-and-Run

Hit-and-run, in its simplest form for a bounded open set S in $\Re^n$, makes a one-step transition from a point $x \in S \subset \Re^n$ to another point $y \in S$ by generating a direction vector uniformly distributed on the surface of a unit hypersphere centered around x, and then generating a point y uniformly distributed on the union of the line segments created by the intersection of a line along the direction vector and S. This line sampling is typically accomplished by employing a one-dimensional rejection method on the line segment intersected by an enclosing box for S.

Hit-and-run generates a sequence of points $\{X_k, k = 0, 1, \ldots\}$ in a bounded open set $S \subseteq \Re^n$ as follows.

Algorithm 1 (Hit-and-Run)

Step 0 Initialize $X_0 \in S$ and set $k = 0$

Step 1 Generate a random direction D_k uniformly distributed over the surface of a unit hypersphere centered around X_k.

Step 2 Generate a random point $X_{k+1} = X_k + \lambda D_k$ uniformly distributed over the line set $L_k = \{x : x \in S \text{ and } \mathrm{x} = \mathrm{X_k} + \lambda \mathrm{D_k}, \lambda \text{ a real-valued scalar}\}$.
If $L_k = \emptyset$, go to Step 1.

Step 3 If a stopping criterion is met, stop. Otherwise increment k and return to Step 1.

The hit-and-run chain has two distinguishing characteristics: (i) it is globally reaching, i.e., it can move from any point $x \in S \subset \Re^n$ to a neighborhood of any other point $y \in S$ in one step, and (ii) it can be implemented easily even when the feasible set S is defined by membership oracles. As Andersen and Diaconis (2007) describe, the algorithm "hits a point on the sphere and runs in that direction."

Smith (1984) proved that hit-and-run converges in total variation to a uniform distribution. Of course, a direct way to sample a point uniformly from S is to enclose it in a box and sample uniformly from the box until a point lands in S. Then this point is exactly uniformly distributed. However, the expected number of points sampled until one lands in S is exponential in dimension, so this is an impractical method for a large-dimensional set. Thus, Markov chain samplers become attractive as a means to approximately sample from a uniform distribution in much less time. Of the MCMC samplers, hit-and-run converges in polynomial time, and is considered to be the most efficient algorithm known to date for generating an

asymptotically uniform point in a convex set (Lovász 1999; Lovász and Vempala 2006).

The analysis of the mixing time of hit-and-run on a convex body in $\Re^n$ by Lovász (1999) assumed that the initial distribution of the Markov chain was not far from uniform, i.e., a 'warm-start'. This assumption was later relaxed, preserving hit-and-run's polynomial efficiency, making it the only known random walk that converges efficiently to a uniform distribution starting from any point inside a convex body (Lovász and Vempala 2006). In contrast, the ball walk takes an exponential time to get out of a corner. Moreover, hit-and-run was also shown to be polynomially efficient for sampling from log-concave distributions over convex bodies.

Some insight into hit-and-run's efficiency is presented by Ghate and Smith (2009), who showed that the network of points and arcs generated by hit-and-run is a small world network in which most nodes are not neighbors of one another, but most nodes can be reached from every other in a small number of steps. Thus another interpretation of hit-and-run is that it generates a small world on the fly.

Given hit-and-run's success at efficiently approximating a uniform distribution, many variations and generalizations have been developed.

Generalizations of Hit-and-Run

The most celebrated Markov chain sampler, introduced by Metropolis et al. (1953), used the idea of an acceptance-rejection step to act as a filter and bias the chain towards a Boltzmann distribution. The original hit-and-run algorithm was extended by Romeijn and Smith (1994) to converge to a target distribution π by adding an appropriate filter, and later further extended using a conditionalization on π to the one-dimensional line segment (Bélisle et al. 1993). Thus, hit-and-run converges to an arbitrary target distribution π in total variation.

Andersen and Diaconis (2007) proposed a generalization of hit-and-run algorithms for MCMC samplers and related it to the Gibbs sampler, Swendsen-Wang block spin dynamics, data augmentation, auxiliary variables, slice sampling, and the Burnside process under a unifying scheme. They describe choosing the point X_{k+1} according to the density π restricted to the line determined by the direction vector, as in Bélisle et al. (1993). The choice of the uniform distribution for direction is replaced by a general choice, and even the concept of a one-dimensional Euclidean line determined by the direction vector is generalized to include subsets of S.

The following algorithm generalizes hit-and-run with a general direction distribution and a Metropolis filter that converges to an arbitrary target distribution π on S, where ν is an absolutely continuous probability distribution defined on the surface of an n-dimensional unit sphere, with density bounded away from zero.

Algorithm 2 (Hit-and-Run for Target Distribution π)

Step 0 Initialize $X_0 \in S$ and set $k = 0$.

Step 1 Generate a random direction D_k from the direction distribution ν on the surface of a unit hypersphere centered around X_k.

Step 2 Generate a candidate point $Z = X_k + \lambda D_k$ uniformly distributed over the line set $L_k = \{x : x \in S \text{ and } \mathrm{x} = \mathrm{X_k} + \lambda \mathrm{D_k},$ λ a real-valued scalar$\}$

If $L_k = \emptyset$, go to Step 1.

Step 3 Accept or reject the candidate point Z with a Metropolis filter for the target distribution π,

$$X_{k+1} = \begin{cases} Z & \text{w.p. } \min\{1, \pi(\mathrm{Z})/\pi(\mathrm{X_k})\} \\ X_k & \text{otherwise.} \end{cases}$$

Step 4 If a stopping criterion is met, stop. Otherwise increment k and return to Step 1.

Note that if π is a uniform distribution, then all candidate points are accepted and Algorithm 1 is a special case of Algorithm 2.

Specific variations and implementations of hit-and-run are discussed next.

Variations and Implementations of Hit-and-Run

Several variations with specific direction distributions and candidate point sampling methods have been studied in the literature.

The most common direction distribution, and one that is readily implemented, is the uniform distribution on the surface of an n-dimensional hypersphere, termed hyperspherical direction (HD) in

Berbee et al. (1987); Zabinsky et al. (1992). It is easily implemented by generating n independent values $d_i, i = 1, 2, \ldots, n$ from a standard normal distribution, $N(0, 1)$ and scaling them to determine the unit direction vector D:

$$D = (d_1, d_2, \ldots, d_n) \left(\sum_{i=1}^{n} {d_i}^2 \right)^{-1/2}. \quad (1)$$

Another natural choice for direction distribution, termed coordinate direction (CD), is a uniform distribution over the n coordinate vectors (spanning $\Re^n$). Both HD and CD versions of direction choice were presented and applied to identifying nonredundant linear constraints in Berbee et al. (1987).

While hit-and-run is guaranteed to converge for a wide class of target distributions π when using the HD choice, the same is not true when using the CD choice. It is possible to construct situations where CD will not converge to π. For example, CD will fail to converge to a uniform distribution on disconnected regions with the property that some points cannot be reached from others along a sequence of coordinate direction moves.

Another modification of the direction choice, introduced by Romeijn et al. (1999), is called a reflection generator. The reflection generator was motivated by the problem of stalling, which may occur if the line intersects a small portion of the feasible set. For example, when the current point x is near a corner of a hypercube, there is a high probability that the next sample point is very close to x, and a very low probability that the next point generated is a substantial distance from x, especially when the number of dimensions is large. This problem is similar to jamming, a well-known problem in nonlinear programming. The reflection generator essentially lengthens the line associated with a chosen direction by reflecting it off the boundaries of the feasible region into the interior. This increases the probability of sampling a point far away from the current point. A general reflection generator is defined in Romeijn et al. (1999), with a straightforward component-by-component reflection implementation. Convergence results are preserved, and positive numerical experience was reported.

Kaufman and Smith (1998) exploited the robustness in direction distribution to accelerate the rate of convergence of hit-and-run. They derived a unique non-uniform direction distribution that optimizes the rate of convergence of hit-and-run to a uniform distribution on a convex set. They used sampled points to fit an ellipsoid to the convex set, and used the parameters of the ellipsoid as bootstrap parameters in the direction distribution to approximate the optimal direction distribution; calling the Markov chain artificial centering hit-and-run.

In addition to variations on choosing the direction distribution in Step 1, there are variations on choosing the random candidate point on the line in Step 2. Theoretically, the point could be chosen according to the target distribution π restricted to the line. However, in practice, this may be computationally difficult to implement. The line sampling is often referred to as step-size distribution. In hit-and-run as stated in Algorithm 1 and Algorithm 2, the step size λ is uniformly distributed on the intersection of the random bidirection with the feasible region. Other variations include a fixed step-size or a variable length interval that can shrink or expand.

A parametrized step-size distribution is used in Ghate and Smith (2009) for solving the Small World problem. The probability density function for the step-size λ is parametrized by a, and is roughly proportional to $(1/|\lambda|^a)$. When $a = 0$, the distribution is the familiar uniform sampling distribution. Ghate and Smith (2009) showed that the expected hitting time for the Small World problem is minimized when the parameter $a = 1$ for the step-size distribution, and that $a = 1$ is the unique choice of a that is scale invariant among all nonnegative values. This parameterized step-size distribution was further explored with hit-and-run in the context of global optimization.

Another consideration in implementing Step 2 is the difficulty in identifying the intersection of the line determined by the random direction, and the feasible set S, even when S is convex. Step 2 can be straightforward to implement if it is possible to determine the points of intersection on the line, i.e., find λ_{min} and λ_{max} such that $X_k + \lambda D_k \in S$ for $\lambda_{min} \leq \lambda \leq \lambda_{max}$. When S is defined by linear inequalities, or analytically invertible functions, the intersection points can be easily expressed (Zabinsky 2003). Then λ can be chosen uniformly over that interval, or according to the conditionalization of π, thus producing the random candidate point.

However, if the feasible region S is nonconvex, and/or the intersection points are not easily determined, then a common implementation is to enclose the feasible set S in a box B, or any regular shape that is easy to determine intersection points, and use a one-dimensional acceptance-rejection scheme to produce the random candidate point.

The following algorithm on an enclosing box B is a modification of hit-and-run with a general direction distribution, as in Bélisle et al. (1993), with details of the one-dimensional acceptance-rejection sampling, as provided in Kiatsupaibul et al. (2011).

Algorithm 3. (Hit-and-Run on a Box)

Step 1 Generate a random direction D_k with direction distribution ν and set $i = 1$.

Step 2 Generate $\lambda_{k,i}$ from the step-size (typically uniform) distribution on

$$R_k = \{r \in \Re : X_k + rD_k \in B\}.$$

Step 3 If $X_k + \lambda_{k,i}D_k$ is not in S, set $i = i + 1$ and return to Step 2. Otherwise, set $Z = X_k + \lambda_{k,i}D_k$.

Step 4 Accept or reject the candidate point Z with a Metropolis filter for the target distribution π,

$$X_{k+1}\begin{cases} Z & \text{w.p}\min\{1,\pi(\mathrm{Z})/\pi(\mathrm{X_k})\} \\ X_k & \text{otherwise} \end{cases}$$

Step 5 If a stopping criterion is met, stop. Otherwise increment k and return to Step 1.

The additional computation due to the one-dimensional acceptance-rejection has been analyzed by Kiatsupaibul et al. (2011) for the case when π is a uniform distribution. They show that the size of the box is not a critical factor to the overall computational effort. More precisely, bounds on the expected mixing time of hit-and-run on a box including all sample points increases only by a linear function of the box diameter (i.e., longest chord in the box).

Another variation to speed up the convergence rate and reduce the number of rejected sample points is to incorporate the shrinking algorithm, also known as a slice sampler, into hit-and-run. The idea is to shrink the interval for selecting λ, as follows.

Algorithm 4. (Hit-and-Run on a Box with Shrinking Step-Size)

Step 0 Initialize $X_0 \in S$ and set $k = 0$.

Step 1 Generate a random direction D_k with direction distribution ν, defining the step-size set as

$$R_k = \{r \in \Re : X_k + rD_k \in B\}.$$

Set $l_1^+ = max_r R_k$ and $l_1^- = min_r R_k$, and set $i = 1$.

Step 2 Generate $\lambda_{k,i}$ from the uniform distribution on the open interval (l_i^-, l_i^+).

Step 3 If $X_k + \lambda_{k,i}D_k$ is not in S, set l_{i+1}^+ and l_{i+1}^- as follows:
if $\lambda_{k,i} > 0$, set $l_{i+1}^+ = \lambda_{k,i}$ and keep $l_{i+1}^- = l_i^-$;
if $\lambda_{k,i} < 0$, set $l_{i+1}^- = \lambda_{k,i}$ and keep $l_{i+1}^+ = l_i^+$.
Then, set $i = i + 1$ and return to Step 2.
Otherwise, if $X_k + \lambda_{k,i}D_k$ is in S, set $Z = X_k + \lambda_{k,i}D_k$.

Step 4 Accept or reject the candidate point Z with a Metropolis filter for the target distribution π,

$$X_{k+1} = \begin{cases} Z & \text{w.p. } \min\{1, \pi(\mathrm{Z})/\pi(\mathrm{X_k})\} \\ X_k & \text{otherwise.} \end{cases}$$

Step 5 If a stopping criterion is met, stop. Otherwise increment k and return to Step 1.

Algorithm 4 differs from Algorithm 3 in that the step-size interval is shrinking. This shrinkage increases the probability of acceptance in Steps 2 and 3. Because every open subset S can still be reached in one step, the convergence property of the new Markov chain remains the same.

When S is convex, the iteration point process generated by Algorithm 4 is the same as that generated by Algorithm 3, so the mixing rate of the two processes is the same. However, when S is not convex, the iteration point processes from the two algorithms distribute differently, and, hence, the mixing rates may be different. Computational results in Kiatsupaibul et al. (2011) suggest that Algorithm 4 is faster than Algorithm 3 when S is not convex.

Other computational results are given in Chen and Schmeiser (1996), where empirical comparisons are made between variations of hit-and-run and other sampling methods including the Gibbs sampler.

Hit-and-run for Discrete and Mixed Continuous/Integer Sets

Given the exceptional performance of hit-and-run on continuous sets in $\Re^n$, it is natural to wonder if it can be extended to discrete sets, or mixed sets in $\Re^n \times Z^m$. The generalized line set in Andersen and Diaconis (2007) with conditions for convergence allows a wide variety of versions that converge to a target distribution. Baumert et al. (2009) provide a detailed definition of discrete hit-and-run (DHR) with mixing times for some specific classes of problems. Mete et al. (2011) introduces a variation on DHR, called pattern hit-and-run (PHR) that is efficiently implemented on both discrete and mixed continuous/integer sets. Both DHR and PHR, summarized next, maintain many of the nice convergence properties of hit-and-run, including polynomial mixing time for some classes of sets.

DHR defines its line set using a bidirectional random walk, called a biwalk. Whereas classical Markov chains such as the nearest neighbor random walk or the coordinate direction random walk fail to converge to a target distribution π on general discrete sets, because they can get trapped in isolated regions of the support set, DHR converges because it retains the global reaching property of hit-and-run.

Consider a finite set S with a membership oracle that is a subset of B given by a a bounded hyper-rectangle intersected with the n dimensional integer lattice $\mathbb{Z}^n$. The third step applies a Metropolis filter with respect to the target distribution to accept or π reject the candidate point and complete the transition of DHR. The DHR algorithm follows.

Discrete Hit-and-Run (DHR)

Step 0 Initialize $X_0 \in S$ and set $k = 0$.

Step 1 Generate a biwalk by generating two independent, nearest neighbor random walks in B that start at X_k and end before they step out of B. The biwalk may have loops but has finite length with probability one. The sequence of points visited by the biwalk is stored in an ordered list.

Step 2 Generate a candidate point Z by choosing a point uniformly distributed from the intersection of the list and S. Note the intersection always contains at least one point, the current point X_k.

Step 3 Accept or reject the candidate point Z with a Metropolis filter for the target distribution π,

$$X_{k+1} = \begin{cases} Z & \text{w.p. } \min\{1, \pi(\mathrm{Z})/\pi(\mathrm{X_k})\} \\ X_k & \text{otherwise.} \end{cases}$$

Step 4 If a stopping criterion is met, stop. Otherwise increment k and return to Step 1.

The reason for employing two independent nearest neighbor walks to define the line set in Step 1 instead of one walk, and for working with the ordered sequence of points in Step 2 as opposed to the set of distinct points visited, is to ensure symmetry of the candidate generator Markov chain. It is easy to construct examples where symmetry fails by employing a single nearest neighbor random walk and/or use the set of distinct points visited (Baumert et al. 2009). The Markov chain of DHR is globally reaching. The global reaching property together with symmetry and other characteristics imply that DHR converges to the target distribution π as desired.

An upper bound on the mixing time of DHR to a uniform distribution is given in Baumert et al. (2009), and polynomial upper bounds for four examples are given. The four examples include: a box within a box, a wedge inside a cube, multiple cubes inside a cube, and isolated yet aligned points within a cube. Note that conventional random walks such as the nearest neighbor random walk and the coordinate direction random walk also mix in polynomial time on the first two examples; however, both of these walks get stuck in isolated regions of S in the third and fourth examples and fail to converge to a uniform distribution. A fifth example given in Baumert et al. (2009) of diagonal points inside a cube only yields exponential bounds for the mixing time, although convergence is still maintained.

The success of discrete hit-and-run with random biwalks inspired the development of pattern hit-and-run for mixed continuous/integer domains. The biwalk in DHR is computationally expensive to implement because each move in the biwalk requires a randomization, and the list associated with the biwalk must be stored to perform the acceptance-rejection step. A more efficient implementation was introduced in Mete et al. (2011), where the biwalk is defined

with the use of patterns, visualized as a repetition of n step-sizes. An advantage to the use of a pattern-generated biwalk is that the pattern is easily generated with only n random number generations, and the acceptance-rejection on the biwalk can be performed by generating a single random number and analytically mapping it to a point on the biwalk. In Mete et al. (2011), two methods for generating patterns are defined; a sphere biwalk and a box biwalk. PHR with either sphere or box biwalk preserves the convergence properties of hit-and-run to a target distribution, and PHR with sphere biwalk converges to continuous hit-and-run as the mesh of the discretized points becomes finer, approaching a continuum. PHR with box biwalk converges to a variation of hit-and-run where the direction distribution is uniform on the surface of a box, instead of the common surface of a hypersphere.

When the feasible set S is ill-structured, the acceptance-rejection on the intersection of the biwalk and S is inevitable. However, a well-structured set of interest is an integer or mixed continuous/integer polytope, as often arises in integer programming or mixed integer linear programming feasible sets. Mete and Zabinsky (2012) remove the inefficiency that arises from rejecting infeasible points, and utilize the linearity of the constraints defining the polytope to directly sample from the intersection of the biwalk and the polytope. This provides an efficient variation of pattern hit-and-run that converges to a target distribution on a discrete or mixed continuous/discrete polytope.

Convergence to π on a general discrete polytope is not simple to attain. For example, a nearest neighbor random walk will not converge to a uniform distribution on a thin polytope that has isolated points without feasible adjacent neighbors. PHR is able to maintain the global reaching property on any polytope by determining all the feasible points on the biwalk, even though they may not be adjacent. Mete and Zabinsky (2012) derive a method to analytically generate a uniform point on the intersection of a biwalk and a discrete polytope by determining the number of feasible points on the biwalk and mapping a uniform point on $[0, 1]$ to a uniform feasible point on the biwalk. They extend the idea to a mixed continuous/discrete lattice of a polytope.

Moreover, PHR converges to a uniform distribution in polynomial time on a class of discrete polytopes; specifically, discrete polytopes that are defined by a finite number of knapsack constraints i.e., $\sum_{j=1}^{n} a_{ij}x_j \leq b_i$ where a_{ij} are nonnegative and b_i are positive for $i = 1, \ldots, m$ and the number of constraints m is independent of the number of dimensions n. This polynomial time performance and the convergence of PHR to hit-and-run on continuous sets suggests the potential efficiency for hit-and-run samplers on a broad class of sets.

Hit-and-Run for Global Optimization

Hit-and-run has been successfully applied to optimization, initially continuous problems, and expanded to mixed continuous/integer problems (Bertsimas and Vempala 2004; Kalai and Vempala 2006; Mete et al. 2011; Romeijn and Smith 1994; Shen et al. 2007; Zabinsky 2003; Zabinsky et al. 1993).

Consider the following global optimization problem:

$$\begin{aligned} &\text{minimize } f(x) \\ &\text{subject to } x \in S \subset B. \end{aligned}$$

An initial application of hit-and-run to optimization was called Improving Hit-and-Run (IHR) by Zabinsky et al. (1993), which modifies Step 3 in Algorithm 2 by simply accepting a candidate point only if it has an improving objective function value, as follows:

Step 3 Complete the transition to X_{k+1} where,

$$X_{k+1} = \begin{cases} Z & \text{if } \mathrm{f(Z)} < \mathrm{f(X_k)} \\ X_k & \text{otherwise.} \end{cases}$$

IHR has been successfully applied to realistic problems (Zabinsky et al. 1992, 2006). The complexity of IHR is, on average, of $O(n^{5/2})$ for a certain class of convex programs (Zabinsky et al. 1993). The direction distribution of IHR on elliptical programs, as defined in Zabinsky et al. (1993), is a multivariate normal distribution with mean zero and covariance matrix equal to the Hessian inverse of the objective function, H^{-1}. If the covariance matrix is the identity matrix, then the direction distribution is simply HD. Although the Hessian is not typically known, the results indicate the ability to guide the direction distribution for better performance.

Romeijn and Smith (1994) embedded hit-and-run into a simulated annealing algorithm and called it Hide-and-Seek. They added acceptance probabilities according to the Metropolis criterion with a temperature parameter T_k so that Step 3 becomes **Step 3**. Accept or reject the candidate point Z according to a Metropolis filter with temperature T_k,

$$X_{k+1} = \begin{cases} Z & \text{w.p. } \min\left\{1, e^{-(f(Z)-f(X_k))/T_k}\right\} \\ X_k & \text{otherwise.} \end{cases}$$

A property of Hide-and-Seek is that it converges to a Boltzmann T distribution, for a fixed temperature (Bélisle et al. 1993). For a general cooling schedule, Romeijn and Smith (1994) showed that if Hide-and-Seek ran long enough at each temperature value to converge to its stationary Boltzmann distribution, then the number of these temperature values would be linear in dimension. This led to an analytically derived adaptive cooling schedule, which was later extended to apply to both continuous and discrete global optimization problems (Shen et al. 2007). The analysis was motivated by the result that a sequence of such Boltzmann distributions achieves a linear complexity on the average number of function evaluations. Hit-and-run embedded as a candidate generator in simulated annealing has both analytical and numerical success.

Even though the acceptance probability for simulated annealing is interpreted as aiding the algorithm to escape local optima, simulated annealing has also been successfully applied to convex programs. Bertsimas and Vempala (2004) and Kalai and Vempala (2006) used hit-and-run as a candidate generator in a simulated annealing-type algorithm for solving convex programs with a membership oracle. In Kalai and Vempala (2006), simulated annealing is shown to converge quickly, and under certain conditions, the Boltzmann distribution is proven to be optimal for annealing on convex problems.

Simulated annealing on finite combinatorial problems has been successful; however, the candidate point generator is specifically chosen for each problem. Pattern hit-and-run, for integer or mixed continuous/integer sets, has been embedded into simulated annealing in Mete et al. (2011) and numerically shown to be very effective on many test problems.

See

- ▶ Global Optimization
- ▶ Markov Chain Monte Carlo
- ▶ Monte Carlo Simulation
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Andersen, H. C., & Diaconis, P. (2007). Hit and run as a unifying device. *Journal de la societe francaise de statistique & revue de statistique appliquee, 148*(4), 5–28.

Baumert, S., Ghate, A., Kiatsupaibul, S., Shen, Y., Smith, R. L., & Zabinsky, Z. B. (2009). Discrete hit-and-run for generating multivariate distributions over arbitrary finite subsets of a lattice. *Operations Research, 57*(3), 727–739.

Bélisle, C. J. P., Romeijn, H. E., & Smith, R. L. (1993). Hitand-run algorithms for generating multivariate distributions. *Mathematics of Operations Research, 18*, 255–266.

Berbee, H. C. P., Boender, C. G. E., Rinnooy Kan, A. H. G., Scheffer, C. L., Smith, R. L., & Telgen, J. (1987). Hit-and-run algorithms for the identification of nonredundant linear inequalities. *Mathematical Programming, 37*, 184–207.

Bertsimas, D., & Vempala, S. (2004). Solving convex programs by random walks. *Journal of the ACM, 51*(4), 540–556.

Chen, M. H., & Schmeiser, B. W. (1996). General hit-and run Monte Carlo sampling for evaluating multidimensional integrals. *Operations Research Letters, 19*, 161–169.

Ghate, A., & Smith, R. L. (2009). A hit-and-run approach for generating scale invariant small world networks. *Networks, 53*(1), 67–78.

Kalai, A. T., & Vempala, S. (2006). Simulated annealing for convex optimization. *Mathematics of Operations Research, 31*(2), 253–266.

Kannan, R., Lovász, L., & Simonovits, M. (1997). Random walks and an $O^*(n^5)$ volume algorithm for convex bodies. *Random Structures and Algorithms, 11*, 1–50.

Kaufman, D. E., & Smith, R. L. (1998). Direction choice for accelerated convergence in hit-and-run sampling. *Operations Research, 46*(1), 84–95.

Kiatsupaibul, S., Smith, R. L., & Zabinsky, Z. B. (2011). An analysis of a variation of hit-and-run for uniform sampling from general regions. *ACM Transactions on Modeling and Computer Simulation (ACM TOMACS), 21*(3), 16:1–16:11.

Lovász, L. (1999). Hit-and-run mixes fast. *Mathematical Programming, 86*, 443–461.

Lovász, L., & Vempala, S. (2006). Hit-and-run from a corner. *SIAM Journal on Computing, 35*(4), 985–1005.

Mete, H. O., Shen, Y., Zabinsky, Z. B., Kiatsupaibul, S., & Smith, R. L. (2011). Pattern discrete and mixed hitand-run for global optimization. *Journal of Global Optimization, 50*(4), 597–627.

Mete, H. O., & Zabinsky, Z. B. (2012). Pattern hit-and-run for sampling efficiently on polytopes. *Operations Research Letters, 40*, 6–11.
Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., & Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics, 21*, 1087–1090.
Romeijn, H. E., & Smith, R. L. (1994). Simulated annealing for constrained global optimization. *Journal of Global Optimization, 5*, 101–126.
Romeijn, H. E., Zabinsky, Z. B., Graesser, D. L., & Neogi, S. (1999). New reflection generator for simulated annealing in mixed-integer/continuous global optimization. *Journal of Optimization: Theory and Applications, 101*(2), 403–427.
Rubinstein, R. Y., & Kroese, D. P. (2008). *Simulation and the Monte Carlo method* (2nd ed.). Hoboken, NJ: Wiley.
Shen, Y., Kiatsupaibul, S., Zabinsky, Z. B., & Smith, R. L. (2007). An analytically derived cooling schedule for simulated annealing. *Journal of Global Optimization, 38*, 333–365.
Smith, R. L. (1984). Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research, 32*, 1296–1308.
Zabinsky, Z. B. (2003). *Stochastic adaptive search for global optimization*. Boston: Kluwer.
Zabinsky, Z. B., Graesser, D. L., Tuttle, M. E., & Kim, G. I. (1992). Global optimization of composite laminate using improving hit and run. In C. A. Floudas & P. M. Pardalos (Eds.), *Recent advances in global optimization* (pp. 343–365). Princeton, NJ: Princeton University Press.
Zabinsky, Z. B., Smith, R. L., McDonald, J. F., Romeijn, H. E., & Kaufman, D. E. (1993). Improving hit and run for global optimization. *Journal of Global Optimization, 3*, 171–192.
Zabinsky, Z. B, Tuttle, M. E., Khompatraporn, C. (2006). A case study: Composite structure design optimization. In J. Pinter (Ed.), *Global optimization: Scientific and engineering case studies* (pp. 507–528). New York: Springer-Verlag.

Homogeneous Lanchester Equations

Simple Lanchester equations with one equation for each side. These equations are used when the weapons for each side are homogeneous in nature (all small-arms) or as a simplified approximation of a heterogeneous situation.

See

► Lanchester's Equations

Homogeneous Linear Equations

A set of linear equations of the form $\boldsymbol{Ax} = \boldsymbol{0}$.

Homogeneous Solution

A solution to the set of equations $\boldsymbol{Ax} = \boldsymbol{0}$. The solution $\boldsymbol{x} = \boldsymbol{0}$ is called a trivial solution, while a solution $\boldsymbol{x} \neq \boldsymbol{0}$ is called a nontrivial solution.

Horn Clause

A logical expression of the form A $\rightarrow$ C, where A (the antecedent) is a simple conjunction of basic (atomic) propositions and C (the consequent) is either null or is a single atomic proposition.

See

► Artificial Intelligence

Hospitals

Yasar A. Ozcan
Virginia Commonwealth University, Richmond, VA, USA

Introduction

Hospitals represented a growing $760 billion industry in the U.S. in 2009 and were responsible at that time for about 32.6% of the nation's health care expenditures. There are 5,815 registered hospitals in U.S., and they have treated 127 million people in emergency departments, admitted 35.1 million for in patient care, and provided 642 million outpatient visits. These hospitals employ 5.4 million professionals or 34.6% of the all health care jobs in U.S. The effect of hospital expenditures on total output in U.S. economy reaches $2.5 trillion (AHA 2011).

There are several types of hospitals: acute care (i.e., defined to be those hospitals with average lengths of stay less than 30 days); psychiatric hospitals; chronic rehabilitation; nursing homes; and Federal (e.g., Veterans Administration Hospitals). The staffed hospital capacity in the U.S. is about 951 thousand beds. The majority of the hospitals are not-for-profit community hospitals (50.3%), and 17% are investor owned (for-profit) community hospitals. Fifty-seven percent of the community hospitals are part of a system, i.e., more than one hospital managed by a central organization. Almost 30% of the community hospitals are part of a network—hospitals, together with physicians and other providers including insurers, coordinate the care delivery in a given community. The major issues linked to the entire hospital industry may be summarized as access or availability, costs, and (broadly defined) quality of care. The key issues and trends associated with these concepts are the following (US Department of Labor 2010):

Access. In 1965, two government-sponsored health insurance plans were created in the U.S., Medicare and Medicaid. The Medicare program provides health insurance for people over 65, the disabled, and people with end-stage renal disease. The Medicaid program provides health insurance for people whose income and resources are below a level established by the individual state-government-sponsored health insurance programs. Despite this, there remain over 51 million uninsured people in the U.S., resulting in a large volume of uncompensated hospital care. In addition, insurance company policies on pre-existing conditions, people changing their jobs, and insurance companies deprive many people access to care. Health care reform in the U.S. has been enacted to overcome some of these access concerns. The Patient Protection and Affordable Care Act and the Health Care and Education Reconciliation Act of 2010 became law in March, 2010. Their implementation, however, will take until 2014 and beyond. By 2019, the uninsured population maybe cut in half, with other changes taking effect sooner, thus easing the access to care for families with children, individuals, people with disabilities, seniors, and young adults.

Costs. Due to spiraling health care costs over the years, cost containment policies have been a focal point for payers, including the U.S. Federal government programs. The introduction of the prospective payment system (PPS) in 1983 created strong incentives for hospitals to reduce costs. Prior to PPS, hospitals were reimbursed on the basis of actual costs plus an allowed return on equity. Following PPS, hospitals have been reimbursed by a fixed fee per patient for each diagnosis-related group (DRG). The DRG is the basic unit of analysis for inpatient hospitalization. The Federal government established DRGs as a way to pay hospitals for Medicare patients; many other payers also use DRGs. Each DRG has a numeric weight or case-severity rating reflecting the national average hospital resource consumption by patients for that DRG compared to the national average resource consumption of all patients. This has forced hospitals to adapt to a new price-competitive environment.

Quality of care. Due to the introduction of PPS, the shift to a price-competition situation, and a fear that hospitals might compromise quality of care to enhance their profitability, there was a surge in efforts to measure, monitor, and improve hospitals' delivered level of quality of care. Part of this effort was also to satisfy the employers who sought relationships with hospitals providing low cost but high quality care. The key outcome measures have typically included mortality rate, infection rate, complication rate, readmission rate (also referred to as adverse outcomes), and functional status. Much of the data analyses involved here is very similar to that used in device and system reliability analyses conducted by operations researchers (Fries 1997; Dhillon 2000).

Efforts towards quality of care created a new discipline called outcomes research. Since patients differ in the severity of their illnesses and the number and nature of medical and social problems they bring with them, there was the recognition of the need for adaptation of risk-adjustment methods (Schwartz et al. 1996). This is quite important in order that outcomes from patients with different severity of illness can be more confidently compared among hospitals.

In the past, many state agencies and private coalitions have used report cards to evaluate risk-adjusted hospital outcomes for specific conditions or procedures, while providing information that patients, employers, and health plans can use to make better decisions. In general, however, hospital leaders view quality-of-care report cards with little enthusiasm and are skeptical about their usefulness. The extent of

lack of knowledge and skepticism appears to vary by hospital size and service volume, ownership type, the method used to generate outcome information (administrative data vs. clinical data), and the time lag between submission of data and publication (Romano et al. 1999).

The U.S. Department of Health and Human Services developed a Web site, Medicare Hospital Compare, where potential patients can explore and compare quality outcomes:

1. Process of care measures in surgical care, heart attack or chest pain, pneumonia, heart failure, and children's asthma;
2. Outcome of care measures including mortality, readmission rate, use of medical imaging;
3. Survey of patients' hospital experiences; and
4. Medicare payment and volume data.

Major Trends and Issues in Hospital Industry

Strongly influenced by PPS, the hospital industry has undergone substantial transformations, most importantly from the rise of managed care plans and the concomitant rise of integrated delivery systems. These areas are of interest to the operations research field because of its ability to assist managers in analyzing systems, designing or redesigning systems, and implementing appropriate changes in systems. Also of interest are the processes that enable efficient and effective delivery modalities. These include innovations to disease management and the delivery process, as well as health information technology (HIT) implementations, of electronic health/medical records.

Innovation in Health Care Delivery. The competitive health market, national quality standards, and outcomes of some delivery processes, will challenge hospitals and health systems to self examine their service lines and redesign their processes and bring innovation and new adaptations to health care delivery. Many hospitals that belong to systems must consider health care delivery issues beyond the hospital. This includes other hospitals in the system or in the market, service line integrations, physicians, health care supply chains, and related entities. An important tool to integrating all these operations is the application of information technology. This will not only help hospital service delivery, but also help to develop a new modality termed the Medical Home—a patient centered delivery system that will coordinate and optimize the health status of each patient (Bradley et al. 2012; Rich et al. 2012).

Evidence based medicine where outcome driven best-practices surface as value to both patients and providers requires redesign of the care delivery process in health care facilities. The goals of redesign processes include elimination of variation in care delivery and produce higher quality outcomes. Hence, hospitals can employ six-sigma, reengineering or lean engineering techniques to achieve their evidence based process delivery goals (Paulus et al. 2008).

As part of redesign effort, the adaptation of Electronic Medical Records (EMR) will alleviate some of the major hospital process and quality issues. The Veterans Administration Hospitals were early adopters of complete EMR and have reaped benefits not only in outcomes, but also in efficient delivery (Ozcan and Luke 2011).

Pay for Performance. Pay-for-Performance (P4P) programs are evolving to include both quality and costs. P4P financially challenges providers to consider to increase quality and safety, thus improve patient outcomes. Accompanying this accountability, is increased transparency on public posting of quality and patient safety data. In addition to these, Accountable Care Organizations (ACOs) may become a reality to allow qualified providers to assume responsibility for overall costs and quality of care for certain populations through bundled payment structures. This may require hospitals to employ more primary care physicians to coordinate the care and enhancing the access. Altogether, the integrated models will have a major role in increasing efficiency, lowering costs, and improving outcomes (Fisher et al. 2007; Shortell et al. 2010).

Role of OR/MS

Specific problem day-to-day hospital management areas in which OR/MS techniques have had some success include: outpatient/ inpatient scheduling; service capacity planning; service demand forecasting; service system design; site location selection; health care supply chains; staffing and

scheduling; strategic planning; and service delivery (Ozcan 2009) Promising possibilities relative to the of access, cost, and quality include:

- Measuring of quality of care on the outpatient side,
- Measuring effectiveness in hospitals, including economies-of-scale and economies of scope. What causes a hospital to be evaluated as efficient—how does it apportion its resources, what procedures and processes does it employ to set itself apart from poorly performing counterparts?
- Predicting hospital risk, early disease diagnosis and treatment, and outcome prediction.
- How does the volume of services in a hospital affect cost and quantity?
- Are the current compliance measures for process of care (in surgical care, hearth attack or chest pain, pneumonia, hearth failure, and children's asthma) reasonable proxies for quality of care assessment? What other measures will are needed to form a more comprehensive outcome portfolio.
- What methods and cultural adaptations or mandates are needed to produce nationwide reports for mishaps in hospital delivery processes. e.g., medication errors, infections, patient falls?
- Measuring the effects of reform efforts.
- What are the impacts of new technologies on efficiency and outcomes?

Concluding Remarks

Only when the answers (or partial insights) to some of the above issues are available will hospital decision makers have the information and tools necessary to make informed tradeoffs and to improve operations both tactically and strategically (Ozcan 2009, Chapter 3). Probably no other area of the economy can benefit more from application of OR/MS than the health care area in general, and the hospital sector in particular.

See

- ▶ Decision Analysis
- ▶ Health Care Management
- ▶ Health Care Strategic Decision Making

References

AHA. (2011). *American Hospital Association*. Chicago, IL: Trend Chartbook.

Bradley, M. G., Weng, W., & Holmboe, E. S. (2012). An assessment of patient-based and practice infrastructure-based measures of the patient-centered medical home: Do we need to ask the patient? *Health Services Research, 47*(1 pt1), 4.

Dhillon, B. S. (2000). *Medical device reliability and associated areas*. Boca Raton, FL: CRC Press.

Fisher, E. S., Staiger, D. O., Bynum, J. P. W., & Gottlieb, D. J. (2007). Creating accountable care organizations: The extended hospital medical staff. *Health Affairs, 26* (Supplement), 44–57.

Fries, R. C. (1997). *Reliable design of medical devices*. New York: Marcel Decker.

Ozcan, Y. A. (2009). *Quantitative methods in health care management: Techniques and applications* (2nd ed.). San Francisco, CA: Jossey-Bass/Wiley.

Ozcan, Y. A., & Luke, R. D. (2011). Healthcare delivery restructuring and productivity change: Assessing the Veterans Integrated Service Networks (VISNs) using Malmquist approach. *Medical Care Research and Review, 68*, 20S–35S.

Paulus, R. A., Davis, K., & Steele, G. D. (2008). Continuous innovation in health care: Implications of the Geisinger experience. *Health Affairs, 27*(5), 1235–1245.

Rais, A., & Vianna, A. (2011). OR in healthcare: A survey. *International Transactions in Operational Research, 18*(1), 1–31.

Rich, E. C., Lipson, D., Libersky, J., Peikes, D. N., & Parchman, M. L. (2012). Organizing care for complex patients in the patient-centered medical home. *Annals of Family Medicine, 10*(1), 60–62.

Romano, P. S., et al. (1999). Grading the graders. How hospitals in California and New York perceive and interpret their report cards. *Medical Care, 37*, 295–305.

Schwartz, M., et al. (1996). A primer: Health care databases, diagnostic coding, severity adjustment systems and improved parameter estimation. *Annals Operations Research, 67*, 23–44.

Shortell, S. M., Casalino, L. P., & Fisher, E. S. (2010). How the center for medicare and medicaid innovation should test accountable care organizations. *Health Affairs, 29*(7), 1293–1298.

U.S. Department of Labor. (2010). *Career guide to industries*, 2010–11 Edition-Health. Washington, DC.

Hundred Percent Rule

Given an optimal basic feasible solution to a linear-programming problem, this rule allows for simultaneous changes in objective function coefficients or right-hand-side values of a linear-programming problem that maintains the optimality of the current

basis. The name comes from the fact that the sum of the ratios of the proposed changes over their respective possible ranges must sum to one or less.

See

- ▶ Sensitivity Analysis
- ▶ Tolerance Analysis

References

Sweeney, D. J., et al. (2009). *Quantitative methods for business*. Mason, OH: South Western Cengage Learning.
Wendell, R. (1992). Sensitivity analysis revisted and extended. *Decision Sciences, 23*(5), 1127–1142.

Hungarian Method

An algorithm for solving the assignment problem that is based on the following version of a theorem that was first stated by the Hungarian mathematician König and later generalized by the Hungarian mathematician Egerváry: if $\boldsymbol{A}$ is a matrix and m is the maximum number of independent zero elements of $\boldsymbol{A}$, then m lines can be drawn in the rows and columns of the matrix that contain all the zero elements of $\boldsymbol{A}$. (A set of elements of a matrix is said to be independent if no two elements lie in the same row or column.)

See

- ▶ Assignment Problem

References

Burkard, R., Dell'Amico, M., & Marterllo, S. (2009). *Assignment problems*. Philadelphia: SIAM.
Kuhn, H. W. (1995). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly, 2*, 83–97.

Hybrid System

A dynamic system that is modeled with a state representation containing both a discrete-valued and continuous-valued component.

See

- ▶ Simulation of Stochastic Discrete-Event Systems

References

Goebel, R., Sanfelice, R. G., & Teel, A. (2009). Hybrid dynamical systems. *IEEE Control Systems Magazine, 29*(2), 28–93.

Hypercube Queueing Model

Richard C. Larson
Massachusetts Institute of Technology, Cambridge, MA, USA

Introduction

The hypercube queueing model was developed in the late 1960s and early 1970s, a period driven by a national commitment to devote scientific energies to the USA's urban ills. The initial application focus for the model was the deployment of urban police patrol cars. Issues that could be examined with the model involved determining appropriate numbers of cars to allocate in each part of the city, spatially deploying the cars to police beats or other territories, and evaluating the impact of alternative dispatch policies. Over the years, the model has been applied to a large number of police departments and ambulance services, and to other services as well, both public and private. This article reviews the history of the model's development, the key ideas of the model, and its implementation, including the evolution and framing of the model and its implementation impact.

Early Work

The hypercube model's roots started with the author's work on the Science and Technology Task Force of President Johnson's Commission on Law Enforcement and Administration of Justice (Government Printing Office 1967) and with MIT-affiliated work with the Boston Police Department. From numerous hours

riding around in the rear seats of police patrol cars and standing behind police radio dispatchers, it was clear that the fleet of police cars in an area of the city can be viewed as spatially distributed servers in a queueing system. Customer inputs to this queueing system are generated by citizens calling 911 and asking for emergency on-scene service. Unlike most multi-server queues, the police queueing system has a heterogeneous pool of servers. Each server faces their own workload situation, dependent of local geography, patterns of customer demand, and workloads of near-by servers. While writing a Ph.D. thesis in 1969, the need for a multi-server queueing model was recognized whose state space retained knowledge of which servers were available and which were busy: "If the state of a server is either 'busy' or 'idle,' then there are 2^N possible states of the system, corresponding to all possible combinations of servers busy and idle. It is convenient to represent a particular state i by a binary number, the 'ones' corresponding to the busy servers and the 'zeros' to idle servers. The state of server 1 is represented by the lth most significant digit. For instance, state $i = 01000\ldots$ corresponds to server $N - 1$ busy and all others idle. State $i = 2^N - 1$ implies that all servers are busy and that a queue may exist." (Larson 1969, p. 124). In the thesis, these hypercube issues are discussed further, and illustrated by numerical examples worked out for small N. But the algorithmic implementation for arbitrary N required further development.

Simultaneously with the thesis, the author sought to confirm the nature of the spatial queueing by conducting a two-week data gathering study in the New York Police Department (NYPD), (Larson 1971). Data were collected by the passenger officer in 54 precinct tours, where a precinct tour is defined to be a full set of operational data from one eight-hour tour or shift gathered over all police vehicles (typically 12 or less) fielded in a precinct or local area police command. While queueing in the usual sense was rare, queueing in the sense of probabilistic congestion was common.

To understand probabilistic congestion, suppose that one lives in police beat A and calls 911 requesting rapid on-scene police response. Further, suppose the police car assigned to beat A is busy with customers a fraction of time ρ_A, representing the utilization factor of the car ostensibly assigned to beat A, the so-called A car. It is assumed that the time the caller needs police service is independent of the real-time status of car A, busy or free. Thus, when calling 911, there is a probability ρ_A that the beat car is currently unavailable for immediate dispatch to the calling address. In that event, the dispatcher will select a near-by car that is available and dispatch that one. Such inter-beat dispatches are sometimes called workload sharing dispatches, because car B, say, will respond when available into beat A and, conversely, car A will occasionally respond into beat B, when needed. In that way, cars A and B share each other's workload. In general, a large number of cars share each other's workload in complex ways. Now suppose that the utilization factors of all cars A, B, C, etc. are all equal, that is, $\rho_A = \rho_B = \rho_C = \ldots = \rho$. In that case, whenever anyone in the service region calls 911, the chance that the responding car will be their beat car will be $1 - \rho$. Consequently, the fraction of dispatches that are inter-beat or workload sharing dispatches is equal to ρ. In urban America, a typical value for in the 1960s was 0.5; in the 2010's, typical values ranged from 0.5 to 0.8.

The results of checking the prediction of this simple aggregate queueing model for inter-beat dispatching for NYPD Division 16, Tour 3, Friday, February 28, 1969, were as follows:

Precinct	Percentage of time unavailable	Percentage of dispatches that are Inter-beat
103	48	55
105	59	57
107	38	48
109	38	37
111	36	48

As can be seen, the extent of inter-beat dispatching is never significantly less than the percentage of time unavailable, and it may be significantly more. There are sound theoretical arguments for suggesting that the simple Poisson model above represents a lower bound on the amount of inter-beat dispatches (Larson 1969).

The results were important for two reasons. First, the percentage of dispatches that are inter-beat dispatches is a useful performance measure of the fielded police force. The officers in each police car are, in theory, supposed to build an identity with the beat to which they are assigned, their patrol beat. This beat identity should cause the officer to feel personally responsible for public order in that beat. However, as it could have been seen empirically and argued theoretically, a patrol car is quite frequently

dispatched to incidents in beats other than its designated beat, a phenomenon known in police circles as flying. The more flying there is, the less the officer builds a strong beat identity. Prior to the research described here, police commanders, in general, had no idea that flying was as rampant as it in fact was. Later, in the 1990s, it was far worse. Second, the results, theoretical and empirical, demonstrated that the fielded police force is a complex spatially distributed queuing system, with the statuses and workloads of the various servers heavily dependent on one another.

Based on these results and the preliminary hypercube models described in Larson (1969), a more general model was developed. Prior to this work, there was very little in terms of analytical guidance for the police planner who wanted to design police beats. A common practice had been to design each beat to have equal internally generated workload. It was thought that equal internal workload would result in equal workloads experienced by the officers in the police cars. Subsequent hypercube developments and related empirical studies showed that this rule-of-thumb can be very wrong (Larson 1974b).

Central Ideas on State, Transition, and Probabilities

State — The hypercube model can be visualized as the corners and edges of a regular cube. For an $N = 3$ police car system, for example, one state of the system could be specified in words: Unit 1 is free or available; Unit 2 is busy; Unit 3 is busy. This state would be depicted by the binary set {0, 1, 1}, which is a corner of the three-dimensional cube. The state {0, 0, 0} represents the situation in which all three units are simultaneously free. The state {1, 1, 1} represents a situation is which all units are simultaneously busy and in which a queue of waiting 911 callers may exist. If there is a queue, the augmentation to the cubic state space may be thought of as an infinitely long tail emanating from state {1, 1, 1}, a situation resembling a Chinese kite.

In generalizing the three-dimensional cube, the analogous figure for an $N = 2$ unit system is a square. For $N > 3$, visualization extends into hyperspace by imagining a unit-volume cube residing in the positive orthant of an N dimensional hyperspace. This is the motivation for calling the model the hypercube model, a model having 2^N states.

Transitions — A state transition occurs whenever a server changes status from free to busy or from busy to free. Each such transition occurs only along a given edge of the hypercube. This requirement imposes the assumption that only one server (e.g., police car) is assigned to each customer, that is, there are no bulk services of customers.

Transitions occur probabilistically. Downward transitions, corresponding to completions of service on customers, occur for server j with rate μ_j. It is assumed that the service time distribution for server j is negative exponential. The rate of upward transitions from a given state to another adjacent state is determined by a complex set of dispatching rules or server assignment policies. Computation of the upward transition rates is a daunting task for a human user of a large system and has to be automated. It is assumed that from each area within the service territory, customers arrive as in a Poisson process, each process in non-overlapping neighborhoods operating independently. Thus, once the set of upward transition rates is known, the process governing upward transitions from any given state is Poisson. Hence, the entire model is a continuous-time Markov model.

State Probabilities — The system performance measures of the hypercube model can be obtained once the limiting probabilistic behavior is determined. To do this requires the computation of the limiting or steady-state probability that the system is operating in some state, $i = 0,1,\ldots, 2^N - 1$, where the hypercube vertices are indexed in some convenient way. This is simply done by employing a balance of flow argument: In the steady state, the probability that the system will enter state i in any small interval of time Δt must equal the probability that the system will exit that state in a time interval of length Δt. That is, inward and outward probability fluxes must be equal. Iif they were not, then there would be a net buildup or builddown of probability in state i, a contradiction to the steady-state hypothesis. The balance-of-flow equations are generated by constructing an N-dimensional sphere around each hypercube vertex, and then equating outward flow to inward flow. In general, there are 2^N equations to solve, with one being redundant and replaced by the condition that the sum of all probabilities must equal unity.

Campbell was the first to create a general computer code for the N-server hypercube model (Campbell 1972). Larson generalized that code, a program written in PL/1, and released it into the public domain in 1975 (Larson 1975b). That version contained several algorithms that sped up the execution time of the model, including an enhanced Gauss-Seidel procedure, a general method for performing a complete unit step tour of the hypercube and more. The code implemented all of the ideas discussed in the first journal paper describing the model (Larson 1974a).

The Physical Assumptions of the Original Model

The original model, called the basic hypercube model, required the following assumptions (Larson 1978; Larson and Odoni 1981):

1. *Geographical atoms*. The area in which the system provides service can be broken down into a number N_A of statistical reporting areas or geographical atoms. These might correspond, for instance, to census blocks, small collections of city blocks, or police reporting areas. In the model, each atom is modeled as a single point located in the center of the atom. Each can also be viewed as a node or vertex of a transportation network over which the servers operate.
2. *Independent Poisson Arrivals*. Each atom is viewed as an independent Poisson generator of customers requiring on-scene service, with the rate λ_j being the Poisson arrival rate from atom j.
3. *Travel times*. Data are available to estimate the mean travel time t_{ij} from each atom i to each atom j. In the absence of such data, plausible approximations for travel times can be made using analytical models and/or transportation network algorithms.
4. *Servers*. There are N spatially distributed servers or response units, each of which can travel to any geographical atom in the service region.
5. *Server locations*. The server location methodology includes both the probabilistic locations of patrolling police cars and the deterministic locations of ambulances. Define a probability l_{nj} = probability that server n is located in atom j at a random time during which server n is known to be free or idle. For an ambulance n with a known fixed location (when idle), there is one l_{nj} having value unity and all other $\{l_{nj}\}$ (for fixed n) equal to zero. For a police car, which may have to patrol several atoms, several $l_{nj} \neq 0$ are assigned, corresponding to the atoms in which the car patrols.
6. *Server assignment*. In response to each customer call, exactly one server is dispatched to the customer, assuming that at least one server is currently available in the service region. If no unit is currently available, there are options of queueing or forwarding the customer to some backup service, for example, a private ambulance service.
7. *Fixed preference dispatching*. Server assignment takes place according to a fixed preference procedure. That is, for each atom there is an ordered list of preferred servers to dispatch to that atom. The dispatcher will search that list in order and always dispatch the first idle server. Usually the list is generated by concerns of geography, such as travel time minimization, but on occasion other concerns such as assigning bilingual personnel could be important.
8. *Service times*. The service time associated with servicing a customer, including travel time, on-scene time, and possible related follow-up time, has a known average value. In general, each server may have its individualized average value. Service-time distribution, as discussed above, is assumed to be negative exponential, an obvious crude approximation in some cases.
9. *Service-time dependence on travel time*. Variations in service times that are due solely to variations in travel time are assumed to be second order compared to variations of on-scene time and related off-scene time.

Given the assumptions above, the model is used to generate a variety of useful performance measures related to server workloads, travel times throughout the service region and disparities among neighborhoods in quality of service received (Larson 1974a, b).

In practice, no actual system will ever conform exactly to all the model's assumptions. There is always a balance to be struck between modeling simplicity and operational reality, with the determining factor being the quality of decisions that can be derived from the model with limited expenditure of effort.

Approximations

In 1973, the author received a telephone call from the New Haven, Connecticut police department. The planners there wanted to use the hypercube model. This was an exciting offer as it was likely to be the first real test-bed application. The hypercube model had been programmed in PL/1 to accommodate up to 15 servers, a limit imposed at the time by the number of bits in a computer word. If New Haven were like New York City or Boston, it would be divided into a number of independently operating precincts or commands, with typically 8 to 12 servers (police cars) in each. In New York City and in many other large U.S. cities, police cars do not routinely cross over precinct boundary lines, so each precinct can be modeled independently with the hypercube model.

For New Haven, however, any police car could be assigned to virtually any address in the city. In fact, for the hypercube model, all of New Haven was one big precinct, with $N = 48$ police cars. This would result in a computational problem of note—an $N = 48$ hypercube model would require the solution to 2^{48} simultaneous linear equations! The curse of dimensionality imposed by the state-space structure of the hypercube model doubled the size of the state space with each additional server.

Motivated to solve the New Haven problem, a simple idea was resorted to: the equations used to compute performance measures suggested that it was not necessary to compute the fine grain 2^N state probabilities in order to evaluate the system performance measures. All that was really needed were the workloads (utilization factors) of the respective units and the dispatch frequencies in the form of the fraction of dispatches that send unit n to atom j. However, the logic behind this sort of argument was wrong because there is an implicit assumption that the units operate independently.

But, in 1975, a probabilistically valid way of dealing with this lack-of-independence problem was derived (Larson 1975a). Using an M/M/N queueing model to represent the aggregate probabilistic behavior of the system, a set of correction factors were developed to make the prior approximation precisely correct for an M/M/N system having a homogeneous pool of servers with a random dispatch policy, and approximately correct for the heterogeneous server system that had to be resolved by the hypercube model.

Armed with the correction factors, one can write a set of N simultaneous nonlinear equations whose solution provides the (approximate) utilization factors of all the N units. The nonlinear equations have a nice geometrically decreasing quality that results in solutions usually within three or four Gauss-Seidel type iterations. From this result, using the correction factors again, the fraction of dispatches that send server n to atom j can be computed. From that, the problem is solved and all required performance measures can be found. For a period of two years, the exact hypercube and the approximate models were run concurrently with numerous different data sets. In almost all of the runs, the approximate model was within about 2% of the exact model's results. This accuracy was judged to be within the modeling accuracy of the exact model. From that point on, It was decided to proceed only with the approximate model in implementations. The results of the use of the approximate model in New Haven were documented in Chelst (1975).

Additional Hypercube Model Applications

Emergency medical services. In Brazil, the hypercube model has been applied and developed further in a number of areas. A key focus has been on the configuration and operation of the emergency medical services on highways, which operate with dispatching policies somewhat different from police and ambulance services in urban settings, (Iannoni and Morabito 2007; Morabito et al. 2008; Iannoni et al. 2009). Their work includes heuristic-based location optimization, dispatch of multiple units, back-up units, non-homogeneous units, plus implementation issues.

Ambulance Location and Relocation. A significant amount of research has been developed using the hypercube model as the physics of the system, while attempting to optimize, usually with respect to home locations of dispatchable servers. Because how busy a server is depends on where the server is located, and location of an ambulance is a decision variable, finding the busy probability of each individual server a priori becomes virtually impossible. Embedding the hypercube model or its approximations in meta-heuristic search methods enables the estimation of individual server busy-probabilities at run time, thereby greatly increasing the location models' realism.

In an effort to increase the accuracy and realism of the Daskin (1983) maximum expected coverage location model (MEXCLP) that uses a system-wide ambulance busy probability (estimated a priori), Saydam and Aytug (2003) developed a genetic algorithm (GA) that combined MEXCLP with an efficient approximation of the hypercube model. Instead of using a system-wide busy probability, this approach enabled location-specific ambulance busy probability estimates. They showed that locations prescribed by the MEXCLP were generally robust, but the corresponding predicted expected coverage could be significantly off. Their hypercube embedded GA model yielded the same solutions as did the MEXCLP for 17 problems, found better solutions for 51 problems, and, for only four of the 72 problems, they were off by a very small margin.

Rajagopalan, Saydam and Xiao (2008) developed the Dynamic Available Coverage Location Problem (DACL), where DACL minimizes the number of ambulances required to cover a city over multiple time periods when demand is fluctuating. The DACL takes into account redeployment of ambulances within a city. It uses the Jarvis (1985) approximation of the hypercube model inside a Reactive Tabu-Search-based Incremental Search Algorithm to calculate the individual server busy probabilities at run time and, thus, ensures that coverage constraints are not violated.

The Minimum Expected Response Location Problem (MERLP) minimizes the expected response time while maintaining coverage requirements (Rajagopalan and Saydam 2009). Minimizing expected response times saves lives, prevents permanent injuries and reduces suffering. In this model, the Jarvis approximation of the hypercube model is embedded in a greedy search algorithm to calculate the expected coverage and to calculate the expected response time using individual server busy probabilities (Jarvis 1985).

Implementations

The hypercube decision technology has been tested extensively (Chelst and Barlach 1981) and generalized over the years, with almost all upgrades due to implementation experience and suggestions; see Larson (1979) for an early detailed technical description. Among notable improvements is inclusion of GPS (Global Positioning Satellite) information or other vehicle locator technologies, supported by a new hypercube dispatch algorithm that assigns the closest real-time available response unit (Larson and Franck 1978).

The hypercube model has been implemented by police departments in many cities, including Hartford, Connecticut; Orlando, Florida (Sacks and Grief 1994); Rotterdam, the Netherlands (Larson and McEwen 1974); Chapel Hill, North Carolina; Dallas, Texas; New York City (Larson and Rich 1987; Larson 1979); and Cambridge, Massachusetts. In Hartford, for instance, the focus was to redesign the spatial deployment of the police cars so that a number of them could be freed from the usual 911 responding force and reassigned to special drug fighting units; this was successfully done in 1991. Using the model, the Orlando Police Department in 1992 essentially redesigned the deployment of its entire force within a project that implemented a new down-town police precinct. The Cambridge, Massachusetts Police Department used the model to demonstrate to city management the deleterious consequences of reducing the size of the force in response to the tax cutting required from Massachusetts Proposition 2 1/2.

The hypercube model has also been implemented by ambulance services in Boston (Brandeau and Larson 1986; Hill et al. 1981; Larson 1982) and New York City. In ambulance deployments, finding nearly optimal locations for the ambulances was a major task. The ambulance services coauthors cited here found the locate-allocate heuristic described in Larson (1979) extremely robust and useful for this task.

See

- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Queueing Theory
- ▶ RAND Corporation

References

Bodily, S. E. (1978). Police sector design incorporating preferences of interest groups for equality and efficiency. *Management Science, 24*, 1301–1313.

Borras, F., & Pastor, J. T. (2002). The ex-post evaluation of the minimum local reliability level: An enhanced probabilistic location set covering model. *Annals of Operations Research, 111*, 51–74.

Brandeau, M., & Larson, R. C. (1986). Extending and applying the hypercube queueing model to deploy ambulances in Boston. In A. Swersey & E. Ignall (Eds.), *Delivery of urban services*. New York: North Holland.

Campbell, G. L. (1972). *A spatially distributed queueing model for police patrol sector design*. S.M. thesis, MIT Press, Cambridge, MA.

Chelst, K. (1975). *Implementing the hypercube model in the New Haven department of police services*. The New York City Rand Institute, R-1566/7.

Chelst, K. (1978). An interactive approach to police sector design. In R. C. Larson (Ed.), *Police deployment, new tools for planners*. Lexington, MA: D.C. Heath.

Chelst, K., & Barlach, Z. (1981). Multiple unit dispatches in emergency services: Models to estimate system performance. *Management Science, 27*, 1390–1409.

Daskin, M. (1983). A maximum expected covering location model: Formulation, properties, and heuristic solution. *Transportation Science, 17*, 48–70.
Government Printing Office. (1967). *Task force report: Science and technology*. President's Commission on Law Enforcement and Administration of Justice, Washington, DC.
Heller, N. (1977). *Field evaluation of the hypercube system for the analysis of police patrol operations: Final report*. St. Louis, MI: The Institute for Public Program Analysis.
Hill, E. D., et al. (1981). *Planning for emergency ambulance service systems, city of Boston*. Department of Health and Hospitals, Boston, MA.
Iannoni, A., & Morabito, R. (2007). A multiple dispatch and partial backup hypercube queuing model to analyze emergency medical systems on highways. *Transportation Research Part E, 43*, 755–771.
Iannoni, A., Morabito, R., & Saydam, C. (2009). An optimization approach for ambulance location and the districting of the response segments on highways. *European Journal of Operational Research, 195*, 528–542.
Jarvis, J. (1975). *Optimization in stochastic service systems with distinguishable servers*. Ph.D. thesis, MIT Press, Cambridge, MA.
Jarvis, J. (1985). Approximating the equilibrium behavior of multi-server loss systems. *Management Science, 31*(2), 235–239.
Larson, R. C. (1969). *Models for the allocation of urban police patrol forces*. Technical report #44, Operations Research Center, MIT Press, Cambridge, MA.
Larson, R. C. (1971). *Measuring the response patterns of New York city police patrol cars*. New York City Rand Institute R-673-NYC/HUD.
Larson, R. C. (1974a). A hypercube queueing modeling for facility location and redistricting in urban emergency services. *Journal of Computers and Operations Research, 1*, 67–95.
Larson, R. C. (1974b). Illustrative police sector redesign in district 4 in Boston. *Urban Analysis, 2*(1), 51–91.
Larson, R. C. (1975a). Approximating the performance of urban emergency service systems. *Operations Research, 23*, 845–868.
Larson, R. C. (1975b). Computer program for calculating the performance of urban emergency service systems: User's manual (Batch processing). *Innovative resource planning in urban public safety systems*, *report TR-14-75*, MIT Press, Cambridge, MA.
Larson, R. C. (Ed.). (1978). *Police deployment: New tools for planners*. Lexington, MA: Lexington Books.
Larson, R. C. (1979). Structural system models for locational decisions: An example using the hyper-cube queueing model. In K. B. Haley (Ed.), *Operational research '78, Proceedings of the eighth IFORS international conference on operations research*, North-Holland, Amsterdam.
Larson, R. C. (1982). Ambulance deployment with the hypercube queueing model. *Medical Instrumentation, 16*, 199–201.
Larson, R. C., & Franck, E. (1978). Evaluating dispatching consequences of automatic vehicle location in emergency services. *Journal of Computers and Operations Research, 5*, 11–30.
Larson, R. C., & Li, V. (1981). Finding minimum rectilinear distance paths in the presence of barriers. *Networks, 11*, 285–304.
Larson, R. C., & McEwen, T. (1974). Patrol planning in the Rotterdam police department. *Journal of Criminal Justice, 2*, 235–238.
Larson, R. C., & McKnew, M. (1982). Police patrol-initiated activities within a system queueing model. *Management Science, 28*, 759–774.
Larson, R. C., & Odoni, A. (1981). *Urban operations research*. Englewood Cliffs, NJ: Prentice-Hall.
Larson, R. C., & Rich, T. (1987). Travel time analysis of New York city police patrol cars. *Interfaces, 17*(2), 15–20.
Li, V. (1977). *Testing the hypercube model in the New York city police department*. S.B. thesis, EE, MIT Press, Cambridge, MA.
McKnew, M. (1978). *The performance of initiated activities and their impact on resource allocation*. Ph.D. thesis, MIT Press, Cambridge, MA.
Morabito, R., Chiyoshib, F., & Galvã, D. (2008). Non-homogeneous servers in emergency medical systems: Practical applications using the hypercube queueing model. *Socio-Economic Planning Sciences, 42*, 255–270.
Rajagopalan, H., & Saydam, C. (2009). A minimum expected response model: Formulation, heuristic solution, and application. *Socio-Economic Planning Sciences, 43*(4), 253–262.
Rajagopalan, H., Saydam, C., & Xiao, J. (2008). A multiperiod set covering location model for dynamic redeployment of ambulances. *Computers and Operations Research, 35*(3), 814–826.
Sacks, S., & Grief, S. (1994). Orlando magic. *OR/MS Today, 21*(1), 30–32.
Saydam, C., & Aytug, H. (2003). Accurate estimation of expected coverage: Revisited. *Socio-Economic Planning Sciences, 37*(1), 69–80.

Hyperexponential Distribution

A continuous random variable is said to be hyperexponential (or mixed exponential) when its probability density function is the convex sum of exponential density functions. The term hyperexponential is due to always having a coefficient of variation greater than 1, which is the coefficient of variation for an exponentially distributed random variable.

See

► Queueing Theory

Hypergame Analysis

A problem structuring method which addresses situations of conflict and cooperation between the independent actors. A key feature is its ability to represent differing perceptions of the situation which may be held by different actors.

See

► Problem Structuring Methods

Hyperplane

A hyperplane in n-dimensional space is defined by the set of vectors $\boldsymbol{X} = (x_1,\ldots, x_n)$ that satisfy a linear function of the form $a_1\, x_1 + \ldots + a_n\, x_n = \boldsymbol{b}$ for given numbers a_j and b. This can be written as $\boldsymbol{ax} = \boldsymbol{b}$, $\boldsymbol{a} = (a_1,\ldots, a_n)$. For $n = 2$, the function defines a line, and for $n = 3$, the function defines a plane.

Identity Matrix

A square matrix $\boldsymbol{A} = a_{ij}$ with $a_{ii} = 1$ and all $a_{ij} = 0$ for $i = j$.

See

► Matrices and Matrix Algebra

IFORS

► International Federation of Operational Research Societies (IFORS)

IFR

Increasing failure rate.

See

► Distribution Selection for Stochastic Modeling
► Failure-Rate Function
► Reliability of Stochastic Systems

IIASA

► International Institute for Applied Systems Analysis (IIASA)

IID

Independent and identically distributed (random variables).

Imbedded Markov Chain

An analysis technique used to analyze a queueing system that is not a continuous-time Markov chain. It appraises the system at selected time points which allow the system to be analyzed via a discrete-parameter Markov chain. The queue length process in the M/G/1 queueing system is not Markovian, but can be analyzed via a Markov chain at service completion time points.

See

► Markov Chains
► Markov Processes
► Queueing Theory

Implementation

R. E. D. Woolsey
Colorado School of Mines, Golden, CO, USA

The Watch It and Model It Approach

The primary assumption of this approach to OR Implementation is that if the OR/MS person is

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

sufficiently educated in the theoretical constructs and methodology that only minimum exposure to the actual situation is required. This assumption often works startlingly well in practice because the graduates of such programs are customarily drawn from the already rich and/or extremely bright quartile of the population. In short, entrance to these schools requires either massive amounts of money or outstanding academic performance which generates a scholarship. The argument is as follows. The product of this approach, when confronted with a real-world problem, could do a memory search from their conceptual education and unerringly choose the proper model for the solution of the problem. It is often implied that the rest is dog work which can be safely handed to others.

The good news about this approach is that if the OR/MS person is quite bright, quick and politically aware, excellent results usually obtain in spite of lack of knowledge of the system. Any tailoring of the process again is accomplished quickly due to the acuteness of the intellect of the person. It must also be pointed out that the customer is often sufficiently in awe of the educational, cultural, and economic background of the consultant that Gestalt psychology plays no small part in acceptance of models. This approach has been found to be particularly effective in strategic and high-level corporate planning with correspondingly high acceptance by top management. Another way to characterize situations where this approach does well is to say that the less measurable the results, the better the acceptance.

The bad news about this approach is that it almost uniformly fails in the tactical world. Manufacturing managers are justly famous for having little time for academic experts with no shop floor experience. This often supports the argument about how little OR/MS has actually been used in the manufacturing workplace as opposed to the corporate levels mentioned above. The time it takes to accomplish a Ph.D. militates strongly against a person having also the shop floor experience in a manufacturing situation. A story going the rounds of the profession is of interest here. It is alleged that a Lanchester Prize winner for nonlinear optimization was suddenly thrown out of work by the Army pulling the monetary plug on his particular Beltway Bandit employer. He well knew that refineries had a multitude of nonlinear problems in the production of hydrocarbons. He therefore hide himself off to the nearest refinery and offered his services to the refinery manager to solve his nonlinear refinery optimization problems. He named a price for his services, and the refinery manager then asked him how much he knew about chemical engineering. When he confessed his total ignorance in this area, the refinery manager politely asked him how much he was prepared to pay the oil company for him to learn enough about chemical engineering to help them! With this cautionary tale, the difference between conceptual excellence and practical reality is addressed next.

The principal reason for failure at the tactical level is that the customer must perceive that the consultant knows enough about their area so that the consultant

(a) Understands that politics wins over optimality all the time and
(b) That such knowledge will create respect for what they have to put up with

Further, people at the tactical level of companies are not impressed by anyone unless the latter have gone through the same boot camp learning process that they have. In short, the author believes that an education from an eminent institution may be actually more of something to be overcome than an asset in the milieu of tactics.

The Get Down and Do It Approach

It is the author's custom to encourage implementation in OR by attending conventions and asking the presenters the following questions.

1. Did you know what was happening on the project before you modeled it?
   If the answer is yes, they are then politely asked:
2. How do you know?
   The only acceptable answer is that the presenter found out by doing the work being modeled under the conditions of the people who are presently doing it until they had enough confidence in the presenter to take a day off and let the presenter do it alone. Anyone that believes that one can learn enough by watching should be treated with the amusement they deserve.
   The next question is:
3. Did they accept and use your model?
   If the answer is yes, the author proceeds to the last question which is:

4. Do you have measurable results in, for example, dollars at present worth, after tax, adjusted for inflation?

It is the author's opinion that people who answer no to any of the above questions have failed the test of operations research implementation.

See

- ▶ Field Analysis
- ▶ Implementation of OR/MS in the Public Sector
- ▶ Practice of Operations Research and Management Science

Implementation of OR/MS in the Public Sector

Kenneth Chelst
Wayne State University, Detroit, MI, USA

Introduction

The successful use and impact of OR/MS in the public sector varies by domain. Areas of continuing operational and policy impact include the military, energy administration, and environmental quality. With forest land management, OR/MS successes extend to all corners of the globe: U.S., Chile, and New Zealand. When policies are debated with regard to illicit drugs, prison populations, homeland security, or air traffic safety, the leading experts are operations researchers who have made a long-term commitment to studying the issues. In contrast, the growing list of city government success stories of the 1970s has now become an infrequent event. With regard to social welfare and educational policy, there are no significant successes to speak of in the U.S. in part because operations researchers have not even attempted to penetrate these areas.

The OR/MS literature of the 1960s and 70s began to document a growing concern about implementation failure of many OR/MS models and studies in business. The primary goal of these papers was to raise the level of awareness of practitioners and cause them to think beyond the technical validity and sophistication of their models. They needed to recognize that OR/MS model implementation often engenders organizational change. OR/MS practitioners were encouraged to view factors affecting model implementation as part of the broader framework of the difficulty of change management. In this article, even more complex implementation challenges faced by OR/MS modelers in the public sector will be explored.

In the context of this article, the term public sector covers primarily governmental agencies and services at every level from local to federal. The focus is mainly on the U.S. perspective, but also includes the greater role OR/MS has played in Great Britain and in the developing world. The term public sector can also be applied to hospitals, health insurance, other health care providers, as well as public utilities. The services provided are of such public importance that in many countries these organizations and systems are government run. For example, in Great Britain, there is one health care system that provides and pays for services. It was then possible for an OR/MS team to plan and help successfully launch a medical assistance hotline called National Health Service (NHS) Direct (Royston et al. 2003). In the U.S., health care providers, insurance companies, and power generators are quasi-public and in some cases regulated monopolies that receive special oversight from all levels of governmental. Thus, the factors that affect implementation are not significantly different from that of the private sector. If anything, legal and regulatory restrictions on major decisions encourage the use of quantitative analysis and models. For example, major power plant location decisions and related rate increases require formal analysis and hearings to justify and assess the impact of these decisions. Similarly, hospital capacity expansion in the U.S. has long been governed by a requirement to prove a need. OR/MS models with explicit assumptions and logic can be valuable tools in these public presentations.

Public Sector Implementation: Definition

OR/MS models play three distinct roles in the public sector and successful implementation has a different meaning in each context. These are:

1. Operating efficiency and effectiveness,
2. Evaluating major policy initiatives, and
3. Public debates

In the first category, successful implementation has the same goal as in the private sector. Do managers of the governmental agency use the model to make decisions or manage and improve operations? In the second category, agencies such as the Federal Energy Agency, U.S. Forestry Service, or the Environmental Protection Agency are charged with assessing the impact of changes in policy, budgeting, regulation, and law. Successful model implementation means that as issues arise and alternatives are considered, models are run and their results play an important part in the internal and external discussions. In essence, model implementation means the OR/MS study fits Little's characterization of model success as "updating the intuition of decision makers," (Little 1970), or agrees with Murphy's goal of "communicate core insights," (Murphy 1991). The fact that politicians may choose to override the results of the analysis is not necessarily reflective of a poorly designed OR/MS model.

The last category is unique to public decisions. For macro policy decisions, there is often extensive public data available for analysis. An OR/MS analyst can build a model and explore the issue. Successful modeling is reflected in the OR/MS researcher playing the role of expert in public testimony before legislative bodies or being interviewed and quoted in the mass media. In this role, the OR/MS researcher may face an ethical dilemma. At what point, if any, as the public debate evolves, does the researcher move from the role of analyst, who simply presents findings and assumptions, to the role of an advocate vigorously supporting and defending positions.

Governmental Operations

Many of the barriers to the implementation of OR/MS models to improve governmental operations in the U.S. are the same as those reported in the Total Quality Management (TQM) experience (Radin and Coffee 1993). The areas in which OR/MS has had limited success have many of the following characteristics:

- Measurement – hard to measure outputs of messy systems
- Accountability – little or none
- Lack of pressure to improve – no crisis
- Fragmentation of governmental systems
- Multiple interest groups with conflicting objectives
- Leadership – top executives and managers are political appointees that often change
- Weak management skills of core organization especially with regard to analysis
- Unstable budgeting and accompanying uncertainty and
- Extensive demagoguery

Conversely, the likelihood of success is greater in organizations with clearly defined measures for which leaders are held accountable, and that are facing major budget crises that demand improved efficiency. Thus, OR/MS had an opportunity to facilitate more effective collection of state taxes in New York (Miller et al. 2012). Similarly, areas not prone to demagoguery, such as allocating resources and scheduling road repaving, have led to OR/MS modeling successes (Feunekes et al. 2011).

The factors listed above are often interrelated. Measurement, accountability, and pressure are obviously linked. Without measures, an agency cannot be held accountable. Where will pressure come from to improve service performance? For example, what is the output of a department of social services or an unemployment agency? It would be absurd to hold any one governmental agency accountable for the number of broken families or unemployed workers. How do you measure the performance of a fire department? Can you blame the fire service for an increase in fires or fire damage? Thus, the agency can sidestep most attempts at accountability. Without total system performance accountability, it would be rare for a political leader to feel any pressure to improve services except in response to a highly publicized mishandled event.

In the 1990s, the federal government launched an initiative to use performance based budgeting to encourage more accountability and continuous improvement. It was also hoped that the federal example would lead to parallel efforts at the state level. There have been sporadic success stories at the federal level and even fewer at the state level (Jordan and Hackbart 1999). These were not enough to create a culture of measurement that might have spurred the greater use of OR/MS to make resource allocation decisions or improve processes. In contrast, in Great Britain, an OR/MS group is an integral part of the Prime Minister's Delivery Unit in the Cabinet Office that has overall responsibility for improving the operation of all government agencies (Turner 2008). In areas in which ultimate outputs cannot be measured,

OR/MS studies have successfully improved operational efficiency. One study, for example, improved the process of managing children who are placed in foster care or put up for adoption.

Even when society has a measure, there is often little known about the measure's link to the operations of relevant agencies. There is ample data on reported crime and victimization studies to monitor crime trends. The success of COMPSTAT in New York City has created a culture of precinct commander accountability for crime levels. Other large cities have attempted to copy New York's usage of crime data analytics to continually refine strategies to suppress and deter crime. However, without research that defines relationships between inputs and outputs, there is little chance for OR/MS to contribute to increased crime fighting efficiency and effectiveness.

The vertical and horizontal fragmentation of governmental systems in the U.S. adds another barrier to implementation. Local, county, state and federal agencies often play complex interrelated roles that impact the ultimate service. Although state and federal agencies assert regulatory and budgetary control, the vast majority of services are provided by an enormously diverse set of independent and relatively small jurisdictions. For example, there are more than 16,000 local area police forces in the U.S., as compared to approximately 50 in Great Britain. For the vast majority of U.S. cities, their size makes it impossible to justify the development of internal operations research groups. Also, the relative scarcity of funds and lack of accountability has discouraged the development of a critical mass of consultants to fill this technical gap. In contrast, the State of Israel has a national police force that has an internal operations research group. Great Britain has highly regarded OR/MS groups deployed broadly in government agencies that carry out operational studies and contribute to policy analysis. They assist in technology transfer and provide analytic support for local jurisdictions. Operations researchers seem to be most effective when they are embedded in interdisciplinary teams. In addition laws that require formal analysis as part of an approval process, as in the case of land use planning, also contribute to OR/MS's success. Table 1 provides 2005 data on the number of OR/MS professionals in various departments of the British government. These totals are double the number in the Civil Service 10 years earlier (Turner 2008).

Implementation of OR/MS in the Public Sector, Table 1 Government operational research services staff by department (Turner 2008)

Department	Headcount	Per cent
Department for work & pensions	89	27
HM revenue & customs	66	20
Department for education & skills	41	12
Home office	40	12
Department of health	35	11
Export credit guarantees department	18	5
Department for trade & industry	13	4
Department for transport	8	2
Office of the Deputy Prime Minister	6	2
Cabinet Office	4	1
Others	9	3

One exception to this lack of critical mass in the U.S. occurred in the 1970s when the federal government helped fund a joint venture between New York City and the RAND Corporation, the New York City-RAND Institute. This organization and its affiliated researchers at MIT developed mathematical models that continue to form the basis for analyzing the deployment of emergency services. These researcher and consultants provided the technical support needed for OR/MS model transfer to other jurisdictions. This organization, however, disappeared before the decade ended as a result of political changes in New York City. With its demise, there was a dramatic decline in the implementation of OR/MS models in city government (Green and Kolesar 2004). Now, when model implementation is attempted, it is generally a story of one or two dedicated academics and their students working with a local agency. These individuals occasionally succeed by force of will and persistence in overcoming the barriers listed above. Rarely can they provide the continuity of support necessary to institutionalize the use of an OR/MS model. One promising development has been due to the International City/County Managers Association (ICMA). In 2007, the ICMA organized an analytic public safety consulting group to help small and medium sized cities. In 2011, this group completed an average of two public safety studies a month.

In addition to fragmentation, the U.S. system of government at every level has built in checks and balances that limit the power and ability of any agency to work with a consistent vision. This dispersion of roles and responsibilities compounds the problem of accountability. For example, state legislation on prison

sentencing and state budgets for corrections may run up against court orders resulting from prison and jail overcrowding.

Public sector OR/MS studies face an added complexity in that public agencies serve multiple interest groups, all of which have a legitimate voice in the operation of government. Almost every change in operation or policy will have winners and losers. The differential impact may vary by geography and social class. Equity and efficiency measures may conflict. As a result, most public sector analyses that use models are descriptive tools for evaluating the impact of a policy or operational change on diverse segments. OR/MS models by their nature explicitly quantify the overall impact of change and identify the winners and losers. This is not necessarily a blessing for the politically appointed managers operating under freedom of information acts and sunshine meeting laws. The political clout of a small group, or in some cases a single individual, can overwhelm the analysis that supports seemingly superior solutions. The political clout could reside with the government workers themselves who might impede changes that would benefit the public, but which they perceive as negatively impacting their jobs. For example, the adamant and militant opposition of firefighters to police-fire mergers has discouraged almost all city officials from proceeding with a change that OR/MS models have shown can save money and improve performance (Matarese and Chelst 1991). As a result, the natural instincts of survival of top political appointees can lead to decisions that are not in the broad public interest and undermine the value of an OR/MS study. OR/MS procedures have been shown to facilitate a negotiated agreement among multiple interest groups with divergent objectives. OR/MS helped design an agreement on water release polices for the Delaware River by clarifying the benefits and minor risks of proposed policies on each constituency (Kolesar and Serio 2011). Similarly, OR models help competitive airlines deal with major weather related schedule disruptions and negotiate equitable allocations of landing slots that enable them to efficiently return to their daily schedule (Sud et al. 2009).

Multi-criteria decision analysis (MCDA) is one set of modeling tools that offers the greatest potential for addressing concerns of multiple interest groups with conflicting objectives. Different perspectives can be reflected in the weights assigned to the various objectives and measures. MCDA can determine whether or not these different weights affect the overall rankings of the alternatives. If they do, the discussion can focus on reasons for the differences in an attempt to identify an alternative that best balances the competing visions (Kersten 2003; Danielson et al. 2008). It is especially useful in an open participatory budget process that has been used in hundreds of municipalities around the world (Cabannes 2004).

Perhaps it could be argued that the models are often at fault because they do not adequately reflect political realities. However, is it appropriate for an operations researcher to design a model that addresses the concern articulated by a well-traveled police chief? "Why can't your model maximize the chance that the police chief will keep his job?" It is similarly inappropriate for an optimal location model to give greater weight to a neighborhood of a wealthy contributor to a recent political campaign or automatically assign a governmental office to the congressional district of a powerful House of Representative committee chairman. This dominance of politics over analysis discourages operations researchers from investing the time and energy needed to bring models to bear on local government decisions.

British OR/MS researchers have developed a structured analysis approach, termed Soft OR (soft systems methodology) that is designed to help analyze the complex multi-stakeholder real-world problems such as those encountered in the public sector, as well as in other areas (Cooper et al. 2006). Soft OR contrasts to the standard mathematical analysis approach that is basic to OR/MS, that is, Hard OR. The latter emphasizes formal data analysis and building and applying OR/MS models. Soft OR assists in structuring the problem context by incorporating the values and perceptions of the multiple stakeholders early in the identification of project goals and viable alternatives. Soft OR is process oriented and may involve workshops designed to integrate competing perspectives. It facilitates successful implementation by explicitly incorporating the views of the stakeholders in the model development and application.

The OR/MS analyst working with local and state government managers quickly notices their weak analytic skills. Many government services are case or incident driven. Promotions tend to be based more on people skills than analysis. The educational experiences of these managers also tend to minimize analysis. In addition, their case or incident based data systems rarely provide good system statistics that would be important for analysis. Thus, the OR/MS analyst is

working against the grain of the organization when attempting to bring in sophisticated mathematical models that are data driven. Interestingly, these perceived weaknesses are considered an important factor in the growth of government OR/MS groups in Great Britain, as they provide the otherwise missing quantitative analysis requested by federal authorities.

In the U.S., federal departments, agencies, and centers have a decided advantage over state and local agencies when it comes to using analysis techniques. These organizations are usually large enough to support a core group of quantitative analysts. Even when top leadership is transient and politicized, these professionals have the long-term stability to integrate OR/MS models into the decision making process. In addition, Congress routinely asks for studies and data when making policy decisions. These are some of the factors that have contributed to the use of OR/MS type models, for example, in the Department of Energy, the Federal Aviation Administration, U.S. Army Core of Engineers, U.S. Forest Service, Environmental Protection Agency, and the Centers for Disease Control and Prevention.

Further, the bizarre political budgeting process that occurs annually at every level of government discourages long-term planning that might warrant the development of OR/MS models. When faced with a sudden budget or personnel cutback, the public agency manager may decide that the wisest course is to maximize the damage caused by the cutback rather than efficiently use the resources that are left. The goal is to maximize the public outcry against the pain caused by the cutbacks so as to generate political support for restoring the lost funds as soon as possible. Needless to say, OR/MS tools are not generally intended for this type of counter-productive decisions. However, the recession of 2007–2009 and accompanying decline in property values in many locales has forced many city officials to take a closer look at the high cost of emergency services. This has contributed to the demand for studies from the ICMA public safety consulting group. Continuing deficits in the U.S. Postal Service have led the leadership of this independent agency to embrace OR/MS models to improve efficiencies (Chakravarthy et al. 2009). However, when it comes to closing local post offices or distribution centers that provide jobs, politics can undermine analysis.

The military is unique in that all of the barriers listed above are not found in most military operations decisions. All aspects of military operations can be measured in terms of effectiveness. Battles are won and lost, air-to-air combat has winners and losers, and all weapons systems have measures of effectiveness. There is also a long tradition on accountability in the military. The next war is always more or less a decade away and the military must constantly update and adjust to the pressure. The overwhelming majority of standard military operations do not run into fragmented government systems and multiple interest groups. Politics become an issue only on big decisions and interest group pressures arise only when economics come into play. Although the top defense department officials are civilian political appointees, few operational decisions get to that level. The education of the military leadership is rooted in the military academies where analytic skills are nurtured. OR/MS research has a proven track record in the military where staff personnel are often experts in military OR/MS. As a result, the military has been and will continue to be a fertile area for successful application of OR/MS.

Major Policy Initiatives and Public Debates

The same barriers discussed earlier apply to most major policy questions at both the local and state level. In addition, it is even harder to validate models at the policy level, which reduces the credibility of the analysis. OR/MS has not been a significant factor in decision making at the state level with one major exception. OR/MS models are used in many states to evaluate legislative actions and policy decisions that affect prison populations. Blumstein (2007) and others were modeling crime and prison populations for more than a decade when the crisis in prison population arose in the 1980s. The problem was clearly measurable; prison populations more than doubled, quickly outstripping available capacity and budgeted resources. The easy solutions disappeared very quickly. Overloading the prisons was prohibited be federal courts. Letting prisoners go early was politically unacceptable after several highly publicized murders. Even a strategy of building more prisons required a model to forecast growth. Some states were adding one new prison each month and watched almost helplessly as corrections' spending on buildings and operations overwhelmed almost every other state budget category. There was an

obvious link between legislative actions on sentence length and policies regarding parole eligibility. Thus, there were a number of elements that facilitated the adoption of OR/MS models to forecast prison populations. In addition, once a model was structured to meet the needs of one state, it did not require major modifications to adopt it to other states.

At the federal level, OR/MS models have played a significant role in policy analysis in a number of areas. Congressional debaters often request the respective agencies provide detailed analysis of policy and budgetary changes. In the area of economic policy, economic model forecasts are a prerequisite for political debate since the early 1970s. This has set a tone for the use of mathematical models in other aspects of federal policy and legislation. The less politicized the debate is, the greater the role of models. The U.S. Forest Service began developing and using OR/MS models in the 1970s at a time when citizens were uninterested in the issues debated by the federal government and the timber interests. As environmentalism became a factor, there was time to revamp the models to make them multi-dimensional and keep them as viable contributors to congressional debate. Models in the area of energy policy, air and water quality, and controlling reservoirs have had an analogous successful implementation experiences. In each of the agencies, the model outputs were variables that were easily measured, although relationships and assumptions may have been hard to validate.

The development of think tanks has also played an important role in the success of OR/MS models at the federal level. These organizations have the critical mass and stability needed for the development and continued refinement of models. The most noted is the RAND Corporation. Initially, it carried out national defense studies and later extended its skills to other policy areas such as health care and criminal justice. Its European division has brought those same analytic modeling skills to address critical questions of water management in the Netherlands (Walker et al., 1994).

OR/MS researchers who have made long-term commitments to a particular problem area are the key to OR/MS's contribution to debates in areas such as criminal justice, aviation safety, and health care. OR/MS models have provided these nationally recognized experts a unique perspective, but their knowledge of the issues extends far beyond just modeling. It is this breadth of knowledge that is critical to achieving credibility in the public debate. Typically, these OR/MS subject area experts head up a group of researchers either at a university or think tank. This has enabled them to maintain a long-term focus as the important topics develop and evolve.

See

- ▶ Community OR
- ▶ Crime and Justice
- ▶ Environmental Systems Analysis
- ▶ Ethics in the Practice of Operations Research
- ▶ Military Operations Research
- ▶ Model Evaluation
- ▶ Politics
- ▶ Practice of Operations Research and Management Science
- ▶ Public Policy Analysis
- ▶ RAND Corporation
- ▶ Soft Systems Methodology
- ▶ Verification, Validation, and Testing of Models

References

Ad hoc Committee. (1971). Guidelines for the practice of operations research. *Operations Research, 19*, 1123–1258.

Blumstein, A. (2007). An OR missionary's visits to the criminal justice system. *Operations Research, 55*(1), 14–23.

Bollinger, D., & Pictet, J. (2003). Potential use of e-democracy in MCDA processes, analysis on the basis of a Swiss case. *Journal of Multi-criteria Decision Analysis, 12*, 65–76.

Botha, S., Gryffenberg, I., Hofmayr, F. R., Lausberg, J. L., Nicolay, R. P., Smit, W. J., Uys, S., van der Merwe, W. L., & Wessels, G. J. (1997). Guns or butter: Decision support for determining the size and shape of the South African National Defense. *Interfaces, 27*(1), 7–28.

Cabannes, Y. (2004). Participatory budgeting: A significant contribution to participatory democracy. *Environment and Urbanization, 16*(1), 27–46.

Caulkins, J. P. (2005). *How goes the war on drugs? An assessment of U.S. drug problems and policy*. Santa Monica, CA: RAND.

Chaiken, J. (1978). Transfer of emergency service deployment models to operating agencies. *Management Science, 24*, 719–731.

Chakravarthy, A., Gu, Q., & Zhang, X. (2009). Review of models and methodology for scheduling problems in USPS mail processing and distribution centres. *Journal of the Operational Research Society, 5*, 445–467.

Cooper, C., Brown, J., & Pidd, M. (2006). A taxing problem: The complementary use of hard and soft OR in the public sector. *European Journal of Operational Research, 172*, 666–679.

Daniel, S. E., Diakoulaki, D. C., & Pappis, C. P. (1997). Operations research and environment planning. *European Journal of Operational Research, 102*, 248–263.
Danielson, M., Ekenberg, L., Ekengren, A., Hokby, T., & Liden, J. (2008). Decision process support for participatory democracy. *Journal of Multi-criteria Decision Analysis, 15*, 15–30.
Feunekes, U., Palmer, S., Feunekes, A., MacNaughton, J., Cunningham, J., & Mathisen, K. (2011). Taking the politics out of paving: Achieving transportation asset management excellence through OR. *Interfaces, 41*(1), 51–65.
Gabriel, S. A., Kydes, A. S., & Whitman, P. (2001). The national energy modeling system: A large-scale energy-economic equilibrium model. *Operations Research, 49*, 14–25.
Gass, S. I. (1983). Decision-aiding models: Validation, assessment, and related issues for policy analysis. *Operations Research, 31*, 603–631.
Gass, S. I. (1991). Model world: Models at the OK corral. *Interfaces, 21*(6), 80–86.
Green, L. V., & Kolesar, P. J. (2004). Improving emergency responsiveness with management science. *Management Science, 50*, 1001–1014.
Greenberg, H. J. (1995). Mathematical programming models for environmental quality control. *Operations Research, 43*, 578–622.
Jordan, M. M., & Hackbart, M. M. (1999). Performance budgeting and performance funding in the states: A status assessment. *Journal of Public Budgeting, Accounting & Financial Management, 19*, 68–88.
Kersten, G. (2003). e-democracy and participatory decision processes: Lessons from e-negotiation experiments. *Journal of Multi-criteria Decision Analysis, 12*, 127–143.
Kolesar, P., & Serio, J. (2011). Breaking the deadlock: Improving water-release policies on the Delaware River through operations research. *Interfaces, 41*(1), 18–34.
Labadie, J. W. (2004). Optimal operation of multi-reservoir systems: State-of-the-art review. *Journal of Water Resources Planning and Management, 130*, 93–111.
Lawless, M. W. (1987). Institutionalization of a management science innovation in police departments. *Management Science, 33*, 244–252.
Little, J. D. C. (1970). Models and managers: The concept of a decision calculus. *Management Science, 16*, B466–B480.
Matarese, L. A., & Chelst, K. R. (1991). Forecasting the outcome of police/fire consolidations. *Management Information Service Report, 23*(4), 1–22.
Miller, G., Weatherwax, M., Gardinier, T., Abe, N., Melville, P., Pendus, C., Jensen, D., Reddy, C. K., Thomas, V., Bennett, J., Anderson, G., & Cooley, B. (2012). Tax collections optimization for New York State. *Interfaces, 42*(1), 74–84.
Murphy, F. H. (1991). Policy analysis in a political environment. *Interfaces, 21*(6), 87–91.
Murphy, F. H., & Shaw, S. H. (1995). The evolution of energy modeling at the federal energy administration and the energy information administration. *Interfaces, 25*(5), 173–193.
Pollock, S. M., Rothkopf, M. H., & Barnett, A. (Eds.). (1994). *Operations research and the public sector* (Handbooks in operations research and management science, Vol. 6). Amsterdam: North Holland.
Quade, E. S. (1989). *Analysis for public decisions* (3rd ed.). New York: North Holland.
Radin, B. A., & Coffee, J. N. (1993). A critique of TQM: Problems of implementation in the public sector. *Public Administration Quarterly, 17*, 42–54.
Rios, J., & Rios, D. (2008). A framework for participatory budget elaboration support. *Journal of the Operational Research Society, 59*, 203–221.
Royston, G., Halsall, J., Halsall, D., & Braithwaite, C. (2003). Operational research for informed innovation: NHS direct as a case study in the design, implementation and evaluation of a New Public Service. *Journal of the Operational Research Society, 54*, 1022–1028.
Sahney, V. K. (1993). Evolution of hospital industrial engineering: From scientific management to total quality management. *Journal of the Society for Health Systems, 4*(1), 3–17.
Sud, V. P., Tanino, M., Wetherly, J., Brennan, M., Lehky, M., Howard, K., & Oiesen, R. (2009). Reducing flight delays through better traffic management. *Interfaces, 39*(1), 35–45.
Turner, H. S. (2008). Government operational research service: Civil OR in UK Central Government. *Journal of the Operational Research Society, 59*, 148–162.
Walker, W. E., Abrahamse, A., Bolten, J., Kahan, J. P., Van De Riet, O., Kok, M., & Braber, M. D. (1994). A policy analysis of Dutch River Dike improvements: Trading Off safety, cost, and environmental impacts. *Operations Research, 42*, 823–836.
Watson, H. J., & Marett, P. J. (1979). A survey of management science implementation problems. *Interfaces, 9*(4), 124–128.
Wein, L. M. (2009). OR forum – homeland security: From mathematical models to policy implementation: The 2008 Philip McCord Morse Lecture. *Operations Research, 57*, 801–811.
Weintraub, A., & Bare, B. B. (1996). New issues in forest land management from an operations research perspective. *Interfaces, 26*(5), 9–25.
White, L., Smith, H., & Currie, C. (2011). OR in developing countries: A review. *European Journal of Operational Research, 208*, 1–11.
Yewlett, C. J. L. (2001). OR in strategic land-use planning. *Journal of the Operational Research Society, 52*, 4–13.

Implicit Enumeration

A process for solving integer-programming problems in which all possible integer solutions need not be investigated (enumerated) due to information obtained in the process that relates to problem feasibility and value of the objective function. That is, certain solutions need not be pursued as it can be shown that they would lead to infeasible solutions or values of the objective function that are worse than those that are known to be possible.

See

► Branch and Bound

Implicit Price

► Marginal Value

Importance Sampling

In stochastic or Monte Carlo simulation, a variance reduction technique whereby the underlying probability distribution is altered to increase the probability of (1) simulating events of highest interest, such as rare events, or (2) sampling from regions that have a larger effect on the quantity being estimated, such as a high-dimensional integral.

See

► Monte Carlo Methods
► Monte Carlo Simulation
► Simulation of Stochastic Discrete-Event Systems
► Variance Reduction Techniques in Monte Carlo Methods

Impossibility Theorem

► Group Decision Making

References

Arrow, K. J. (1951). *Social choice and individual values* (Cowles commission monograph, Vol. 12). New York: Wiley.

Inactive Constraint

An inequality constraint of an optimization problem that is satisfied as a strict inequality.

See

► Active Constraint
► Slack Variable
► Surplus Variable

Incidence Matrix

► Node-Arc Incidence Matrix

Incident

An edge of a graph is said to be incident with the two nodes it connects, and conversely.

See

► Adjacent
► Node-Arc Incidence Matrix

Independent Float

The amount of time that an activity can be delayed without affecting the earliest start of the preceding activity and the latest finish of the succeeding activity in a project network.

See

► Network Planning

Independent Private Values Bidding Model

A bidding model in which a bidder's estimate of value for what is being auctioned is statistically independent of the value estimate for any other bidder. In such a model, no bidder has any reason to adjust an estimate of value upon learning the information of any other bidder.

See

► Bidding Models

Indirect Costs

In the simplex method, the indirect costs are found by taking the inner products of the multiplier (pricing) vector with each column of the problem's defining $\boldsymbol{A}$ matrix. This product, for a column j, is usually denoted by z_j. For c_j, the original objective function coefficient for column j, the term $(z_j - c_j)$ or $(c_j - z_j)$ is used to determine if the associated variable is a candidate to enter the basic feasible solution. For any basic variable x_k, $(z_k - c_k) = 0$. The $(z_j - c_j)$ terms are called relative costs (relative to the basis) or reduced costs.

See

► Prices

Industrial Applications

Jaya Singhal[1], Leonard Fortuin[2], Paul van Beek[3] and Luk Van Wassenhove[4]
[1]University of Baltimore, Baltimore, MD, USA
[2]Eindhoven University of Technology, Eindhoven, The Netherlands
[3]Wageningen University, Wageningen, The Netherlands
[4]INSEAD, Fontainebleau, France

Introduction

Although this article is based largely on some of the authors' experiences and views of European OR/MS, it is of direct importance to the worldwide OR/MS community. The field is better known in Europe as Operational Research or OR, and it occupies itself with quantitative methods for the analysis and solution of management problems. Its origins lie in military organizations during World War II: first the Royal Air Force (U.K.) preparing for the Battle of Britain and later on the U.S. Navy fighting German U-Boote (submarines). After the war, there was a general feeling that OR/MS could also be helpful to managers in industry, government, public services, and financial institutions. The logic was obvious: industrial activities such as production planning, inventory control, and physical distribution were quite suitable for model building and other forms of abstraction that lead to challenging mathematical problems, and trained OR workers (analysts) were available. But soon it became evident that solutions capable of being applied in practice were not as numerous as expected. Causes for this phenomenon can be traced to the following.

On the one hand, models running on the then available computers were so strongly a simplification of reality that managers did not recognize their problems any longer. On the other hand, OR researchers in academia moved their attention to the basics of the discipline. Their theoretical results were very impressive, especially in the field of mathematical programming, combinatorial analysis, and queueing theory. But for managerial problems of daily life, these OR researchers tended to have little interest. Consequently, decision makers felt disappointed and lost their confidence in OR/MS and returned to simple, often too simple, rules of thumb. In this way, a practicality gap came into existence, a gap between the managers with real, urgent decision problems demanding simple solutions, and the OR/MS scientists who strived for elegant solutions to abstract problems of their own invention. For a discipline founded on solving real-world problems, as OR claimed to be, the gap caused a highly unsatisfactory situation. Hence, after a while, professional journals tried remedy the situation, while an outstanding cadre of OR/MS analysts attempted to regain managers' interest for management science. But all efforts seemed in vain. One of the gurus of OR/MS even concluded that "The future of OR is past" (Ackoff 1979).

This is how OR/MS lost the good reputation earned during the war. Even OR analysts in staff departments of industrial companies had to fight for their existence and often lost their jobs. Many departments were dissolved or put at work on other tasks, for example, on automation projects. Mainly, the so-called loners, working in decentralized positions, continued to do OR/MS work (see Fortuin and Lootsma 1985).

But OR/MS analysts never lost faith in their discipline. Gradually they improved their position by

rediscovering real-world problems. In the 1980, two developments fostered this process: the availability of low-cost and versatile computer power (PCs) and the establishment of special university chairs for OR/MS and other quantitative methods. Ten years after Ackoff, a completely different sound could be heard: "The future of OR is bright?" (Rinnooy Kan 1989).

The Faces of OR/MS

OR/MS has two faces: on the one hand it concentrates on operations and as such it tries to be practical and to provide solutions to real-world problems; on the other hand, OR/MS means research, involving theoretical studies of problems that at best may be regarded as abstract version of problems that actually exist in the real world. These two faces of OR/MS have brought into existence two types of OR/MS workers: the practitioners and the theoreticians. The practitioners are to be found primarily in consultancies, but also at universities, for example in departments such as industrial engineering and industrial mathematics. In large companies, loners can still be found. As for the theoreticians, they tend to work only at universities and related institutions.

The two types of OR/MS workers are carrying out their tasks independently, but contact between them is improving, with exchange of ideas at conferences and seminars, and bilaterally. This situation originated in a natural way:

- Most consultants graduated from a university. They maintain their university network to learn about theoretical breakthroughs. In return, they inform their fellow OR/MS analysts in academia about the problems their clients in industry are grappling with.
- Many consultants are working as part-time professors at universities. They use their experiences to keep their teaching up-to-date and use the results of their academic studies to support their consultancy work.

In this way, opportunities for OR/MS have improved considerably. Other factors have enhanced this process:

- Modern managers in industry have an academic background. During their studies, they have become acquainted with the basics of OR/MS and, thus, they are easier to convince that OR/MS can help them. As most managers no longer have OR/MS staff departments in their organizations, they become clients of OR/MS consultants.
- Universities have discovered the importance of good relations with business companies:
  1. It makes academic OR/MS workers more practical and teaches them to cooperate with managers.
  2. It enhances their cash flow by doing contract research in OR/MS.
  3. It offers students an opportunity for working temporarily in an industry as part of their program, to the benefit of the quality of their education.
  4. It helps universities to assign priorities to the items on their research program.
- The pure theoreticians less often select in isolation the subjects of their investigations. Instead, they have opportunities to pay attention to the signals that reach them from their colleagues operating with their students in industry.
- Information technology has produced powerful computer software and hardware. Consequently, model building has become very realistic; all relevant details are taken into account, and animated graphics convince managers more easily than words that indeed their problems are being analyzed.

This improvement process is reflected in professional journals on OR/MS. Many successful applications of OR/MS in practice have been reported in the literature. These case studies usually have following sequence: (1) the problem and its environment; (2) the OR/MS approach towards a solution; (3) results of the OR/MS analysis; (4) selection by management of a solution from a set of alternatives; (5) implementation of the solution; and (6) results in terms of improvements with respect to the situation before the OR/MS intervention. Examples can be found in (Bell 1985; Lootsma 1991; Fortuin and Korsten 1988).

There will always be managers who have to make decisions in complex and complicated situations, each with far-reaching consequences. They are obliged, mostly under time pressure, to select the best solution. Here they can be supported with OR/MS in its modern version, given that they are aware of the powerful methods and tools that OR/MS practitioners have at hand and the impact that OR/MS has had in

solving real-world problems; see (Fortuin et al. 1992; Davenport 2006; Liberatore and Luo 2010).

The main problem lies in gaining the confidence of managers so that they are prepared to give OR/MS a chance. The practicality gap has been narrowing, and there exist opportunities to enhance engagement between research and practice (Corbett and van Wassenhove 1993; Sodhi and Tang 2008). To bridge the gap, OR consultants and academic OR workers have to act as missionaries to the benefit of their profession, to their career perspectives, and to help managers trying desperately to improve their business in the face of an ever increasing global competition.

OR/MS In Industry: Where And What?

Fortuin and Zijlstra (1989) reported on the experiences of an OR group within Philips Electronics, a multinational company producing consumer products (e.g., domestic appliances, lighting, television sets, high-fidelity consumer electronics, and razors) and professional products (e.g., medical systems, telephone exchanges, and lighting systems). They analyzed over 200 projects in OR/MS and showed which areas were most important for OR/MS application in industry and which OR/MS tools were most frequently used. A 1992 update of these investigations confirmed these results. Apparently, the most frequently occurring projects were the ones on the design of a production systems, whereby discrete computer simulation is the OR/MS tool employed to take complex interactions into account (Tables 1 and 2). This conclusion holds for a large multi-national company in Europe. In the U.S., the picture seems to be slightly different. A longitudinal survey in the journal *Interfaces*, for instance, mentions statistics, linear programming, and discrete simulation as the top three OR tools, in that order (Harpell et al. 1989).

Industrial Applications, Table 1 Application areas for OR/MS in industry

Areas of application	Number of applications
Design of production systems	95
Production	86
Transport and storage	35
Training and courses	16
Design of systems for transport and storage	13
Performance of systems	11
Miscellaneous such as portfolio analysis, measuring the quality of information systems, and performance indicators	20

Industrial Applications, Table 2 OR/MS tools applied in industry

OR/MS tools	Number of applications
Discrete event simulation	95
Waiting theory models	82
Combinatorial analysis	48
Inventory models	46
Mathematical programming with emphasis on linear programming	23
Miscellaneous such as the structuring of facts and figures (many projects start with this type of OR), which is sometimes is all that the client desires	29

Model Building

In most OR/MS projects in industry, an important part of the work is model building. A model is the description of a piece of reality that has to be analyzed in the course of the project, leaving out all irrelevant details while maintaining essential characteristics. This gives model building the character of an art rather than of a science.

Model building plays an important part in modern OR/MS projects, often in combination with discrete simulation and optimization. Models offer insights and the possibility to compare decision scenarios in both the qualitative and the quantitative senses. The computer is almost always an indispensable tool in this pursuit. The driving force exerted by development in informatics cannot easily be overestimated. A large part of the arsenal of OR techniques can be used on a PC, thanks to software that is becoming more and more user-friendly and cheaper. The ease with which a problem area can be represented by a model that the problem owners consider sufficiently realistic has grown enormously. Large quantities of data can easily be stored in databases that are simple to access. The opportunities

are great for OR/MS to really contribute to the reduction of uncertainty in complex industrial situations and to increasing control of business processes. But, there still are managers who are unaware of the help they can get from OR/MS: quickly calculating the consequences of decision variants, the preparation of decisions, and decision support. These managers are unaware that they can save considerable amounts of money and improve effectiveness of their decisions.

The Position of OR/MS in Industry

Times have changed. In the 1980s, OR/MS was primarily in the hands of a company's staff department. Since then, many companies have reorganized about their core business. Consequently, staff departments were reduced to a bare minimum, if not completely eliminated. Also, many were disconnected from their original company. For instance, the OR/MS department discussed in Fortuin and Zijlstra (1989) became an independent consultancy with, not surprisingly, Philips Electronics as its main client. Such reorganizations also occurred in many industrial companies in Europe, as well as in North America.

In Europe, OR/MS support is offered to industry from two sources. First, there are the consultancies. They usually follow the project approach, according to which the work is done in phases. To a certain extent, they have to compete with the second source, that is, OR/MS university departments whose study program includes real-world student-oriented projects that stem from companies willing to invest in such OR/MS projects. Both parties profit from this alliance: the students learn to practice the profession, while the company obtains a relatively inexpensive solution to a problem or, at least, begins to explore it by using the experience and talents of both professors and students. These academic OR activities are an important means for showing managers how advantageous the support of external consultants can be.

The Project Management Approach

Table 3 gives an overview of the OR/MS project approach. A contract between the client (company) and the OR/MS consultancy is agree to, with the contract stating what problem will be studied, and possibly solved, at what cost, and the study time interval. The contract also states the contributions the client and company's staff are obligated to contribute, and indicates what deliverables the client may expect. More details can be found in Fortuin et al. (1992). This approach has proven to be very successful, for two reasons:

- The preliminary phase is usually short. Its aim is reconnaissance of the problem area and a problem description that the problem owner (manager) can agree with. Costs are relatively modest so that financial risk is low. This facilitates the process of making the manager confident that the consultancy is indeed able to help solve the problem.

Industrial Applications, Table 3 Summary of the steps in an OR/MS project: In phases 1 and 2, the consultant is heavily involved in the project. Usually, Phase 3 is carried out by the client and/or staff

	Activities	Details
Phase 1	General survey	Discussions with client and staff
		Interviews, study of document
		Global problem description
		Generation of ideas for a possible approach
	Reporting	Outline of results to be expected
		Proposal for Phase 2
Phase 2	Model building	Systematic description of the problem area. In order not to make the model too complicated, only the most relevant factors are taken into account
	Verification	Discussions with client and staff: Is the model correctly presenting the problem area, the organization, the methods, processes, and procedures?
	Experiments	Translation of the model into computer program
		Calculations under various circumstances
	Analysis	Investigation of results
	Reporting	Presentation of the most important results, conclusions, and recommendations. A proposal for Phase 3
Phase 3	Implementation	Working out and implementation of recommendations
		Teaching client and/or staff to work with the new method

- It may happen, that during the preliminary phase, the problem becomes so transparent that a solution can be seen immediately. Then a follow-up phase is not necessary.

Concluding Remarks

The prospects for the continued application of OR/MS in industry are excellent:

1. Managing an industrial company is becoming ever more complicated. Global competition, globalization of the economy, demanding customers, decreasing profit margins, new markets, and fluctuating exchange rates are just a few of the causes. The time for simple solutions is over, and only fundamental and theoretically sound analyses can justify management decisions. Managers lack the time and the expertise for such analyses.
2. Practitioners now working in independent consultancy units can make their own business plans and follow their own strategy when promoting OR/MS, rather than being ruled, or overruled, by a general company policy.
3. Computer power is widespread in industry, which facilitates the implementation of OR/MS solutions, even if the problems are complex and their resolution requires large and diverse data bases.

See

► Practice of Operations Research and Management Science

References

Ackoff, R. L. (1979). The future of OR is past. *Journal of Operational Research Society, 30*, 93–104.

Bell, P. (1985). *Successful operational research in Canada*. Ottawa, Canada: Canadian Operational Research Society.

Corbett, C., & van Wassenhove, L. (1993). The natural drift: What happened to operations research? *Operations Research, 41*, 625–640.

Davenport, T. (2006). Competing on analytics. *Harvard Business Review, 84*, 98–107.

Fortuin, L., & Korsten, A. (1988). Quantitative methods in the field: Two case studies. *European Journal of Operational Research, 37*, 187–193.

Fortuin, L., & Lootsma, F. (1985). Future directions in operations research. In A. H. G. Rinnooy Kan (Ed.), *New challenges for management research*. Amsterdam: North-Holland.

Fortuin, L., van Beek, P., & Van Wassenhove, L. (1992). Operational research can do more for managers than they think! *OR Insight, 5*(1), 3–8.

Fortuin, L., & Zijlstra, M. (1989). Operational research in practice: Experiences of an OR group in industry. *European Journal of Operational Research, 41*, 108–121.

Harpell, J., Lane, M., & Mansour, A. (1989). Operations research in practice: A longitudinal study. *Interfaces, 19*, 65–74.

Liberatore, M., & Luo, W. (2010). The analytics movement: Implications for operations research. *Interfaces, 40*, 313–324.

Lootsma, F. (1991). Perspectives on operations research in long-term planning. *European Journal of Operational Research, 50*, 76–84.

Rinnooy Kan, A. H. G. (1989). The future of OR is bright. *European Journal of Operational Research, 38*, 282–285.

Sodhi, M., & Tang, C. (2008). The OR/MS ecosystem: Strengths, weaknesses, opportunities, and threats. *Operations Research, 56*, 267–277.

I

Industrial Dynamics

► System Dynamics

Infeasible Solution

In general, a proposed solution to an optimization problem that does not satisfy all the constraints. For the linear-programming problem $\boldsymbol{Ax} = \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}$, a vector $\boldsymbol{x}^0$ is an infeasible solution if it does not fully satisy the equations or the nonnegativity condition.

See

► Feasible Solution

Inference Engine

A piece of software or a computational strategy that is based on a problem statement from the user, uses reasoning knowledge about the problem area in attempting to derive a solution, gathers needed problem-specific information (e.g., from the user) in the course of reasoning, explains why it needs this

added information, presents the solution to the user, and explains the line of reasoning used in reaching the solution.

See

- ▶ Artificial Intelligence
- ▶ Expert Systems

Infinitesimal Generator Matrix

- ▶ Rate Matrix

Infinitesimal Perturbation Analysis

- ▶ Perturbation Analysis

Influence Diagrams

James E. Matheson
SmartOrg, Inc., Menlo Park, CA, USA

Introduction

Before influence diagrams were developed, describing and solving decision problems under uncertainty was quite difficult. The first difficulty was determining the probabilistic relationship among uncertain variables, because it is easy to model many variables as jointly related, but extremely difficult to assess their probabilistic relationship. An experienced decision analyst can reasonably assess the uncertainty in a single variable (Spetzler and Staël von Holstein 1975), but more than two variables make the task almost impossible. A better way was needed to understand the relationship among uncertain variables. The second difficulty was understanding and describing the relationship between decisions and uncertainties, particularly indicating which uncertainties would be revealed before which decisions and then transforming the probabilistic descriptions to condition the probabilities in the proper order of information revelation, using Bayes' Rule. The usual, but very awkward, method was to describe the uncertainties in a possibly very large probability tree, called nature's tree, and to describe the sequence of revelation of uncertainties and decisions in a decision tree, followed by the calculations necessary to transform the probabilities into the sequence needed for the decision tree (Howard 1965). Only an expert could hope to deploy these methods, both to think through and describe the problem and to do the complex assessments and computations involved. Then, communicating what had been done, especially to decision makers, was also a daunting task.

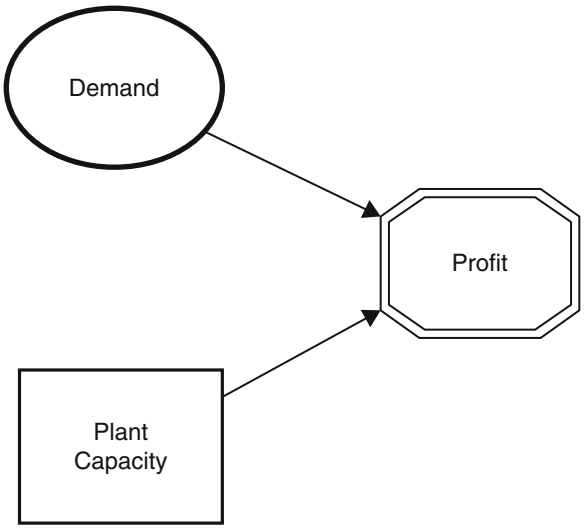
The SRI International Decision Analysis Group was highly active from about 1968 to the early1980's. This group comprised many motivated individuals doing pioneering applications and research on decision analysis, largely defining the Stanford School approach to decision analysis. It included several award-winning individuals: Ronald A. Howard, James E. Matheson, D. Warner North, and Carl S Spetzler. These individuals, along with several others, Miley W. (Lee) Merkhofer and Allen C. Miller, were engaged in many problems of probabilistic inference, especially on a project initiated in 1973 for the Defense Intelligence Agency, and a previous project on space-mission planning. Motivated by the need to efficiently assess highly-dependent information regarding the conflict in the Persian Gulf, the group tried many methods, such as coalesced decision trees – ones having repetitive structures – to capture that information. Ultimately, this exploration developed methods for graphically mapping probabilistic dependence and produced a signal flow graph method for treating states of information, termed "influence diagrams". These formed a directed graph showing the relationships (arrows) among both decisions and uncertainties (nodes) that became simultaneously a presentational device and a computational tool. A Defense Projects Research Agency project followed, where these ideas were solidified and fully described in Howard et al., (1976). This seminal paper was published privately and became one of the most-referenced works in the field. It was reprinted in a special issue of the *Decision Analysis Journal* about 30 years later (Howard et al. 1976; Howard and Matheson 1983, 2005a, b).

The Nature of Influence Diagrams

Influence diagrams represent both uncertainties and decisions in a single compact graph. They contain both the nature's tree and the decision tree of the older cumbersome method, showing the probabilistic relationships among the uncertainties, the sequencing of the decisions, and the information revealed before each decision is taken. These relationships are shown by a set of decision and chance nodes with arrows connecting them in an acyclic graph. The graphical relationships are easy to comprehend so that influence diagrams have become a standard method for describing decision problems and explaining them to subject-matter experts and to decision makers. In addition, the nodes of the influence diagram capture the numerical data describing the situation and the whole diagram may be processed to solve the decision problem, to do probabilistic inference, and to make many different insightful derivative graphs and calculations. One potential output of an influence diagram processor would be a decision tree describing all or a reduced portion of the decision situation. Decision trees have shifted from a computational one to more of a presentation role, and are easily derived as a byproduct of influence diagram representations.

About a decade later, a parallel development arose in the statistics and artificial intelligence communities, called Belief Nets or Bayesian Networks, that are roughly equivalent to influence diagrams, but with no decision nodes. The focus of work in this area has been in treating complex probabilistic inference problems and causality issues (Pearl 1986, 2005; Neapolitan 2004). In contrast, the development of influence diagrams has focused on decision making under uncertainty.

An influence diagram for a typical real problem is shown in Fig. 1. The decision nodes are represented by squares or rectangles, while chance nodes are represented by circles or ovals. It has become a common convention to use a double border, especially on chance nodes, to indicate that the node contains only deterministic relationships, such as equations or data tables. Also, often a payoff or profit node is included, usually as a deterministic function (which might be as complex as a large Excel spreadsheet), usually indicated by an octagon, representing a stop sign, which sometimes degenerates into a hexagon for easier graphics.

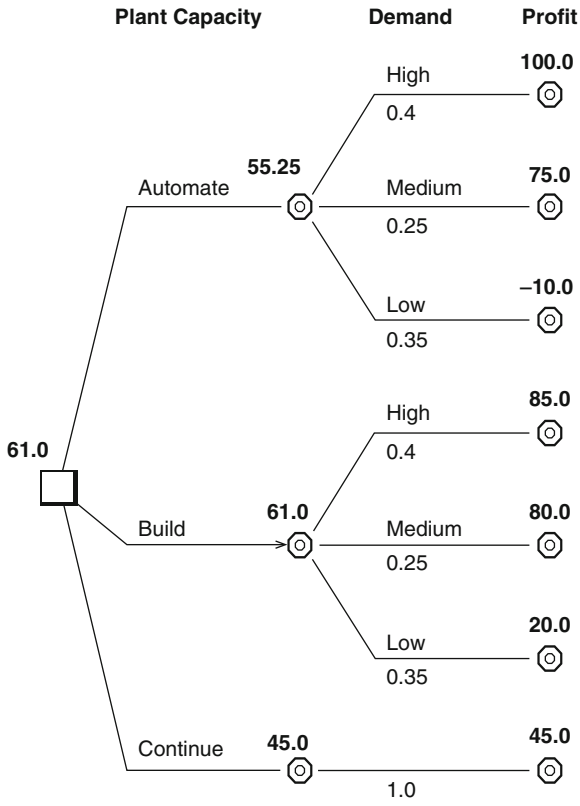


Influence Diagrams, Fig. 1 Influence Diagram for the Capacity Example

A Straightforward Example

The following simplified example is from Matheson and Matheson, (2005). A company is expanding an old plant and wants to decide on how much and how to expand. Their only uncertainty is what the demand will be, because they do not want to over-build the plant and pay for unneeded capacity or under-build and lose profit opportunities from insufficient production. Figure 1 is an influence diagram for this problem, which has three kinds of nodes, a decision node, representing the Plant Capacity; a chance node, representing Demand; and a value, or in this case, a Profit node, that contains the calculation needed to determine profit. The double outline on the profit node means that at present the information inside the node is a deterministic function of its inputs. (During processing of the diagram the uncertainty in the Demand node could be pushed into the Profit node, at which point the Profit node would become uncertain and have a single border).

Using a computer interface analogy, inside each of these nodes is embedded information describing that node. Inside the Plant Capacity node is a list of possible alternatives under consideration, in this case Continue as is, with no increase in capacity; Build a conventional expansion; or retrofit, and fully Automate an expanded plant. In each case, expansion plans and schedules have been specified, but are characterized by only



Influence Diagrams, Fig. 2 The Solved Decision Tree for the Capacity Example

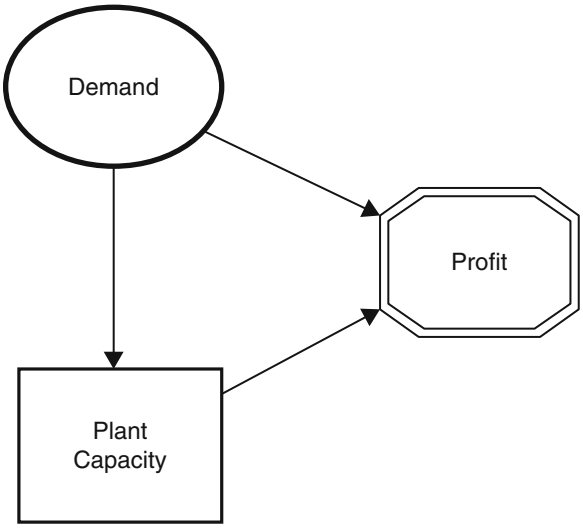
one word for convenience. Similarly, there are three well-specified potential demand scenarios: Low, Medium and High; with probabilities carefully assessed as 0.35, 0.25, and 0.40. Again, this information can be thought of as inside the node. It is critical to note that this demand probability assessment does not depend upon the plant capacity selected, which is indicated structurally by the absence of an arrow from Plan Capacity to Demand. Lastly, a financial value model has been developed for this decision situation which projects cash flows for each combination of decision (Capacity) and uncertain outcome (Demand) and reduces each set of cash flows to a Net Present Value (NPV). This model can be thought of as residing inside the Profit node.

Because this example is so limited, the influence diagram can be solved by using it to generate the decision tree of Fig. 2. It shows that the best decision is to Build an expanded conventional plant and make an expected profit of 61.0. For simplicity, assume that the company desires to maximize expected value, but the influence diagram can also treat risk aversion, for example, by using certain equivalents in place of expected values. While this result solves the primary decision problem, many further analyses are possible to gain insight into the situation. Decision analysis often asks hypothetical questions about modifying the original problem before making the primary decision. Influence diagrams are ideal for clearly specifying the nature of these hypothetical interventions, see Matheson, (1990). Two interventions to be considered are gathering more information and gaining control over demand. Assume that these hypothetical interventions can be made without disturbing the rest of the problem structure. In other words, there are no unintended consequences. If some of these idealized interventions look valuable, then real interventions with all of their potential interactions should be investigated.

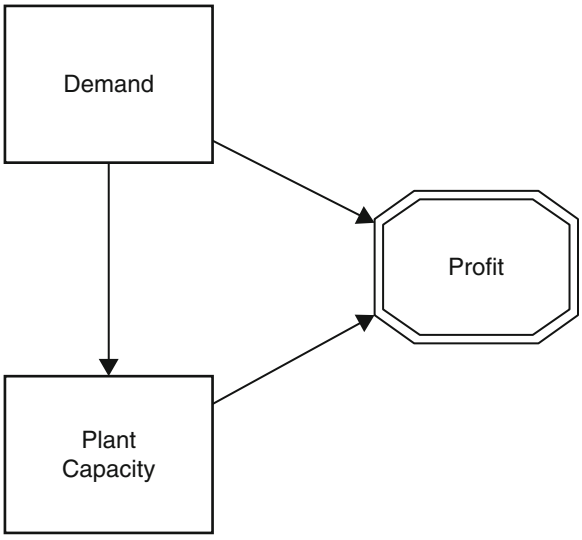
Valuing Interventions

The first intervention to consider is gathering perfect information on Demand before making the original primary decision. This intervention is incorporated into the structure of the influence diagram by adding an arrow from the Demand node into the Plant Capacity node, which indicates that the decision maker will know demand (e.g. the demand will be revealed by a clairvoyant) before the capacity is selected, as shown in Fig. 3. A new decision tree could be generated from this diagram, using the new ordering, but this problem can be solved by just inspecting Fig. 2. If the decision maker knows the demand is going to be Low, the decision maker makes a maximum of 45.0 by choosing Continue as is; if the demand will be Medium, value is maximized with Build for 80.0; if the demand will be High, the maximum value is achieved by Automate for 100.0. Since a true clairvoyant will report each of these forecasts with the same probabilities that have already been assessed, these numbers are weighted by those probabilities to get an expected value of 75.75 and subtracting the value of the original situation of 61 to give an expected value of (free) perfect information of 14.75, sometimes called EVPI.

Another basic intervention is perfect control. If the decision maker could somehow set a specific demand of interest (e.g. employing a wizard), before deciding

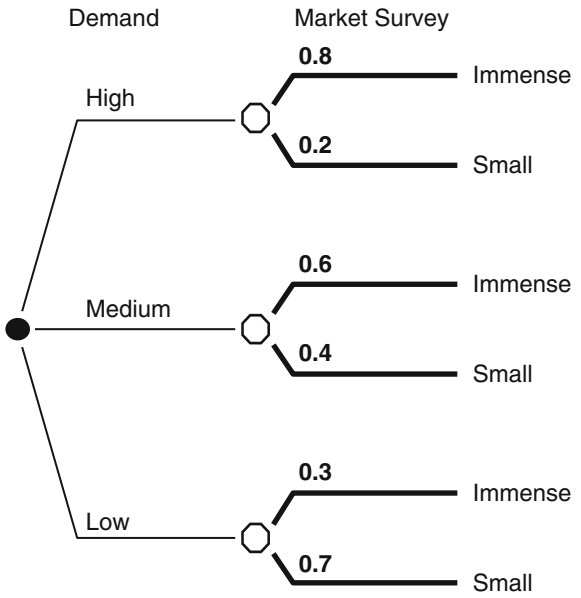


Influence Diagrams, Fig. 3 The Capacity Example with Perfect Information on Demand



Influence Diagrams, Fig. 4 The Capacity Example with Perfect Control of Demand

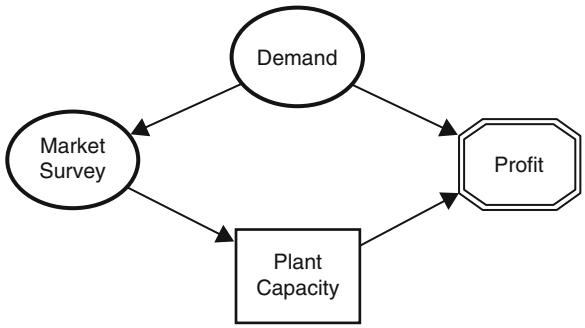
on capacity, the decision maker could pick the best of all possible worlds. In Fig. 4, the Demand node has been changed to a decision node representing what to ask of the wizard, with an arrow to indicate the demand choice is made before selecting the Plant Capacity. Again, inspecting the earlier tree determines that it would be best to chose the High demand and Automate the plant leading to a profit value of 100,



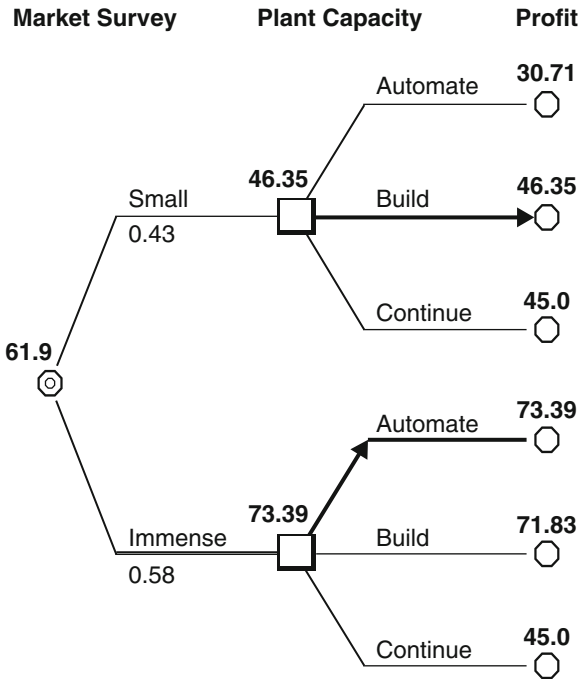
Influence Diagrams, Fig. 5 The Conditional Distribution of the Market Survey

which subtracting the original 61.0 lead to a value of perfect control of 39.0.

The analyst and the decision maker now look for more realistic interventions that maybe undertaken. Assume they cannot find a legal means to control demand, but they do have a proposal to undertake a market survey, that can report that the market will be either Immense or Small. Such a report is usually modeled by assigning a probability of each report conditional on each case of what the market might actually be, as shown if Fig. 5. This information is captured by inserting a new chance node into the original influence diagram for the result of the Market Survey, drawing an arrow from the Demand node to that node, indicating the functional dependence or probabilistic relevance, and drawing another arrow from the Market Survey node to the Plant Capacity decision node indicating that the decision maker will know the market survey results when making the primary decision, as shown in Fig. 6. Now there is a problem of how to draw a proper decision tree to solve this influence diagram, as simply following the arrows will not lead to the correct tree. Most readers will recognize that what is needed is the application of Bayes' Rule to reverse the arrow between the Demand node and the Market Survey node. In an influence diagram, an arrow can



Influence Diagrams, Fig. 6 Influence Diagram for the Capacity Example with a Market Survey



Influence Diagrams, Fig. 7 Abbreviated Solved Decision Tree for the Capacity Example with a Market Survey

always be reversed (by Bayes' Rule) if both nodes have the same arrows into them, except for the arrow between them, because this means both nodes are conditioned on the same state of information. New arrows may be added to create this condition, but only if they do not create directed loops in the diagram. In the original paper (Howard et al. 1976), a valid influence diagram describing a decision was called a decision diagram, and one with all nodes reversed so that a tree could be drawn by simply following the arrows is called a decision tree diagram. The ability to express the complete problem the way it is assessed, and then do this kind of manipulation, is a unique advantage of influence diagrams.

Prior to influence diagrams, a decision analyst used two separate constructs, the decision tree (to state the sequential nature of the problem) and nature's tree (to state the original probability assessments), and then manipulated nature's tree into the sequence needed for inserting revised probabilities into the decision tree. Influence diagrams allow both the problem statement and solution to be captured and treated in the same structure. Since trees expand exponentially as nodes are added, only the first two levels in the solution of Fig. 7 are shown, followed by their expected values. Here, the real market survey raises the value of the original problem by only 1.9 compared to the value of perfect information of 14.7 or only about 11.5%. Perfect information is often not a good guide to the value of real information, see Matheson and Matheson, (2005). Influence diagrams can be used to explore similar questions in much more complex situations. An elaborate example, that illustrates a sequence of more complex but useful influence diagram models of a space program decision, including dealing with perplexing counter-factual probability assessments, appears in Matheson, (1990). Formulating and solving general interventions to observe or control decision situations are also discussed in Matheson and Matheson, (2005).

While influence diagrams can grow large, they usually can be constructed so they fit onto one sheet of paper. A corresponding decision tree, however, would be very unwieldy (it might wrap around the room!) and be difficult to manipulate. Fortunately, methods have been developed for solving influence diagrams directly, without building large trees, making solutions of large influence diagrams practical, see (Olmsted 1983; Shachter 1986, 1988).

Summary of Rules, Conventions, and Issues

The references show many rules for constructing and manipulating influence diagrams, including manipulations that constitute formal proofs, with no

numbers involved. Some rules and terminology are provided here.

An influence diagram usually represents the state of information of an author. In the graphical representation, the author is making assertions about the structure of the problem, usually before assigning the functions and numbers inside the nodes. If information is supplied by several parties, a single author should first review and accept it, then consolidate that information into the influence diagram. Large organizations sometimes designate individual information certifiers, or a small group to perform these functions, who sign off on the content of the influence diagrams, much like an architect or engineer signs blueprints.

One can think of a decision node as a node where the decision maker specifies a choice by a usually degenerate probability distribution. The node could represent a randomized strategy in a gaming context, so the whole diagram can be viewed as one joint probability assignment of the decision maker that has been simplified and specified using the influence diagram structure. Thus, influence diagrams represent a conditioning order for representing a joint probability distribution; they are always directed graphs indicating a particular conditioning order and cannot have directed loops.

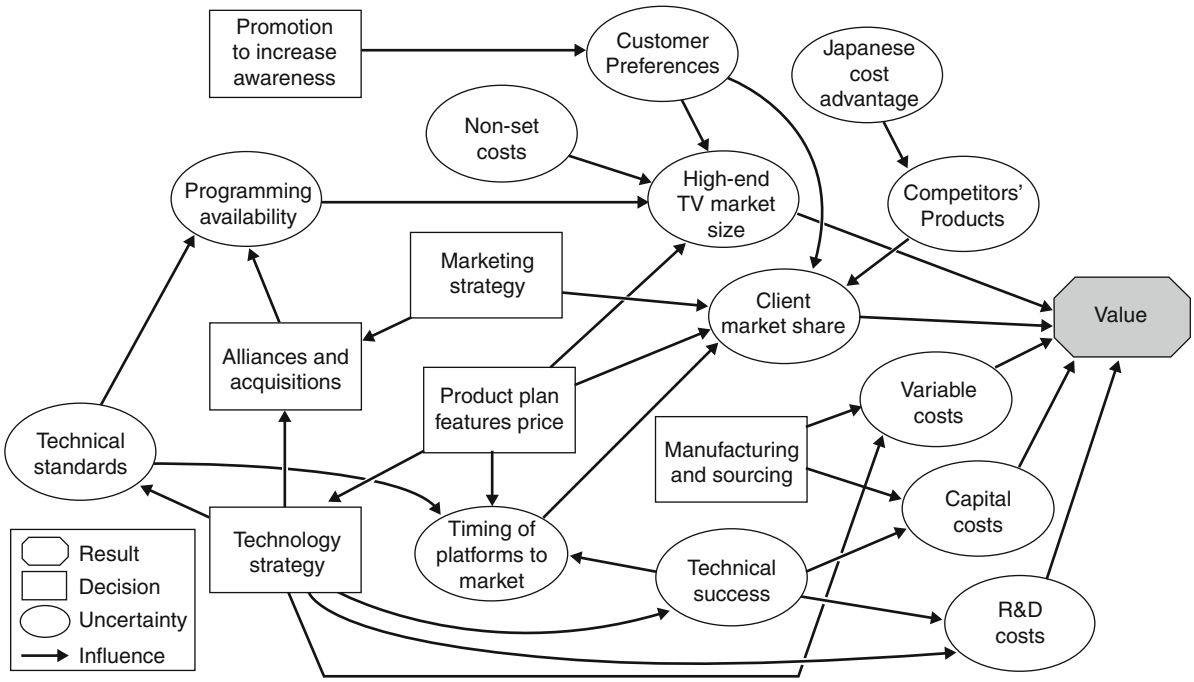
Influence diagrams usually describe a single decision maker who remembers previous decisions. This no forgetting condition means that the decision nodes should be directly ordered by arrows from all preceding decisions, although sometimes implied arrows are left off of the diagram to eliminate clutter. Arrows into decision nodes are informational influences. Arrows into chance nodes are conditioning influences.

The original influence diagram paper did not insist upon having an explicit value node, essentially considering it to be deterministic and in another dimension. But in practice, value nodes are normally used to show what is being maximized, creating what are called decision diagrams. Also, in practice, many deterministic chance nodes surround the value node to link value from various parts of the problem; for example, stating that profit is revenue less cost or revenue is sales volume multiplied by price. Influence diagrams containing a value node and other deterministic relationships are sometimes referred to as value maps. These help to put the main features of the entire decision model into one diagram that can be readily explained and understood. A diagram with only chance nodes is sometimes called a knowledge map with the arrows relevance arrows, see Howard, (1990).

The most important assertions in an influence diagram are given by the lack of possible arrows, as this lack asserts that the author believes that there are structural reasons for a lack of influence (or relevance), and that any subsequent assessment of actual functions or numbers will preserve this property. This feature allows graphical dialog and reasoning about issues that are separate from the particular assessments that might be made later or by different authors who all agree on the structure represented by the diagram. They also allow graphical proofs about features of the diagram that are true no matter what numerical assessments are made. For example, an influence diagram that is a chain of nodes and arrows between each successive node, having no arrows bypassing other nodes, can easily shown to be reversible into the opposite order, while retaining the same features, regardless of what assessments are put into the nodes themselves.

Influence diagrams are often constructed top down. After establishing a value node, the assessor asks the subject, “if you wanted to eliminate some of your uncertainty about value, what question might you ask (of a clairvoyant)?” The subject might answer, “our revenue,” which further breaks down into “the amount sold” and “our realized price.” Asked what information would help reduce those uncertainties, the subject might say, “time of entry of foreign competition” and “U.S. tariff barrier level.” The cost side might be broken into capital cost and variable cost. The process continues until the influence diagram becomes an adequate description of the problem for assessment and analysis.

Influence diagrams can interconnect many decisions and uncertainties to express highly complex problems that would be very difficult to treat with decision trees. A virtual tree always exists as a hypothetical construct, but may be too large to be useful. Decision trees, however, can be useful to display parts of the diagram or the nature of solutions. There are many solution methods that do



Influence Diagrams, Fig. 8 Complex Industrial Influence Diagram

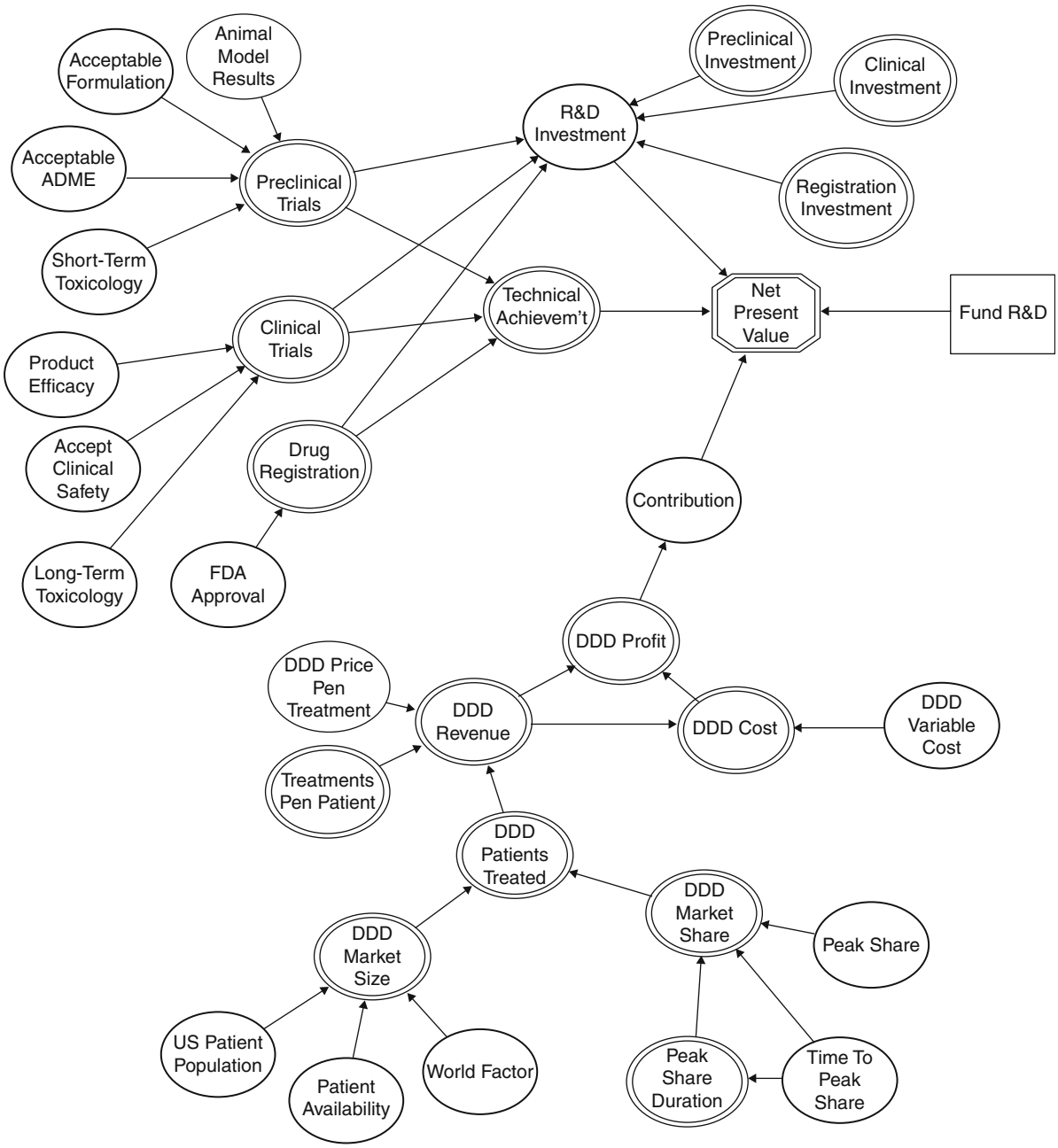
not require developing trees, some for solving decision problems and others specialized to limited situations, such as propagating the impact of information on marginal probabilities.

Adding Complexity: Real-World Influence Diagrams

An example of an influence diagram use to capture and treat a real-world industrial problem is illustrated in Fig. 8. A major international electronics company was concerned about slipping behind in their television and related electronics strategy as the era of HDTV was approaching. The Japanese were already using an early version of HDTV, and an international standards battle was brewing. These standards would influence the strategic position of the company. The company also needed to determine what areas they would keep in-house as proprietary. For example, design and manufacturing technologies, and in what areas they would partner or outsource. In this instance, all of the decision nodes are simultaneous, and the arrows among them have been left off to eliminate clutter. The diagram shows how the company perceived their decisions and the influences on the development of the markets for HDTV and their own participation in it. This work ultimately set their strategy for the following decade. (See Matheson and Matheson 1998) for further discussion).

Another real-world influence diagram, shown here for illustrative purposes is that of a typical drug development, Fig. 9 (SmartOrg 2005). The diagram splits into two: an upper portion that deals with R&D, and a lower portion that deals with the commercial contribution that might accrue if the drug is successfully developed and approved by the FDA for marketing. This diagram is typical of research or development problems, including projects such as oil exploration, or litigation. These problems are all characterized by a development phase, that determines whether the project is commercially feasible, followed by a commercialization phase, that determines the project's contribution if it is commercialized.

Some companies find that parts of an influence diagram repeat themselves in many problems. For example, an oil company investment may often have a portion of the influence diagram representing oil-price economics. This allows the company to do



Influence Diagrams, Fig. 9 A typical Drug Development

a careful assessment once and to distribute the assessed influence diagram to its decision analysts as a starting point for many situations. This process clearly saves labor and assures consistency. But beyond that, most of these companies have a portfolio of investments to make involving the same assessments. In this case the common influence diagram insures consistency and conveys the information needed to determine dependency among the investments, permitting analysis of risk concentration and risk compensation, and providing the analytics needed for diversification and hedging (Matheson 1983).

See

- Bayes Rule
- Bayesian Decision Theory, Subjective Probability, and Utility
- Decision Analysis
- Decision Analysis in Practice
- Decision Trees

References

Howard, R. (1965). Bayesian models for systems engineering. *IEEE Transactions on Systems Science and Cybernetics*, Vol. SSC-1 No. 1, 36–40.

Howard, R. A. (1989). Knowledge maps. *Management Science, 35*(8), 903–922.

Howard, R. A. (1990). From influence to relevance to knowledge. In R. M. Oliver, & J. Q. Smith (Eds.), *Influence diagrams, belief nets and decision analysis*, *Proceedings, May 1988 conference* (pp. 3–23). New York: John Wiley & Sons.

Howard, R., & Matheson, J. (1983). Influence diagrams. In R. A. Howard & J. E. Matheson (Eds.), *Readings on the principles and applications of decision analysis* (pp. 719–763). Menlo Park, CA: Strategic Decisions Group.

Howard, R., & Matheson, J. (2005a). Influence diagrams. Reprinted in the special issue on graphical methods. *Decision Analysis*, 2 (3), 127–143.

Howard, R. A., & Matheson, J. E. (2005b). Influence diagrams retrospective. Special issue on graphical methods. *Decision Analysis*, 2 (3), 144–147.

Howard, R. A., Matheson, J. E., Merkhofer, M. W. (lee), Miller, A. C., & Warner North, D. (2006). Comment on influence diagram retrospective. *Decision Analysis*, 3 (2), 117–119.

Howard, R., Matheson, J., Merkhofer, M., Miller III, A., and Rice, T. (1976). *Development of automated aids for decision analysis*. DARPA Contract MDA 903-74-C-0240. Menlo Park, CA: SRI International.

Matheson, J. (1983). Managing the corporate business portfolio. In R. A. Howard & J. E. Matheson (Eds.), *Readings on the principles and applications of decision analysis* (pp. 311–326). Menlo Park, CA: Strategic Decisions Group.

Matheson, J. (1990). Using influence diagrams to value information and control. In R. M. Oliver, & J. Q. Smith (Eds.), *Influence diagrams, belief nets and decision analysis*. Proceedings, May 1988 conference (pp. 25–63). New York: John Wiley & Sons.

Matheson, D., & Matheson, J. (2005). Describing and valuing interventions that observe or control decision situations. In the special issue on graphical methods. *Decision Analysis* 2 (3), 165–181.

Matheson, D., & Matheson, J. (1998). *The smart organization, creating value through strategic R&D*. Cambridge, MA: Harvard Business School Press.

Neapolitan, R. (2004). *Learning Bayesian networks*. Englewood Cliffs, NJ: Prentice Hall.

Olmsted, S. (1983). *On representing and solving decision problems*. Ph.D. thesis, EES Department, Stanford University, Stanford, CA.

Pearl, J. (1986). Fusion, propagation, and structuring in belief networks. *Artificial Intelligence, 29*, 24l–288l.

Pearl, J. (2005). Influence diagrams – historical and personal perspectives. *Decision Analysis, 2*(4), 232–234.

Shachter, R. (1986). Evaluating influence diagrams. *Operations Research, 34*(6), 871–882.

Shachter, R. (1988). Probabilistic inference and influence diagrams. *Operations Research, 36*(4), 589–604.

SmartOrg. (2005). *Dynamic depression drug: Tutorial*. Menlo Park, CA: SmartOrg, Inc.

Spetzler, C., & Staël von Holstein, C. (1975). Probability encoding in decision analysis. *Management Science, 22*, 340–358.

World Economic Forum. (2011). *Global risks 2011: Sixth Edition, an initiative of the risk response network*. January, Cologny, Geneva.

Information Systems and Database Design in OR/MS

Heiner Müller-Merbach
Technische Universität Kaiserslautern, Kaiserslautern, Germany

Introduction

There are many close relations between information systems, database structures, and operations research (OR). The models and algorithmic procedures of OR are becoming more integrated parts of information systems. The task of OR may continually shift towards the comprehensive design of information systems, database structures included.

Architechture of Comprehensive Information Systems

Traditional data processing was based on collections of individual programs, separated from one another, each with its own individual data organization. Similarly, the characteristic OR packages were stand-alone solutions for singular types of problems, be it mathematical programming, network analysis, and simulation or even more specialized packages for the knapsack problem, traveling salesman problem, set covering problem, etc.

Future information systems, in contrast, will have a comprehensive architecture. The vast majority of data will be stored and maintained centrally, on a data management computer, or on a network of such computers. Most of the programs will mainly process such centralized data and the programs themselves will be available from the comprehensive information system.

A particular feature of the comprehensive information systems is the client–server structure, that is, a network with a huge number of clients (client computers) being provided with data and programs from a server (server computer) or networks of servers.

Relational Databases

The design of such comprehensive information systems and their databases requires standards, in particular those for data structures. A quite common standard today is that of relational databases, such as designed by Codd (1970). The main principles of relational databases are (in non-technical terms):

- All the information is organized in terms of attributes to entity sets. Entity sets are collections of entities with identical attributes — but individual attribute values.
- There is no hierarchy between the entity sets. All the entity sets are at the same level and allow for immediate access. However, it is sometimes advantageous to distinguish between elementary and connecting entity sets. The first ones are self-contained, while the latter ones connect other entity sets and, therefore, depend partly on them.
- Any information is only stored once and no redundancy is allowed. Any attribute, therefore, has to be attached to its corresponding (elementary or connecting) entity set. This is the essence of the normalization concept of relational database structures.

Models and Databases

There exists a narrow correspondence between mathematical models and relational database structures. Indices of a mathematical model indicate the individual entities of an entity set, single indices those of elementary, multiple indices those of connecting entity sets. The constants and variables of a mathematical model correspond with the attributes of the entity sets (Müller-Merbach 1983, 1989; Geoffrion 1989).

This correspondence can easily be shown by a production function that connects the quantities of production factors with the quantities of products:

$$r_j = \sum_k a_{jk}\, x_k$$

with j and k indicating the entities of the entity sets **FACTOR**(j) and **PRODUCT**(k), respectively, and

r_j = quantity of production factor j required

x_k = quantity of product k to be produced

a_{jk} = production coefficient, representing the quantity of factor j required per unit of product k.

The relational database structure corresponding with the production function is given in Fig. 1. There are the elementary entity sets **FACTOR**(j) and **PRODUCT**(k), as well as the connecting entity set $\mathbf{F} \times \mathbf{P}(j, k)$. This database structure is to be considered as a subset of the comprehensive database of the corresponding enterprise with many more entity sets and many more attributes to the entity sets.

Any mathematical model (be it from OR, statistics, etc.) should have such an immediate correspondence with the database. The entity sets correspond with the mathematical indices and the attributes to the entity sets correspond with the constants and variables of the model.

Therefore, model design and database design follows the same logical structure. Either one can proceed the other. However, normally database design is prior to model design and the attributes required for a model can be derived from the data-base. Should, however, the attributes required for a model not be available in the database, an appropriate extension of the database may become necessary.

Mnemonic Notation

Large-scale mathematical models and — even more so — databases in general tend to cover huge numbers of entity sets and attributes. In order to cope with them, a mnemonic notation of the attributes is useful.

The notation should (i) refer to the entity set, (ii) specify the content of the attribute, and (iii) indicate the formal property of the attribute.

Considered be a two-stage production and cost function, respectively, connecting the three elementary entity sets **LABOR**, **MACHINE**, and **PRODUCT** (Fig. 2). The indices indicate the qualification class of labor (i), the machines (j) and the products (k). All the attributes of the entity set **LABOR** start with an L, the others with an M or a P, respectively, referring to the entity sets. The content is represented by a Q (quantity), a T (time required), or a C (cost) in the second position. The third letter indicates constants (C), variables (V) and other formal properties such as discrete variables (D), Boolean variables (B), etc.

Thus, the constants vector PQC_k represents the known (therefore, C) quantities (Q) of the single products (P). The variables vector MTV_j represents the times (T) of the machines (M) required for producing the given quantities of the products. The variables vector LQV_i stands for the quantity (Q) of labor (L) necessary for running the machines.

In addition, the production coefficients have to be introduced. They are attributes of the dependent entity sets, connecting the elementary entity sets **MACHINE** and **PRODUCT** as well as **LABOR** and **MACHINE**. It is convenient that the attributes of the dependent entity sets refer immediately to the attributes of the elementary entity sets. Thus, the constants matrix $MTPQC_{jk}$ represents the machine time (MT) per unit of the product quantities (PQ). This leads immediately to the production function for the machine times required for the given product quantities:

$$MTV_j = \sum_k MTPQC_{jk} PQC_K.$$

In a similar way, the quantity of labor hours (LQ) per unit of the machine times (MT) is represented by the constants matrix $LQMTC_{ij}$, the basis for the production function for the labor quantities required for the computed machine times:

$$LQV_i = \sum_j LQMTC_{ij} MTV_j.$$

The cost functions (here only labor costs) are dual to the production functions. They use the same production coefficients matrices as the production functions, but the attributes of the elementary entity sets are different:

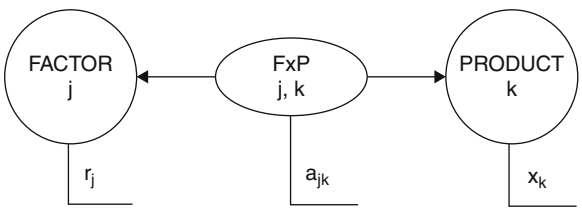
LCC_i = cost of quantity unit of labor (qualification class i)

MCV_j = labor cost per time unit of machine j

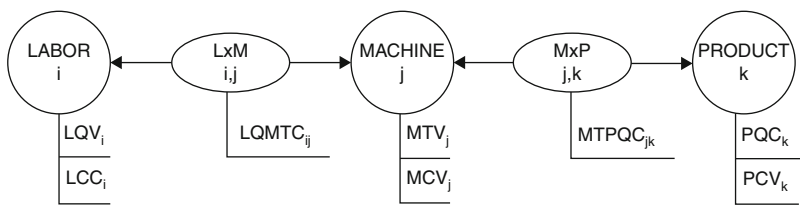
PCV_k = labor cost per quantity unit of product k.

By the first cost function, the labor costs are assigned to the machines:

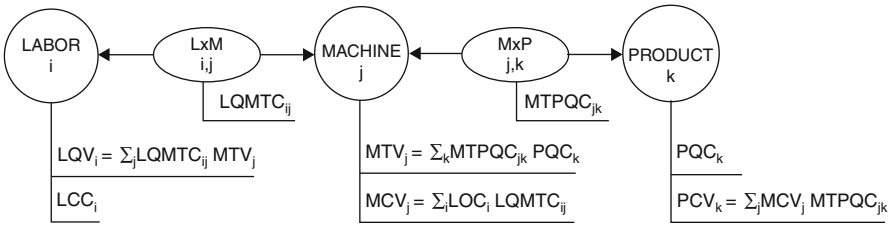
$$MCV_j = \sum_i LCC_i LQMTC_{ij}.$$



Information Systems and Database Design in OR/MS, Fig. 1 Relational database structure for a production function (The attributes are attached to the entity sets **FACTOR**, **PRODUCT**, and **F** × **P**.)



Information Systems and Database Design in OR/MS, Fig. 2 Relational database structure for a two-stage production function and cost function (The mnemonic attributes are attached to the elementary entity sets **LABOR**, **MACHINE**, and **PRODUCT**, as well as to the dependent entity sets **L** × **M** and **M** × **P**.)



Information Systems and Database Design in OR/MS, Fig. 3 Object-oriented database structure for two-stage production function and cost-function (The mnemonic attributes and the functions for the variables are attached to the elementary entity sets **LABOR**, **MACHINE**, and **PRODUCT**, as well as to the dependent entity sets **L** × **M** and **M** × **P**.)

By the second cost function, the resulting labor costs per machine time unit are assigned to the products:

$$PCV_k = \sum_j MCV_j MTPQC_{jk}.$$

Object-Oriented Modeling

There is a tendency from relational databases and modeling towards object-oriented databases and modeling. One of the object-oriented features is the integration of functions and data. Even if there is no unique standard as yet for object-oriented databases, the idea of object-oriented mathematical models can be presented (Fig. 3), with all the production functions integrated into the database structure.

Advantage of Integration

The integration of models and algorithmic procedures of OR into comprehensive information systems has many convenient properties. The main advantage is: the data required for a model can immediately be taken from the centralized database. The results derived from the model can immediately be transferred back to the database and is then available to other users.

See

- ▶ Model Management
- ▶ Structured Modeling
- ▶ Systems Analysis

References

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM, 13*, 377–387.

Geoffrion, A. M. (1989). Computer-based modeling environments. *European Journal of Operational Research, 41*, 33–43.

Müller-Merbach, H. (1983). Model design based on the systems approach. *Journal of the Operational Research Society, 34*, 739–751.

Müller-Merbach, H. (1989). Database-oriented design of planning models. *IMA Journal of Mathematics Applied in Business and Industry, 2*, 141–155.

Information Technology Benefits

Douglas A. Samuelson
Infologix, Inc., Annandale, VA, USA

Many companies and other organizations have adopted new information technology, expending considerable effort and resources to do so. Understandably, managers wish to evaluate whether the organization's benefits from these new technologies exceeded the cost of acquisition and adoption. What was in 2000 a large and growing literature on this subject seems to have reached a plateau, as early studies questioning the value of information technology collided with the reality of growth and profitability for companies that employed information technology heavily. Methods to evaluate utility and appropriateness of information technology have not kept pace with the technological advances. Good methods of evaluation do exist, but they are not prominent in the literature. Using such methods and developing new ones represent an important challenge and opportunity for OR/MS analysts.

Much of the earlier published work on the subject, especially in business journals, focused on examining whether firms' profits, typically reported quarterly, rose substantially in a fairly short time after adoption of new information technology. This approach proved uninformative and often misleading, as corporate tax accounting practices tend to strive for smallish, level net profits. Hence, the profits metric obscures the true value of information technology investments.

The high-tech boom and bust of 1999–2000 raised awareness of the disparities between the claims and reality for many such investments. Perceptive analysts began to ask not whether information technology pays off in general, but what factors most influence whether it pays off in particular cases. This area of study, most prominently in the software engineering literature, as typified and summarized by Gilb (1988), had been ongoing for some time, but remained unfamiliar many business analysts. Gilb emphasized the need to develop quantified requirements, that is, clear objectives that could be measured numerically and unambiguously, and to monitor continuously whether the project met intermediate milestones as captured by the metrics. Gilb (2005) expanded on these principles, summarizing additional lessons learned and elaborating on the method and some case studies to support his assessment, with little change in the fundamental ideas.

Strassmann (1990) asserted that growth of assets and return on investment, that are more promising measures of benefit, also have drawbacks: return on investment is hard to define precisely and, in most organizations, hard to obtain; growth of assets is also subject to problems of definition, although not to the same extent. Instead, he suggested computing the total expenditure on all forms of management, which in his view includes most support services of any kind, and computing the company's return on management. This is usually not possible without extensive internal auditing of the company; in some cases, Strassmann reported, he arrived at a first approximation of the percentage of total company expenditures devoted to management by counting the number of windows in the company's headquarters that correspond to employees involved in some level of management and support.

Brynjolfsson and Hitt (1998) and Brynjolfsson and Yang (1996, 1996) approached the conceptualization somewhat differently, asserting that information technology and the attendant restructuring of organizational processes create a valuable intangible asset. The resources expended to maintain and protect this asset — or by competitors to diminish its value or duplicate it — can then be used as an indication of its perceived value. This approach has the additional advantage of being applicable to non-profit organizations, rather than just corporations.

Shapiro and Varian (1999) gave a considerably more thorough discussion of the economics of value of information. They argued strongly that accurate and timely information has value which may be considerably greater than one's first attempts at measurement would suggest. If a company uses its information advantage to achieve customer lock-in and, thereby, enables itself to price its products and services, within selected market niches at a high profit margin, it obtains a large return on its information investment — a return which would escape many approaches to measurement, as the benefit is confounded with that of general marketing. Similarly, accumulating know-how which reduces costs of production is confounded with other aspects of cost control.

Another apparently popular method of evaluation is to survey users of the technology. These surveys need not and, in fact, should not be restricted to simple "Do you like it?" items. Subject only to the respondents' limits on the time they are willing to spend, analysts can ask numerous specific questions about how certain tasks, such as disseminating background materials for an important decision, are performed before and after the introduction of new information technology.

Unfortunately, even the best of these surveys are of limited value. Respondents asked how much time they spend finding lost information, for example, may not know, as many of them delegate such tasks to others. Individuals, regardless of level, rarely see group costs. Even more fundamental is the Baywatch Syndrome, which is that people do not realize what they know. That is, in the few years after the show left the air in the U.S., many people who claimed to be unfamiliar with the television series Baywatch were nevertheless able to identify readily a still photograph from the show, rated as the most widely watched show in the world for several years in the mid-1990s. What happened, however, is that much of the exposure took place via channel surfing, or in some other rather haphazard fashion, or via promotions for the show during other programs. For many people, the viewing experience never entered

into their central, attention-directing mental activity, so they did not recall it without specific prompting.

A similar phenomenon occurs when people adopt new information technology. Like a manual gearshift in a car, the information technology slips out of the person's central mental activity and into semi-automatic behavior — and such slippage precisely defines adoption. Once this has occurred, the respondent is less and less likely to recall specifics of use and value of the information technology, as the person is using it with less and less conscious, directed thought about it.

This change in attention and in involvement of higher-order thinking also helps to explain a phenomenon commonly encountered in surveys: respondents state that the new technology is of little value, and one of the primary reasons they give — often with some irritation — is that it breaks too often. Upon closer examination, these responses are contradictory: if the technology is truly of little value, one would not care how often it breaks. Such responses should be interpreted as positive with regard to the perceived value (or at least potential value) of the technology, though clearly not positive with regard to satisfaction and ease of use.

The most useful approach in evaluation, therefore, is to utilize the technology itself to make unobtrusive measurements of actual usage and patterns of use. Most Web servers can readily accommodate software which measures numbers of hits on various resources, numbers of logins and duration of sessions (for systems which have a formal login and logout as part of their access protocols), and similar statistics. With more effort, and custom software, one can also track discussion threads by topic, which makes it possible to answer such questions as how long it takes to get a specific type of order filled, a specific type of question answered, or a specific directive carried out and verified as done. If these measures then change shortly after additions or modifications to the information system, association is easy to identify, and the causal connection is easier to assert.

Finally, some organizations genuinely experience disappointing results with new information technology because they do not change their work processes to take advantage of it. Attempting to maintain hierarchical structures and traditional controls greatly reduces the organization's ability to use the information technology to support coherent, simultaneous activities which do not require direct coordination — the types of activity for which enriched information systems are most critical. As Arquilla and Ronfeldt (1997) pointed out in their review of military information warfare, "The information revolution favors and strengthens networks, while it erodes hierarchies.... Hierarchies have a difficult time fighting networks."

An organization, therefore, which adopts new information technology but does not have clear, measurable objectives and the commitment to change its work processes, will be highly unlikely either to achieve significant benefits or to be able to measure the effects accurately. The proper evaluation of the benefits of information technology requires not only assessment of use of the technology, but also a clear statement of the organization's objectives, realistic appraisal of how well those objectives are being met before and after the technology is introduced, and unflinching accounting for the extent to which organizational, structural, and political resistance to change are affecting the results.

Hubbard (2010) gave a number of good examples of better ways to measure seemingly intangible aspects of organizational performance, some drawn from experience with information technology projects. Hubbard (2009) also noted explicitly the tendency of managers to over-value some techniques for decision-making. According to surveys he conducted, some popular methods, such as balanced scorecards and some applications of the Analytic Hierarchy Process (AHP), consistently did better at raising managers' confidence in decisions than in improving outcomes. He concluded that methods of decision making should be subjected to the same rigorous quantitative evaluation as other subject areas – if anything, even more so – with the purpose of driving out underperforming methods.

Hubbard also noted that metrics' value is not inherent, but depends on what is already known. An approach he invented (or at least stated cogently and named), Applied Information Economics, addresses this valuation by calculating the Expected Value of Perfect Information (EVPI), that is, the risk associated with the decision if a selected data element were known with certainty, and devoting resources to those areas of uncertainty (regarding data elements) in which more information would most decrease the decision risk. For example, in information technology investments, his studies indicated that managers generally knew the

costs of information projects much better than the benefits. Hence, they tended to underestimate one of the biggest risks: cancellation of the project partway through because of some combination of changing management objectives, changing perception of requirements, and frustration with the lack of tangible progress. Thus, Hubbard arguing from probabilistic-decision making, reinforces Gilb's emphasis on the importance of well-defined quantitative metrics that are systematically and periodically reviewed throughout the project's life.

See

- ▶ Computer Science and Operations Research Interfaces
- ▶ Information Systems and Database Design in OR/MS
- ▶ Model Management
- ▶ Systems Analysis

References

Arquilla, J., & Ronfeldt, D. (1997). The advent of netwar. In J. Arquilla & D. Ronfeldt (Eds.), *Athena's camp: Preparing for conflict in the information age*. Santa Monica, CA: RAND Corporation Press.

Brynjolfsson, E., & Hitt, L. M. (1998). Beyond the productivity paradox: Computers are the catalyst for bigger changes. *Communications Association for Computing Machinery, 41*(8), 49–55.

Brynjolfsson, E., & Yang, S. (1996). Information technology and productivity: A review of the literature. *Advances in Computers, 43*, 179–214.

Brynjolfsson, E., & Yang, S. (1997). The intangible costs and benefits of computer investments: Evidence from the financial markets. In *Proceedings international conference on information systems*, Atlanta, GE

Gilb, T. (1988). *Principles of software engineering management*. Harlow, UK: Pearsons Education.

Gilb, T. (2005). *Competitive engineering*. Burlington, MA: Elsevier.

Hubbard, D. (2009). *The failure of risk analysis: Why it's broken and how to fix it*. Hoboken, NJ: Wiley.

Hubbard, D. (2010). *How to measure anything: Finding the value of intangibles in business* (2nd ed.). Hoboken, NJ: Wiley.

Shapiro, C., & Varian, H. (1999). *Information rules: A strategic guide to the network economy*. Boston: Harvard Business School Press.

Strassmann, P. (1990). *The business value of computers*. New Canaan, CT: Information Economics Press.

INFORMS

▶ Institute for Operations Research and the Management Sciences (INFORMS)

Initial Feasible Solution

▶ First Feasible Solution

Input Process

The stochastic point process representing some aspect of customers actually entering a queueing system (or nodal part of one) or some aspect of the state of the node at the instant of input, with points representing the instants of entrance. For example, in finite capacity queues, and $(\boldsymbol{X}^a, \boldsymbol{T}^a)$ process has the $\boldsymbol{X}^a$ process as a sequence of 1s and 0s representing whether the queue is full or not at an arrival and the $\boldsymbol{T}^a$ process represents the times of arrivals. The subset of the $\boldsymbol{T}^a$ process for which $\boldsymbol{X}^a = 0$ represents the set of arrival epochs at which a customer actually enters the node, while the $\boldsymbol{T}^a$ for which $\boldsymbol{X}^a = 1$ represents the set of arrival times at which customers do not gain access to the node but overflow.

See

- ▶ Arrival Process
- ▶ Networks of Queues
- ▶ Queueing Theory

Input–Output Analysis

The economic theory developed by the economist W.W. Leontief to study a national economy. The approach requires the development of an input–output table (matrix) in which the coefficients in a row indicate how much of the industry designated by that row is required to produce a unit of output for itself and all other industries, and the coefficients in

a column represent the amounts of each industry required to produce one unit of output for the industry designated by that column. Under the assumption that the input–output coefficients are stable over the near future and reflect a constant return to scale (linear) relationship, a square set of equations can be established to determine production levels for the industries that meets projected demand.

See

► Input–Output Coefficients

Input–Output Coefficients

For some linear programming and other production problems, the $\boldsymbol{A} = (a_{ij})$ coefficients of the constraints $\boldsymbol{Ax} = \boldsymbol{b}$ can be interpreted as the amount of resource i required (input) to produce one unit of product j (output). More generally, an input–output matrix of American industries formed the bases of the economist Leontief's contribution to economic theory.

See

► Activity-Analysis Problem
► Input–Output Analysis

Insensitivity

A property of queueing systems wherein some measure of effectiveness does not depend on a particular distribution assumption except through its mean value. The classical example is the Erlang loss call formula in the multi-server M/G/c/c queue that depends on the service-time process only through its mean value.

See

► Erlang B Formula
► Queueing Theory

Institute for Operations Research and the Management Sciences (INFORMS)

The main organization for operations research and the management sciences in the United States begun officially on January 1, 1995 upon the merger of the Operations Research Society of America (ORSA) and The Institute of Management Sciences (TIMS).

Integer and Combinatorial Optimization

Karla L. Hoffman[1] and Ted K. Ralphs[2]
[1]George Mason University, Fairfax, VA, USA
[2]Lehigh University, Bethlehem, PA, USA

Introduction

Integer optimization problems are concerned with the efficient allocation of limited resources to meet a desired objective when some of the resources in question can only be divided into discrete parts. In such cases, the divisibility constraints on these resources, which may be people, machines, or other discrete inputs, may restrict the possible alternatives to a finite set. Nevertheless, there are usually too many alternatives to make complete enumeration a viable option for instances of realistic size. For example, an airline may need to determine crew schedules that minimize the total operating cost, an automotive manufacturer may want to determine the optimal mix of models to produce in order to maximize profit, or a flexible manufacturing facility may want to schedule production for a plant without knowing precisely what parts will be needed in future periods. In today's changing and competitive industrial environment, the difference between ad hoc planning methods and those that use sophisticated mathematical models to determine an optimal course of action can determine whether or not a company survives.

A common approach to modeling optimization problems with discrete decisions is to formulate them as mixed integer optimization problems. This entry focuses on problems in which the functions required to represent the objective and constraints are additive, i.e., linear functions. Such a problem is called a mixed

integer linear optimization problem (MILP) and its general form is

$$\max \sum_{j\in B} c_j x_j + \sum_{j\in I} c_j x_j + \sum_{j\in C} c_j x_j \tag{1}$$

$$\text{subject to } \sum_{j\in B} a_{ij} x_j + \sum_{j\in I} a_{ij} x_j + \sum_{j\in C} a_{ij} x_j \begin{Bmatrix} \le \\ = \\ \ge \end{Bmatrix} b_i \quad \forall i \in M, \tag{2}$$

$$l_j \le x_j \le u_j \quad \forall j \in N = B \cup I \cup C, \tag{3}$$

$$x_j \in \{0, 1\} \quad \forall j \in B, \tag{4}$$

$$x_j \in \mathbb{Z} \quad \forall j \in I, \text{and} \tag{5}$$

$$x_j \in \mathbb{R} \quad \forall j \in C. \tag{6}$$

A solution to (1)–(6) is a set of values assigned to the variables $x_j, j \in N$. The objective is to find a solution that maximizes the weighted sum (1), where the coefficients c_j, $j \in N$ are given. B is the set of indices of binary variables (those that can take on only values 0 or 1), I is the set of indices of integer variables (those that can take on any integer value), and C is the set of indices of continuous variables. As indicated above, each of the first set of constraints (2) can be either an inequality constraint ("$\le$" or "$\ge$") or an equality constraint ("$=$"). The data l_j and u_j are the lower- and upper-bound values, respectively, for variable $x_j, j \in N$.

This general class of problems has many important special cases. $B = I = \emptyset$ gives what is known as a linear optimization problem (LP). If $C = I = \emptyset$, then the problem is referred to as a (pure) binary integer linear optimization problem (BILP). Finally, if $C = \emptyset$, the problem is called a (pure) integer linear optimization problem (ILP). Otherwise, the problem is simply a MILP. Throughout this discussion, refer to the set of points satisfying (1)–(6) as $\mathcal{S}$, and the set of points satisfying all but the integrality restrictions (4)–(5) as $\mathcal{P}$. The problem of optimizing over $\mathcal{P}$ with the same objective function as the original MILP is called the LP relaxation and arises frequently in algorithms for solving MILPs.

A class of problems closely related to BILPs are the combinatorial optimization problems (COPs). A COP is defined by a ground set $\mathcal{E}$, a set $\mathcal{F}$ of subsets of ε that are called the feasible subsets, and a cost c_e associated with each element $e \in \mathcal{E}$. Each feasible subset $F \in \mathcal{F}$ has an associated (additive) cost taken to be $\Sigma_{e \in F} c_e$. The goal of a COP is find the subset $F \in \mathcal{F}$ of minimum cost. The set $\mathcal{F}$ can often be described as the set of solutions to a BILP by associating a binary variable x_e with each member e of the ground set, indicating whether or not to include it in the selected subset. For this reason, combinatorial optimization and integer optimization are closely related and COPs are sometimes informally treated as being a subclass of MILPs, though there are COPs that cannot be formulated as MILPs.

Solution of an MILP involves finding one or more best (optimal) solutions from the set $\mathcal{S}$. Such problems occur in almost all fields of management (e.g., finance, marketing, production, scheduling, inventory control, facility location and layout, supply chain management), as well as in many engineering disciplines (e.g., optimal design of transportation networks, integrated circuit design, design and analysis of data networks, production and distribution of electrical power, collection and management of solid waste, determination of minimum energy states for alloy construction, planning for energy resource problems, scheduling of lines in flexible manufacturing facilities, and design of experiments in crystallography).

This entry gives a brief overview of the related fields of integer and combinatorial optimization. These fields have by now accumulated a rich history and a rich mathematical theory. Texts covering the theory of linear and integer linear optimization include those of Bertsimas and Weismantel (2005), Chvátal (1983), Nemhauser and Wolsey (1988), Parker and Rardin (1988), Schrijver (1986), and Wolsey (1998). Overviews of combinatorial optimization are provided by Papadimitriou and Steiglitz (1982) and Schrijver (2003). Jünger et al. (2010) have produced a marvelous and comprehensive volume containing an overview of both the history and current state of the art in integer and combinatorial optimization.

Applications

This section describes some classical integer and combinatorial optimization models to provide an overview of the diversity and versatility of this field.

Knapsack Problems

Suppose one wants to fill a knapsack that has a weight capacity limit of W with some combination of items from a list of n candidates, each with weight w_i and value v_i, in such a way that the value of the items packed into the knapsack is maximized. This problem has a single linear constraint (that the weight of the items selected not exceed W), a linear objective function (to maximize the sum of the values of the items in the knapsack), and the added restriction that each item either be in the knapsack or not—it is not possible to select a fractional portion of an item. For solution approaches specific to the knapsack problem, see Martello and Toth (1990).

Although this problem might seem too simplistic to have many practical applications, the knapsack problem arises in a surprisingly wide variety of fields. For example, one implementation of the public-key cryptography systems that are pervasive in security applications depends on the solution of knapsack problems to determine the cryptographic keys (Odlyzko 1990). The system depends on the fact that, despite their simplicity, some knapsack problems are extremely difficult to solve.

More importantly, however, the knapsack problem arises as a substructure in many other important combinatorial problems. For example, machine-scheduling problems involve restrictions on the capacities of the machines to be scheduled (in addition to other constraints). Such a problem involves assigning a set of jobs to a machine in such a way that the capacity constraint is not violated. It is easy to see that such a constraint is of the same form as that of a knapsack problem. Often, a component of the solution method for problems with knapsack constraints involves solving the knapsack problem itself, in isolation from the original problem (see Savelsbergh (1997)). Another important example in which knapsack problems arise is the capital budgeting problem. This problem involves finding a subset of the set of (possibly) thousands of capital projects under consideration that will yield the greatest return on investment, while satisfying specified financial, regulatory, and project relationship requirements (Markowitz and Manne 1957; Weingartner 1963). Here also, the budget constraint takes the same form as that of the knapsack problem.

Network and Graph Problems

Many optimization problems can be represented by a network, formally defined as a set of nodes and a set of arcs (unidirectional connections specified as ordered pairs of nodes) or edges (bidirectional connections specified as unordered pairs of nodes) connecting those nodes, along with auxiliary data such as costs and capacities on the arcs (the nodes and arcs together *without* the auxiliary data form a graph). Solving such network problems involves determining an optimal strategy for routing certain commodities through the network. This class of problems is thus known as network flow problems. Many practical problems arising from physical networks, such as city streets, highways, rail systems, communication networks, and integrated circuits, can be modeled as network flow problems. In addition, there are many problems that can be modeled as network flow problems even when there is no underlying physical network. For example, in the assignment problem, one wishes to assign people to jobs in a way that minimizes the cost of the assignment. This can be modeled as a network flow problem by creating a network in which one set of nodes represents the people to be assigned, and another set of nodes represents the possible jobs, with an arc connecting a person to a job if that person is capable of performing that job. A general survey of applications and solution procedures for network flow problems is given by Ahuja et al. (1993).

Space-time networks are often used in scheduling applications. Here, one wishes to meet specific demands at different points in time. To model this problem, different nodes represent the same entity at different points in time. An example of the many scheduling problems that can be represented as a space-time network is the airline fleet assignment problem, which requires that one assign specific planes to prescheduled flights at minimum cost (Abara 1989; Hane et al. 1995). Each flight must have one and only one plane assigned to it, and

a plane can be assigned to a flight only if it is large enough to service that flight and only if it is on the ground at the appropriate airport, serviced and ready to depart when the flight is scheduled for takeoff. The nodes represent specific airports at various points in time and the arcs represent the flow of aircraft of a variety of types into and out of each airport. There are layover arcs that permit a plane to stay on the ground from one time period to the next, service arcs that force a plane to be out of duty for a specified amount of time, and connecting arcs that allow a plane to fly from one airport to another without passengers.

A variety of important combinatorial problems are graph-based, but do not involve flows. Such graph-based combinatorial problems include the node-coloring problem, the objective of which is to determine the minimum number of colors needed to color each node of a graph in order that no pair of adjacent nodes (nodes connected by an edge) share the same color; the matching problem, the objective of which is to find a maximum weight collection of edges such that each node is incident to at most one edge; the maximum clique problem, the objective of which is to find the largest subgraph of the original graph such that every node is connected to every other node in the subgraph; and the minimum cut problem, the objective of which is to find a minimum weight collection of edges that (if removed) would disconnect a set of nodes s from a set of nodes t.

Although these graph-based combinatorial optimization problems might appear, at first glance, to be interesting only from a mathematical perspective and to have little application to the decision-making that occurs in management or engineering, their domain of application is extraordinarily broad. The four-color problem, e.g., which is the question of whether a map can be colored with four colors or less, is a special case of the node-coloring problem. The maximum clique problem has important implications in the growing field of social network analysis. The minimum cut problem is used in analyzing the properties of real-world networks, such as those arising in communications and logistics applications.

Location, Routing, and Scheduling Problems

Many network-based combinatorial problems involve finding a route through a given graph satisfying specific requirements. In the Chinese postman problem, one wishes to find a shortest walk (a connected sequence of arcs) through a network such that the walk starts and ends at the same node and traverses every arc at least once (Edmonds and Johnson 1973). This models the problem faced by a postal delivery worker attempting to minimize the number of traversals of each road segment on a given postal route. If one instead requires that each node be visited exactly once, the problem becomes the notoriously difficult traveling salesman problem (Applegate et al. 2006). The traveling salesman problem has numerous applications within the routing and scheduling realm, as well as in other areas, such as genome sequencing (Avner 2001), the routing of SONET rings (Shah 1998), and the manufacturing of large-scale circuits (Barahona et al. 1988; Ravikumar 1996). The well-known vehicle routing problem is a generalization in which multiple vehicles must each follow optimal routes subject to capacity constraints in order to jointly service a set of customers (Golden et al. 2010).

A typical scheduling problem involves determining the optimal sequence in which to execute a set of jobs subject to certain constraints, such as a limited set of machines on which the jobs must be executed or a set of precedence constraints restricting the job order (see Applegate and Cook (1991)). The literature on scheduling problems is extremely rich and many variants of the basic problem have been suggested (Pinedo 2008). Location problems involve choosing the optimal set of locations from a set of candidates, perhaps represented as the nodes of a graph, subject to certain requirements, such as the satisfaction of given customer demands or the provision of emergency services to dispersed populations (Drezner and Hamacher 2004). Location, routing, and scheduling problems all arise in the design of logistics systems, i.e., systems linking production facilities to end-user demand points through the use of warehouses, transportation facilities, and retail outlets. Thus, it is easy to envision combinations of these classes of problems into even more complex combinatorial problems and much work has been in this direction.

Packing, Partitioning, and Covering Problems

Many practical optimization problems involve choosing a set of activities that must either cover certain requirements or must be packed together so as

not to exceed certain limits on the number of activities selected. The airline crew scheduling problem, e.g., is a covering problem in which one must choose a set of pairings (a set of flight legs that can be flown consecutively by a single crew) that cover all required routes (Hoffman and Padberg 1993; Vance et al. 1997). Alternatively, an example of a set packing problem is a combinatorial auction (Cramton et al. 2006). The problem is to select subsets of a given set of items that are up for auction in such a way that each item is included in at most one subset. This is the problem faced by an auctioneer in an auction in which bidders can bid on sets of items rather than just single items. If one requires that all items be sold, then the auctioneer's problem becomes a partitioning problem. There are a variety of languages that allow users to express the interrelationship among their bids. Such languages (e.g., "OR," "XOR," "ORofXOR," "XORofOR") create a somewhat different structure to the combinatorial problem.

In the above examples, the coefficients in constraints (2) are either zero or one and all variables are binary. The variables represent the choice of activities, while each constraint represents either a covering ("$\geq$"), packing ("$\leq$"), or partitioning ("$=$") requirement. In many cases, these problems can be easily interpreted by thinking of the rows as a set of items to be allocated or a set of activities to be undertaken and the columns as subsets of those items/activities. The optimization problem is then to find the best collection of subsets of the activities/items (columns) in order to cover/partition/pack the row set. Surveys on set partitioning, covering, and packing are given in Balas and Padberg (1976), Borndörfer and Weismantel (2000), Hoffman and Padberg (1993), and Padberg (1979b).

Other Nonconvex Problems

The versatility of the integer optimization model (1)–(6) might best be exemplified by the fact that many nonlinear/nonconvex optimization problems can be reformulated as MILPs. For example, one reformulation technique for representing nonlinear functions is to find a piecewise linear approximation and to represent the function by adding a binary variable corresponding to each piece of the approximation. The simplest example of such a transformation is the fixed-charge problem in which the cost function has both a fixed charge for initiating a given activity, as well as marginal costs associated with continued operation. One example of a fixed-charge problem is the facility location problem in which one wishes to locate facilities in such a way that the combined cost of building the facility (a onetime fixed cost) and producing and shipping to customers (marginal costs based on the amount shipped and produced) is minimized (see Drezner and Hamacher (2004)). The fact that nothing can be produced in the facility unless the facility exists creates a discontinuity in the cost function. This function can be transformed to a linear function by the introduction of additional variables that take on only the values 0 or 1. Similar transformations allow one to model separable nonlinear functions as integer (linear) optimization problems.

Solution Methods

Solving integer optimization problems (finding an optimal solution), can be a difficult task. The difficulty arises from the fact that unlike (continuous) linear optimization problems, for which the feasible region is convex, the feasible regions of integer optimization problems consists of either a discrete set of points or, in the case of general MILP, a set of disjoint polyhedra. In solving a linear optimization problem, one can exploit the fact that, due to the convexity of the feasible region, any locally optimal solution is a global optimum. In finding global optima for integer optimization problems, on the other hand, one is required to prove that a particular solution dominates all others by arguments other than the calculus-based approaches of convex optimization. The situation is further complicated by the fact that the description of the feasible region is implicit. In other words, the formulation (1)–(6) does not provide a computationally useful geometric description of the set $\mathcal{S}$. A more useful description can be obtained in one of two ways described next.

The first approach is to apply the powerful machinery of polyhedral theory. Weyl (1935) established the fact that a polyhedron can either be defined as the intersection of finitely many half-spaces, i.e., as a set of points satisfying inequalities of the form (2) and (3), or as the convex hull of a finite set of extreme points plus the conical hull of a finite set of extreme rays. If the data describing the original problem formulation are rational

numbers, then Weyl's theorem implies the existence of a finite system of linear inequalities describing the convex hull of $\mathcal{S}$, denoted by conv($\mathcal{S}$) (Nemhauser and Wolsey 1988). Optimization of a linear function over conv($\mathcal{S}$) is precisely equivalent to optimization over $\mathcal{S}$, but optimizing over conv($\mathcal{S}$) is a convex optimization problem. Thus, if it were possible to enumerate the set of inequalities in Weyl's description, one could solve the integer optimization problem using methods for convex optimization, in principle. The difficulty with this method, however, is that the number of linear inequalities required is too large to construct explicitly, so this does not lead directly to a practical method of solution.

A second approach is to describe the feasible set in terms of logical disjunction. For example, if $j \in B$, then either $x_j = 0$ or $x_j = 1$. This means that, in principle, the set $\mathcal{S}$ can be described by replacing constraints (4)–(5) with a set of appropriately chosen disjunctions. In fact, it is known that any MILP can be described as a set of linear inequalities of the form (2) and (3), plus a finite set of logical disjunctions (Balas 1998). Similarly, however, the number of such disjunctions would be too large to enumerate explicitly and so this does not lead directly to a practical method of solution either.

Although neither of the above methods for obtaining a more useful description of $\mathcal{S}$ leads directly to an efficient methodology because they both produce descriptions of impractical size, most solution techniques are nonetheless based on generating partial descriptions of $\mathcal{S}$ in one of the above forms (or a combination of both). The general outline of such a method is as follows:

1. Identify a (tractable) convex relaxation of the problem and solve it to either
   - Obtain a valid upper bound on the optimal solution value; or
   - Prove that the relaxation is infeasible or unbounded (and thus, the original MILP is also infeasible or unbounded)
2. If solving the relaxation produces a solution $\hat{x} \in \mathbb{R}^N$ that is feasible to the MILP, then this solution must also be optimal to the MILP.
3. Otherwise, either
   - Identify a logical disjunction satisfied by all members of $\mathcal{S}$, but not by $\hat{x}$ and add it to the description of $\mathcal{P}$ (more on how this is done below); or
   - Identify an implied linear constraint (called a valid inequality or a cutting plane) satisfied by all members of $\mathcal{S}$, but not by $\hat{x}$ and add it to the description of $\mathcal{P}$

In Step 1, the LP relaxation obtained by dropping the integrality conditions on the variables and optimizing over $\mathcal{P}$ is commonly used. Other possible relaxations include Lagrangian relaxations (Fisher 1981; Geoffrion 1974), semi-definite programming relaxations (Rendl 2010), and combinatorial relaxations, e.g., the one-tree relaxation for the traveling salesman problem Held and Karp (1970). This discussion initially considers use of the LP relaxation, since this is the simplest one and the one used in state-of-the-art software. Additional relaxations are considered in more detail in section "Advanced Procedures."

By recursively applying the basic strategy outlined above, a wide variety of convergent methods that generate partial descriptions of $\mathcal{S}$ can be obtained. These methods can be broadly classified as either implicit enumeration methods (employing the use of logical disjunction in Step 3) or cutting plane methods (based on the generation of valid inequalities in Step 3), though these are frequently combined into hybrid solution procedures in computational practice. In the next two sections, more details about these two classes of methods are given.

Enumerative Algorithms

The simplest approach to solving a pure integer optimization problem is to enumerate all finitely many possibilities (as long as the problem is bounded). However, due to the combinatorial explosion resulting from the fact that the size of the set $\mathcal{S}$ is generally exponential in the number of variables, only the smallest instances can be solved by such an approach. A more efficient approach is to only implicitly enumerate the possibilities by eliminating large classes of solutions using domination or feasibility arguments. Besides straightforward or implicit enumeration, the most commonly used enumerative approach is called branch and bound.

The branch-and-bound method was first proposed by Land and Doig (1960) and consists of generating disjunctions satisfied by points in $\mathcal{S}$ and using them to partition the feasible region into smaller subsets. Some variant of the technique is used by practically all state-of-the-art solvers. An LP-based

branch-and-bound method consists of solving the LP relaxation as in Step 1 above to either obtain a solution and an associated upper bound or to prove infeasibility or unboundedness. If the generated solution $\hat{x} \in \mathbb{R}^N$ to the relaxation is infeasible to the original MILP, then $\hat{x}_j \notin \mathbb{Z}$ for some $j \in B \cup I$. However, $x_j \in \mathbb{Z}$ for all $x \in \mathcal{S}$. Therefore, the logical disjunction

$$x_j \leq \lfloor \hat{x}_j \rfloor \text{ OR } x_j \geq \lceil \hat{x}_j \rceil \tag{7}$$

is satisfied by all $x \in \mathcal{S}$, but not by $\hat{x}$. In this case, one can impose the disjunction implicitly by branching, i.e., creating two subproblems, one associated with each of the terms of the disjunction (7).

The branch-and-bound method consists of applying this same method to each of the resulting subproblems recursively. Note that the optimal solution to a subproblem may or may not be the global optimal solution. Each time a new solution is found, it is checked to determine whether it is the best seen so far and if so, it is recorded and becomes the current incumbent. The true power of this method comes from the fact that if the upper bound obtained by solving the LP relaxation is smaller than the value of the current incumbent, the node can be discarded. Mitten (1970) provided the first description of a general algorithmic framework for branch and bound. Hoffman and Padberg (1985) provided an overview of LP-based branch-and-bound techniques. Linderoth and Savelsbergh (1999) reported on a computational study of search strategies used within branch and bound.

Cutting Plane Algorithms

Gomory (1958, 1960) was the first to derive a cutting plane algorithm following the basic outline above for integer optimization problems. His algorithm can be viewed, in some sense, as a constructive proof of Weyl's theorem. Although Gomory's algorithm converges to an optimal solution in a finite number of steps (in the case of pure integer optimization problems), the convergence to an optimum may be extraordinarily slow due to the fact that these algebraically derived valid inequalities are weak—they may not even support conv($\mathcal{S}$) and are hence dominated by stronger (but undiscovered) valid inequalities. Since the smallest possible description of conv($\mathcal{S}$) is desired, one would like to generate only the strongest valid inequalities, i.e., those that are part of some minimal description of conv($\mathcal{S}$). Such inequalities are called facets. In general, knowing all facets of conv($\mathcal{S}$) is enough to solve the MILP (though this set would still be very large in most cases).

A general cutting plane approach relaxes the integrality restrictions on the variables and solves the resulting LP relaxation over the set $\mathcal{P}$. If the LP is unbounded or infeasible, so is the MILP. If the solution to the LP is integer, i.e., satisfies constraints (4) and (5), then one has solved the MILP. If not, then one solves a separation problem whose objective is to find a valid inequality that cuts off the fractional solution to the LP relaxation while assuring that all feasible integer points satisfy the inequality—i.e., an inequality that separates the fractional point from the polyhedron conv($\mathcal{S}$). Such an inequality is called a cut for short. The algorithm continues until termination in one of two ways: either an integer solution is found (the problem has been solved successfully) or the LP relaxation is infeasible and therefore the integer problem is infeasible.

For ILPs, there are versions of Gomory's method that yield cutting plane algorithms that will produce a solution in a finite number of iterations, at least with the use of exact rational arithmetic. In practice, however, the algorithm could terminate in a third way—it may not be possible to identify a new cut even though the optimal solution has not been found either due to numerical difficulties arising from accumulated round-off error or because procedures used to generate the cuts are unable to guarantee the generation of a violated inequality, even when one exists. If one terminates the cutting plane procedure because of this third possibility, then, in general, the process has still improved the original formulation and the bound resulting from solving the LP relaxation is closer to the optimal value. By then switching to an implicit enumeration strategy, one may still be able to solve the problem. This hybrid strategy, known as branch and cut, is discussed in the next section.

Advanced Procedures

Branch and Cut

The two basic methods described above can be hybridized into a single algorithm that combines the

power of the polyhedral and disjunctive approaches. This method is called branch and cut. A rather sizable literature has sprung up around these methods. Papers describing the basic framework include those of Hoffman and Padberg (1991) and Padberg and Rinaldi (1991). Surveys of the computational issues and components of a modern branch-and-cut solver include Atamtürk and Savelsbergh (2005), Linderoth and Ralphs (2005), and Martin (2001). The major components of the algorithm consist of automatic reformulation and preprocessing procedures (see next section), heuristics that provide good feasible integer solutions, procedures for generating valid inequalities, and procedures for branching. All of these are embedded into a disjunctive search framework, as in the branch-and-bound approach. These components are combined so as to guarantee optimality of the solution obtained at the end of the calculation. The algorithm may also be stopped early to produce a feasible solution along with a bound on the relative distance of the current solution from optimality. This hybrid approach has evolved to be an extremely effective way of solving general MILPs. It is the basic approach taken by all state-of-the-art solvers for MILP.

Ideally, the cutting planes generated during the course of the algorithm would be facets of conv($\mathcal{S}$). In the early years of integer optimization, considerable research activity was focused on identifying part (or all) of the list of facets for specific combinatorial optimization problems by exploiting the special structure of conv($\mathcal{S}$) (Balas and Padberg 1972; Balas 1975; Bauer et al. 2002; Hammer et al. 1975; Nemhauser and Sigismondi 1992; Nemhauser and Trotter 1974; Nemhauser and Vance 1994; Padberg 1973, 1974, 1979a; Pochet and Wolsey 1991; Wolsey 1975, 1976). This led to a wide variety of problem-dependent algorithms that are nevertheless based on the underlying principle embodied in Weyl's theorem. An extensive survey of the use of these techniques in combinatorial optimization is given by Aardal and van Hoesel (1996a, b).

Research on integer optimization is increasingly focused on methods for generating inequalities based purely on the disjunctive structure of the problem and not on properties of a particular class of problems. Part of the reason for this is the need to be able to solve more general MILPs for which even the dimension of conv($\mathcal{S}$) is not known. With this approach, it is not possible to guarantee the generation of facets in every iteration, but theoretical advances have resulted in vast improvements in the ability to solve general unstructured integer optimization problems using off-the-shelf software. A survey of cutting plane methods for general MILPs is provided by (Cornuéjols 2008). Other papers on techniques for generating valid inequalities for general MILPs include Balas et al. (1993, 1996, 1999), Gu et al. (1998, 1999, 2000), Nemhauser and Wolsey (1990), Marchand and Wolsey (2001), and Wolsey (1990).

Equally as important as cutting plane generation techniques are branching schemes, though these methods have received far less attention in the literature. Branching methods are generally based on some method of estimating the impact of a given branching disjunction and trying to choose the best one according to certain criteria. Papers discussing branching methods include Achterberg et al. (2005), Fischetti and Lodi (2002), Karamanov and Cornuéjols (2009), and Owen and Mehrotra (2001).

There has been a surge in research on the use of heuristic methods within the branch-cut-cut framework in order to generate good solutions and improve bounds as the search progresses. Many search methods are based on limited versions of the same search procedures used to find globally optimal solutions. The development of such methods has led to marked improvements in the performance of exact algorithms (Balas and Martin 1980; Balas et al. 2004; Fischetti and Lodi 2002; Nediak and Eckstein 2001). In current state-of-the-art software, multiple heuristics are used because they are likely to produce feasible solutions more quickly than tree search, which helps both to eliminate unproductive subtrees and to calculate improved variable bounds that result in a tighter description of the problem. These heuristics include techniques for searching within the local neighborhood of a given linear feasible solutions for integer solutions using various forms of local search. Achtenberg and Berthold (2007), Danna et al. (2005), Fischetti et al. (2009), and Rothberg (2007) provide descriptions of heuristics built into current packages.

Automatic Reformulation

Before solving an integer optimization problem, the first step is that of formulation, in which a conceptual model is translated into the form (1)–(6). There are often different ways of mathematically representing the same problem, both because different systems of

the form (1)–(6) may define precisely the same set $\mathcal{S}$ and because it may be possible to represent the same conceptual problem using different sets of variables. There are a number of different ways in which the conceptual model can be translated into a mathematical model, but the most common is to use an algebraic modeling language, such as AIMMS, AMPL (Fourer et al. 1993), GAMS (Brooke et al. 1988), MPL, or OPL Studio.

The time required to obtain an optimal solution to a large integer optimization problem usually depends strongly on the way it is formulated, so much research has been directed toward the effective automatic reformulation techniques. Unlike linear optimization problems, the number of variables and constraints representing an integer optimization problem may not be indicative of its difficulty. In this regard, it is sometimes advantageous to use a model with a larger number of integer variables, a larger number of constraints, or both. Discussions of alternative formulation approaches are given in Guignard and Spielberg (1981) and Williams (1985), and a description of approaches to automatic reformulation or preprocessing is given in Anderson and Anderson (1995), Atamturk and Savelsbergh (2000), Brearley et al. (1975), Hoffman and Padberg (1991), Roy and Wolsey (1987), and Savelsbergh (1994).

A variety of difficult problems have been solved by reformulating them as either set-covering or set-partitioning problems having an extraordinary number of variables. Because for even small instances, such reformulations may be too large to solve directly, a technique known as column generation, which began with the seminal work of Gilmore and Gomory (1961) on the cutting stock problem, is employed. An overview of such transformation methods can be found in Barnhart et al. (1998). For specific implementations, for the vehicle routing problem, see Chabrier (2006), for the bandwidth packing problem, see Hoffman and Villa (2007) and Parker and Ryan (1995), for the generalized assignment problem, see Savelsbergh (1997), and for alternative column-generation strategies for solving the cutting stock problem see Vance et al. (1994). Bramel and Simchi-Levi (1997) have shown that the set-partitioning formulation for the vehicle routing problem with time windows is very effective in practice—that is, the relative gap between the fractional linear optimization solutions and the global integer solution is small. Similar results have been obtained for the bin-packing problem (Chan et al. 1998a) and for the machine-scheduling problem (Chan et al. 1998b).

Decomposition Methods

Relaxing the integrality restriction is not the only approach to relaxing the problem. An alternative approach to the solution to integer optimization problems is to relax a set of complicating constraints in order to obtain a more tractable model. This technique is effective when the problem to be solved is obtained by taking a well-solved base problem and adding constraints specific to a particular application. By capitalizing on the ability to solve the base problem, one can obtain bounds that are improved over those obtained by solving the LP relaxation. These bounding methods can then be used to drive a branch-and-bound algorithm, as described earlier. Such bounding methods are called constraint decomposition methods or simply decomposition methods, since they involve decomposing the set of constraints. By removing the complicating constraints from the constraint set, the resulting subproblem is frequently considerably easier to solve. The latter is necessary for the approach to work because the subproblems must be solved repeatedly. The bound found by decomposition can be tighter than that found by linear optimization, but only at the expense of solving subproblems that are themselves integer optimization problems. Decomposition requires that one understand the structure of the problem being solved in order to then relax the constraints that are complicating.

The bound resulting from a particular decomposition can be computed using two different computational techniques—Dantzig-Wolfe decomposition (Dantzig and Wolfe 1960; Vanderbeck 2000) (column generation) and Lagrangian relaxation (Fisher 1981; Geoffrion 1974; Held and Karp 1970). In the former case, solutions to the base problem are generated dynamically and combined in an attempt to obtain a solution satisfying the complicating constraints. In the latter case, the complicating constraints are enforced implicitly by penalizing their violation in the objective function. Overviews of the theory and methodology behind decomposition methods and how they are used in integer programming can be found in Ralphs and Galati (2005) and Vanderbeck and Wolsey (2010).

A related approach is that of Lagrangian decomposition (Guignard and Kim 1987), which consists of isolating sets of constraints so as to obtain multiple, separate, easy-to-solve subproblems. The dimension of the problem is increased by creating copies of variables that link the subsets and adding constraints that require these copies to have the same value as the original in any feasible solution. When these constraints are relaxed in a Lagrangian fashion, the problem decomposes into blocks that can be treated separately.

Most decomposition-based strategies involve decomposition of constraints, but there are cases in which it may make sense to decompose the variables. These techniques work well in the case when fixing some subset of the variables (the complicating variables) to specific values reduces the problem to one that is easy to solve. Benders' decomposition algorithm projects the problem into the space of these complicating variables and treats the remaining variables implicitly by adding so-called Benders cuts violated by solutions that do not have a feasible completion and adding a term to the objective function representing the cost of completion for any given set of value of the complicating variables (Benders 1962). For a survey on Benders cuts, see Hooker (2002).

Concluding Remarks

There are a number of topics related to combinatorial and integer optimization that have not been covered here. One such topic is the complexity of integer optimization problems (Garey and Johnson 1979), an area of theoretical study that has increased understanding of the implicit difficulty of integer optimization dramatically. Another important topic is that of heuristic solution approaches—that is, techniques for obtaining good but not necessarily optimal solutions to integer optimization problems quickly. In general, heuristics do not provide any guarantee as to the quality of the solutions they produce, but are very important in practice for a variety of reasons. Primarily, they may provide the only usable solution to very difficult optimization problems for which the current exact algorithms fail to produce one. Research into heuristic algorithms has applied techniques from the physical sciences to the approximate solution of combinatorial problems. For surveys of research in simulated annealing (based on the physical properties of heat), genetic algorithms (based on properties of natural mutation), and neural networks (models of brain function) see Hansen (1986), Goldberg (1989), and Zhang (2010), respectively. Glover and Laguna (1998) have generalized some of the attributes of these methods into a method called tabu search. Worst-case and probabilistic analysis of heuristics are discussed in Cornuejols et al. (1980), Karp (1976), and Kan (1986).

Another developing trend is the use of approaches from other disciplines in which optimization problems also arise. In some cases, multiple approaches can be used to handle difficult optimization problems by merging alternative strategies into a single algorithm (the so-called algorithm portfolio approach). As an example, constraint-logic programming was developed by computer scientists in order to work on problems of finding feasible solutions to a set of constraints. During the last decade, many of the advances of constraint-logic programming have been embedded into mathematical programming algorithms in order to handle some of the difficult challenges of combinatorial optimization such as those related to scheduling where there is often significant symmetry. For example, see Hooker (2007) and Rasmussen and Trick (2007) for some applications that use both Benders decomposition and constraint programming to handle difficult scheduling problems. For research that relates issues in computational logic to those associated with combinatorial optimization see McAloon and Tretkoff (1996).

See

- ▶ Air Traffic Management
- ▶ Airline Industry Operations Research
- ▶ Benders Decomposition Method
- ▶ Bin-Packing
- ▶ Branch and Bound
- ▶ Capital Budgeting
- ▶ Chinese Postman Problem
- ▶ Combinatorial Auctions
- ▶ Computational Complexity
- ▶ Facility Location
- ▶ Heuristics
- ▶ Linear Programming

- ▶ Network
- ▶ Set-Covering Problem
- ▶ Set-Partitioning Problem
- ▶ Tabu Search
- ▶ Traveling Salesman Problem

References

Aardal, K., & van Hoesel, C. (1996a). Polyhedral techniques in combinatorial optimization I: Applications and computations. *Statistica Neerlandica, 50*, 3–26.

Aardal, K., & van Hoesel, C. (1996b). Polyhedral techniques in combinatorial optimization II: Applications and computations. *Statistica Neerlandica, 50*, 3–26.

Abara, J. (1989). Applying integer linear programming to the fleet assignment problem. *Interfaces, 19*, 20–28.

Achtenberg, T., & Berthold, T. (2007). Improving the feasibility pump. *Discrete Mathematics, 4*, 77–86.

Achterberg, T., Koch, T., & Martin, A. (2005). Branching rules revisited. *Operations Research Letters, 33*, 42–54.

Ahuja, R., Magnanti, T., & Orlin, J. (1993). *Network flows: Theory, algorithms, and applications*. Englewood Cliffs, NJ: Prentice Hall.

Anderson, E., & Anderson, K. (1995). Presolving in linear programming. *Mathematical Programming, 71*, 221–245.

Applegate, D., & Cook, W. (1991). A computational study of the job-shop scheduling problem. *INFORMS Journal on Computing, 3*, 149–156.

Applegate, D., Bixby, R., Chvátal, V., & Cook, W. (2006). *The traveling salesman problem: A computational study*. Princeton, NJ: Princeton University Press.

Atamturk, A., & Savelsbergh, M. (2000). Conflict graphs in solving integer programming problems. *European Journal of Operational Research, 121*, 40–55.

Atamtürk, A., & Savelsbergh, M. (2005). Integer-programming software systems. *Annals of Operations Research, 140*, 67–124.

Avner, P. (2001). A radiation hybrid transcript may of the mouse genome. *Nature Genetics, 29*, 194–200.

Balas, E. (1975). Facets of the knapsack polytope. *Mathematical Programming, 8*, 146–164.

Balas, E. (1998). Disjunctive programming: Properties of the convex hull of feasible points. *Discrete Applied Mathematics, 89*, 3–44.

Balas, E., & Martin, R. (1980). Pivot and complement: A heuristic for 0-1 programming. *Management Science, 26*, 86–96.

Balas, E., & Padberg, M. (1972). On the set-covering problem. *Operations Research, 20*, 1152–1161.

Balas, E., & Padberg, M. (1976). Set partitioning: A survey. *SIAM Review, 18*, 710–760.

Balas, E., Ceria, S., & Corneujols, G. (1993). A lift-and-project cutting plane algorithm for mixed 0-1 programs. *Mathematical Programming, 58*, 295–324.

Balas, E., Ceria, S., & Cornuejols, G. (1996). Mixed 0-1 programming by lift-and-project in a branch-and-cut framework. *Management Science, 42*, 1229–1246.

Balas, E., Ceria, S., Cornuejols, G., & Natraj, N. (1999). Gomory cuts revisited. *Operations Research Letters, 19*, 1–9.

Balas, E., Schmieta, S., & Wallace, C. (2004). Pivot and shift—A mixed integer programming heuristic. *Discrete Optimization, 1*, 3–12.

Barahona, F., Grötschel, M., Jünger, M., & Reinelt, G. (1988). An application of combinatorial optimization to statistical physics and circuit layout design. *Operations Research, 36*, 493–513.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P., & Vance, P. H. (1998). Branch and price: Column generation for solving huge integer programs. *Operations Research, 46*, 316–329.

Bauer, P., Linderoth, J., & Savelsbergh, M. (2002). A branch and cut approach to the cardinality constrained circuit problem. *Mathematical Programming, 9*, 307–348.

Benders, J. F. (1962). Partitioning procedures for solving mixed variable programming problems. *Numerische Mathematik, 4*, 238–252.

Bertsimas, D., & Weismantel, R. (2005). *Optimization over integers*. Cambridge, MA: Dynamic Ideas.

Borndörfer, R., & Weismantel, R. (2000). Set packing relaxations of some integer programs. *Mathematical Programming, 88*, 425–450.

Bramel, J., & Simchi-Levi, D. (1997). On the effectiveness of set covering formulations for the vehicle routing problem with time windows. *Operations Research, 45*, 295–301.

Brearley, A., Mitra, G., & Williams, H. (1975). Analysis of mathematical programming problems prior to applying the simplex method. *Mathematical Programming, 8*, 54–83.

Brooke, A., Kendrick, D., & Meeraus, A. (1988). *GAMS, a user's guide*. Redwood City, CA: The Scientific Press.

Chabrier, A. (2006). Vehicle routing problem with elementary shortest path based column generation. *Computers and Operations Research, 33*(10), 2972–2990.

Chan, L., Muriel, A., & Simchi-Levi, D. (1998a). Parallel machine scheduling, linear programming, and parameter list scheduling heuristics. *Operations Research, 46*, 729–741.

Chan, L., Simchi-Levi, D., & Bramel, J. (1998b). Worst-case analyses, linear programming and the bin-packing problem. *Mathematical Programming, 83*, 213–227.

Chvátal, V. (1983). *Linear programming*. New York: W. H. Freeman.

Cornuéjols, G. (2008). Valid inequalities for mixed integer linear programs. *Mathematical Programming B, 112*, 3–44.

Cornuejols, G., Nemhauser, G., & Wolsey, L. (1980). Worst-case and probabilistic analysis of algorithms for a location problem. *Operations Research, 28*, 847–858.

Cramton, P., Shoham, Y., & Steinberg, R. (2006). *Combinatorial auctions*. Cambridge, MA: MIT Press.

Danna, E., Rothberg, E., & LePape, C. (2005). Exploring relaxation induced neighborhoods to improve MIP solutions. *Mathematical Programming, 102*, 71–90.

Dantzig, G., & Wolfe, P. (1960). Decomposition principle for linear programs. *Operations Research, 8*, 101–111.

Drezner, Z., & Hamacher, H. (2004). *Facility location: Applications and theory*. Berlin: Springer.

Edmonds, J., & Johnson, E. L. (1973). Matching, Euler tours, and the Chinese postman. *Mathematical Programming, 5*, 88–124.

Fischetti, M., & Lodi, A. (2002). Local branching. *Mathematical Programming, 98*, 23–47.

Fischetti, M., Lodi, A., & Salvagnin, D. (2009). Just MIP It! In V. Maniezzo, T. Stuetzle, & S. Voss (Eds.), *MATHEURISTICS: Hybridizing metaheuristics and mathematical programming* (pp. 39–70). Berlin: Springer.

Fisher, M. L. (1981). The lagrangian method for solving integer programming problems. *Management Science, 27*, 1–18.

Fourer, R., Gay, D. M., & Kernighan, B. W. (1993). *AMPL: A modeling language for mathematical programming*. San Francisco: The Scientific Press.

Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. New York: W. H. Freeman.

Geoffrion, A. (1974). Lagrangian relaxation for integer programming. *Mathematical Programming Study, 2*, 82–114.

Gilmore, P. C., & Gomory, R. E. (1961). A linear programming approach to the cutting stock problem. *Operations Research, 9*, 849–859.

Glover, F., & Laguna, M. (1998). *Tabu search*. Berlin: Springer.

Goldberg, D. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley.

Golden, B., Raghavan, S., & Wasail, E. (2010). *The vehicle routing problem: Latest advances and new challenges*. Berlin: Springer.

Gomory, R. E. (1958). Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Monthly, 64*, 275–278.

Gomory, R. E. (1960). *An algorithm for the mixed integer problem* (Tech. Rep. RM-2597). The RAND Corporation. Santa Monica, California.

Gu, Z., Nemhauser, G. L., & Savelsbergh, M. W. P. (1998). Cover inequalities for 0-1 linear programs: Computation. *INFORMS Journal on Computing, 10*, 427–437.

Gu, Z., Nemhauser, G. L., & Savelsbergh, M. W. P. (1999). Lifted flow covers for mixed 0-1 integer programs. *Mathematical Programming, 85*, 439–467.

Gu, Z., Nemhauser, G. L., & Savelsbergh, M. W. P. (2000). Sequence independent lifting. *Journal of Combinatorial Optimization, 4*, 109–129.

Guignard, M., & Kim, S. (1987). Lagrangian decomposition: A model yielding stronger lagrangian bounds. *Mathematical Programming, 39*, 215–228.

Guignard, M., & Spielberg, K. (1981). Logical reduction methods in zero-one programming: Minimal preferred inequalities. *Operations Research, 29*, 49–74.

Hammer, P. L., Johnson, E. L., & Peled, U. N. (1975). Facets of regular 0-1 polytopes. *Mathematical Programming, 8*, 179–206.

Hane, C., Barnhart, C., Johnson, E., Marsten, R., Nemhauser, G., & Sigismondi, G. (1995). The fleet assignment problem: Solving a large-scale integer program. *Mathematical Programming, 70*, 211–232.

Hansen, P. (1986). The steepest ascent mildest descent heuristic for combinatorial programming. *Proceedings of Congress on Numerical Methods in Combinatorial Optimization*, Italy.

Held, M., & Karp, R. M. (1970). The traveling salesman problem and minimum spanning trees. *Operations Research, 18*, 1138–1162.

Hoffman, K., & Padberg, M. (1985). LP-based combinatorial problem solving. *Annals of Operations Research, 4*, 145–194.

Hoffman, K. L., & Padberg, M. W. (1991). Improving LP-representations of zero-one linear programs for branch and cut. *ORSA Journal on Computing, 3*, 121–134.

Hoffman, K., & Padberg, M. (1993). Solving airline crew scheduling problems by branch-and-cut. *Management Science, 39*, 667–682.

Hoffman, K., & Villa, C. (2007). A column-generation and branch-and-cut approach to the bandwidth-packing problem. *Journal of Research of the National Institute of Standards and Technology, 111*, 161–185.

Hooker, J. (2002). Logic, optimization, and constraint programming. *INFORMS Journal on Computing, 14*, 295–321.

Hooker, J. (2007). Planning and scheduling by logic-based benders decomposition. *Operations Research, 55*, 588–602.

Jünger, M., Liebling, T., Naddef, D., Nemhauser, G., Pulleyblank, W., Reinelt, G. (2010). *Fifty years of integer programming: 1958–2008*. Berlin: Springer.

Kan, A. R. (1986). An introduction to the analysis of approximation algorithms. *Discrete Applied Mathematics, 14*, 111–134.

Karamanov, M., & Cornuéjols, G. (2009). Branching on general disjunctions. *Mathematical Programming*, 128, 403–406.

Karp, R. (1976). Probabilistic analysis of partitioning algorithms for the traveling salesman problem. In J. F. Traub (Ed.), *Algorithms and complexity: New directions and recent results* (pp. 1–19). New York: Academic.

Land, A. H., & Doig, A. G. (1960). An automatic method for solving discrete programming problems. *Econometrica, 28*, 497–520.

Linderoth, J., & Ralphs, T. (2005). Noncommercial software for mixed-integer linear programming. In J. Karlof (Ed.), *Integer programming: Theory and practice* (pp. 253–303). Boca Raton, FL: CRC Press.

Linderoth, J. T., & Savelsbergh, M. W. P. (1999). A computational study of search strategies in mixed integer programming. *INFORMS Journal on Computing, 11*, 173–187.

Marchand, H., & Wolsey, L. (2001). Aggregation and mixed integer rounding to solve MIPs. *Operations Research, 49*, 363–371.

Markowitz, H., & Manne, A. (1957). On the solution of discrete programming problems. *Econometrica, 2*, 84–110.

Martello, S., & Toth, P. (1990). *Knapsack problems*. New York: Wiley.

Martin, A. (2001). Computational issues for branch-and-cut algorithms. In M. Juenger & D. Naddef (Eds.), *Computational combinatorial optimization* (pp. 1–25). Berlin: Springer.

McAloon, K., & Tretkoff, C. (1996). *Optimization and computational logic*. New York: Wiley.

Mitten, L. (1970). Branch-and-bound methods: General formulation and properties. *Operations Research, 18*, 24–34.

Nediak, M., & Eckstein, J. (2001). *Pivot, cut, and dive: A heuristic for mixed 0-1 integer programming* (Tech. Rep. RUTCOR Research Report RRR 53-2001). Rutgers University, Newark, New Jersey.

Nemhauser, G. L., & Sigismondi, G. (1992). A strong cutting plane/branch-and-bound algorithm for node packing. *Journal of the Operational Research Society, 43*, 443–457.

Nemhauser, G. L., & Trotter, L. E., Jr. (1974). Properties of vertex packing and independence system polyhedra. *Mathematical Programming, 6*, 48–61.

Nemhauser, G., & Vance, P. (1994). Lifted cover facets of the 0-1 knapsack polytope with GUB constraints. *Operations Research Letters, 16*, 255–264.
Nemhauser, G., & Wolsey, L. A. (1988). *Integer and combinatorial optimization*. New York: Wiley.
Nemhauser, G., & Wolsey, L. (1990). A recursive procedure for generating all cuts for 0-1 mixed integer programs. *Mathematical Programming, 46*, 379–390.
Odlyzko, A. M. (1990). The rise and fall of knapsack cryptosystems. In C. Pomerance (Ed.), *Cryptology and computational number theory* (pp. 75–88). Ann Arbor: American Mathematical Society.
Owen, J., & Mehrotra, S. (2001). Experimental results on using general disjunctions in branch-and-bound for general-integer linear programs. *Computational Optimization and Applications, 20*(2).
Padberg, M. (1973). On the facial structure of set packing polyhedra. *Mathematical Programming, 5*, 199–215.
Padberg, M. (1974). Perfect zero-one matrices. *Mathematical Programming, 6*, 180–196.
Padberg, M. (1979a). Covering, packing and knapsack problems. *Annals of Discrete Mathematics, 4*, 265–287.
Padberg, M. W. (1979b). A note on 0-1 programming. *Operations Research, 23*, 833–837.
Padberg, M. W., & Rinaldi, G. (1991). A branch and cut algorithm for the solution of large scale traveling salesman problems. *SIAM Review, 33*, 60–100.
Papadimitriou, C., & Steiglitz, K. (1982). *Combinatorial optimization: Algorithms and complexity*. New Jersey: Prentice-Hall.
Parker, R., & Rardin, R. (1988). *Discrete optimization*. San Diego: Academic.
Parker, M., & Ryan, J. (1995). A column generation algorithm for bandwidth packing. *Telecommunication Systems, 2*, 185–196.
Pinedo, M. (2008). *Scheduling: Theory, algorithms, and systems*. Berlin: Springer.
Pochet, Y., & Wolsey, L. (1991). Solving multi-item lot sizing problems using strong cutting planes. *Management Science, 37*, 53–67.
Ralphs, T., & Galati, M. (2005). Decomposition in integer programming. In J. Karlof (Ed.), *Integer programming: Theory and practice* (pp. 57–110). Boca Raton, FL: CRC Press.
Rasmussen, R., & Trick, M. (2007). A benders approach to the constrained minimum break problem. *European Journal of Operational Research, 177*, 198–213.
Ravikumar, C. (1996). *Parallel methods for VLSI layout design*. Norwood, NJ: Ablex Publishing Corporation.
Rendl, F. (2010). Semidefinite relaxations for integer programming. In M. Jünger, T. Liebling, D. Naddef, G. Nemhauser, W. Pulleyblank, G. Reinelt, G. Rinaldi, & L. Wolsey (Eds.), *Fifty years of integer programming: 1958–2008* (pp. 687–726). Berlin: Springer.
Rothberg, E. (2007). An evolutionary algorithm for polishing mixed integer programming solutions. *INFORMS Journal on Computing, 19*, 534–541.
Roy, T. J. V., & Wolsey, L. A. (1987). Solving mixed integer 0-1 programs by automatic reformulation. *Operations Research, 35*, 45–57.
Savelsbergh, M. W. P. (1994). Preprocessing and probing techniques for mixed integer programming problems. *ORSA Journal on Computing, 6*, 445–454.
Savelsbergh, M. W. P. (1997). A branch and price algorithm for the generalized assignment problem. *Operations Research, 45*, 831–841.
Schrijver, A. (1986). *Theory of linear and integer programming*. Chichester: Wiley.
Schrijver, A. (2003). *Combinatorial optimization: Polyhedra and efficiency*. Berlin: Springer.
Shah, R. (1998). *Optimization problems in SONET/WDM ring architecture*. Master's Essay, Rutgers University, Newark, NJ.
Vance, P. H., Barnhart, C., Johnson, E. L., & Nemhauser, G. L. (1994). Solving binary cutting stock problems by column generation and branch and bound. *Computational Optimization and Applications, 3*, 111–130.
Vance, P., Barnhart, C., Johnson, E., & Nemhauser, G. (1997). Airline crew scheduling: A new formulation and decomposition algorithm. *Operations Research, 45*, 188–200.
Vanderbeck, F. (2000). On Dantzig-Wolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm. *Operations Research, 48*, 111–128.
Vanderbeck, F., & Wolsey, L. (2010). Reformulation and decomposition of integer programs. In M. Jünger, T. Liebling, D. Naddef, G. Nemhauser, W. Pulleyblank, G. Reinelt, G. Rinaldi, & L. Wolsey (Eds.), *Fifty years of integer programming: 1958–2008* (pp. 431–504). Berlin: Springer.
Weingartner, H. (1963). *Mathematical programming and the analysis of capital budgeting problems*. Englewood Cliffs, NJ: Prentice Hall.
Weyl, H. (1935). Elementare theorie der konvexen polyheder. *Commentarii Mathematici Helvetici, 7*, 290–306.
Williams, H. (1985). *Model building in mathematical programming* (2nd ed.). New York: Wiley.
Wolsey, L. A. (1975). Faces for a linear inequality in 0-1 variables. *Mathematical Programming, 8*, 165–178.
Wolsey, L. A. (1976). Facets and strong valid inequalities for integer programs. *Operations Research, 24*, 367–372.
Wolsey, L. A. (1990). Valid inequalities for mixed integer programs with generalized and variable upper bound constraints. *Discrete Applied Mathematics, 25*, 251–261.
Wolsey, L. A. (1998). *Integer programming*. New York: Wiley.
Zhang, X. (2010). *Neural networks in optimization*. Berlin: Springer.

Integer Goal Programming

A goal-programming methodology that generates an integer solution for decision variables.

See

► Goal Programming

Integer-Programming Problem

A mathematical-programming problem in which some or all of its variables are restricted to integer values.

See

► Integer and Combinatorial Optimization

Intelligent Manufacturing Systems

Automated processing systems for manufacturing operations that include intelligent machines, advanced sensors for real-time-in-process measurements, software for precision control of machine tools, and information technology for integrating all elements of a product's life cycle.

See

► Automation in Manufacturing and Services
► Flexible Manufacturing Systems
► Industrial Applications
► Operations Management
► Production Management

Intelligent Transportation Systems

Systems in which synergistic communication and information technologies are combined with operations research methodologies such as simulation and optimization to improve transportation systems of all modes, from road and rail to air and water, including their interfaces. Problems addressed span the spectrum from system design to real-time control and management of operations, e.g., traffic management and mobility management, and encompass infrastructure, vehicles and users.

Intensity Function

► Failure-Rate Function
► Point Stochastic Processes
► Renewal Process

Interactive Multiple Objective Mathematical Programming

Julia Pet-Armacost, Mansooreh Mollaghasemi and Robert L. Armacost
University of Central Florida, Orlando, FL, USA

Introduction

Decision making typically involves a decision maker selecting a course of action that optimizes some criterion while respecting the resources and other conditions that must be satisfied. When multiple criteria are involved, this class of problems is generally referred to as multiple criteria decision problems. In some circumstances, the number of alternatives is limited and the decision maker identifies a number of (multiple) desirable measureable attributes. Each of the alternatives is assessed with respect to each of the attributes to provide information to the decision maker to aid in selecting the desired alternative. This type of problem is generally termed a multiple attribute decision problem. In other situations, the set of alternatives may be very large and represented as various types and levels of particular actions. The decision maker may be able to determine how various combinations of these alternatives contribute to a particular objective (e.g., completion time, cost, profit). When decision problems involve multiple objectives, the challenge is to select a set of alternatives (and values) that best satisfy (optimize) those objectives while respecting resources and other required conditions. This type of decision problem is generally referred to as a multiple objective decision problem. When the multiple objective problems are expressible in a mathematical structure, the problems are referred to as multiple

objective mathematical programming (optimization) problems. When a mathematical programming formulation exists, there is typically an analyst involved in the decision process to support the decision maker.

The literature on multiple objective decision making is extensive. White (1990) identified over 500 methods and applications published between 1955 and 1986. Detailed treatments of multiple objective optimization methods are available in Chankong and Haimes (1983) and Steuer (1986), the latter of which provides extensive treatment of linear multiple objective optimization methods. The text by Miettinen (1999) is a comprehensive treatment of nonlinear multiple objective optimization that includes the essential theory, detailed descriptions of multiple methods, and over 700 references that address all aspects of the problem. Branke, Deb, Miettinen, and Slowiński, (2008) provide basics on multiple objective optimization including noninteractive and interactive approaches, a description of recent interactive and preference-based approaches including evolutionary methods, and chapters on visualization, modeling, implementation and applications. Finally, Zopounidis and Pardalos (2010) provide a compilation of methods that includes chapters on interactive and evolutionary approaches for multiple objective optimization problems.

In general, a multiple objective mathematical programming problem can be expressed as follows:

$$\begin{aligned} &\text{Minimize } \{f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_p(\boldsymbol{x})\} \\ &\text{Subject to } g_j(\boldsymbol{x}) \leq 0 \; j = 1, 2, \ldots, m \end{aligned} \tag{1}$$

where $\boldsymbol{x}$ is an n-dimensional vector of decision variables, $f_i(\boldsymbol{x})$, $i = 1, 2, \ldots, p$, are p distinct objective functions, and $g_i(\boldsymbol{x})$, $j = 1, 2, \ldots, m$, are m distinct constraint functions.

Based on the classification given in Evans (1984), almost all methods for solving multiple objective problems involve two general sub-processes: (1) articulation of the decision maker's preference structure, and (2) optimization over the preference structure. The various methods for solving multiple objective problems have been categorized into three types according to the timing of these sub-processes: (1) prior articulation of preferences where the preference structure is obtained prior to the optimization process (a priori), (2) posterior articulation of preferences where the decision maker's preference is elicited after the generation of a candidate solution set (a posteriori), and (3) progressive articulation of preferences where the elicitation of information about the preference structure is interspersed with optimization processes (interactive). With a priori methods, the decision maker acts first and explicitly expresses his or her preferences and then the analyst generates the best solution consistent with the decision maker's preferences. With a posteriori methods, the analyst acts first to generate a number of alternative solutions and then the decision maker applies his or her preference structure (perhaps implicit) to select the best solution. With interactive methods, there is a back and forth exchange between the decision maker and the analyst that continues until the decision maker is satisfied that no significantly better solution exists or can easily be found.

Most multiple objective mathematical programming methods require the generation of nondominated solutions. Let $\mathbf{X}$ be the set of all feasible solutions to problem (1). An efficient (nondominated, Pareto optimal) solution is a feasible solution $\boldsymbol{x}* \in X$, for which there does not exist any other feasible solution, $\boldsymbol{x} \in \boldsymbol{X}$, that is the same or better in each of the objectives. In other words, you cannot find another solution $\boldsymbol{x}$, where $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}*)$ for $i = 1, 2, \ldots, p$, and for at least one $i, f_i(\boldsymbol{x}) < f_i(\boldsymbol{x}*)$. (Note that this definition assumes that smaller values of $f_i(\boldsymbol{x})$ are preferred.) Moving from one Pareto optimal solution to another requires some sort of trade-off reflecting the value equivalence in improving one objective relative to degrading another objective.

In some cases, a decision maker does not have any particular preferences with respect to the values of the objective functions. In these cases, a method that can identify a single Pareto optimal point may suffice. Miettinen (1999) identifies the global criterion method and the multiobjective proximal bundle method as two viable approaches. In the global criterion method, a reference point is selected and a chosen metric is used to minimize the distance to the reference point. In this case, all objective functions

are assumed to be equally important. When the L_p-metric ($1 \leq p < \infty$) is used, the solution is Pareto optimal. When $p = \infty$, the metric is called the Tchebycheff metric. Objective function scaling can influence the relative importance of particular objectives in the optimization. Another approach is the multiobjective proximal bundle (MPB) method that seeks to move in a direction where the values of all of the objective functions improve simultaneously. Unlike other approaches, MPB does not transform the problem into one with a single objective function. Rather, its scalarization takes place inside a special (nondifferentiable) optimizer. The method is described in detail in Mäkelä (1993).

When a decision maker does have preferences with respect to the values of the objective functions, it is necessary to be able to evaluate sets of nondominated solutions. The following two methods are common approaches for generating nondominated solutions. Clearly, the two methods can serve as a posteriori methods since a decision maker can use the set of solutions to select the one that best satisfies his or her preferences. However, many of the interactive solution methods incorporate either or both of these two basic approaches.

A Weighting (Scalarization) Approach for Generating Nondominated Solutions. A common approach to finding nondominated solutions to a multiple objective mathematical programming problem is to convert the set of multiple objectives into a single objective through the use of weights. The weighting technique transforms problem (1) into a single objective problem given in (2):

$$\begin{aligned} \text{Minimize} \quad & f(\boldsymbol{x}) = w_1 f_1(\boldsymbol{x}) + w_2 f_2(\boldsymbol{x}) + \ldots + w_p f_p(\boldsymbol{x}) \\ \text{Subject to} \quad & g_j(\boldsymbol{x}) \leq 0 \; j = 1, 2, \ldots, m \\ & w_1 + w_2 + \ldots + w_p = 1 \\ & w_i > 0, i = 1, 2, \ldots, p \end{aligned} \tag{2}$$

Given a set of weights that are nonnegative and that sum to one, the solution to problem (2) is a nondominated solution. This means that the weighting approach is guaranteed to generate at least some of the possible nondominated solutions if the weights are varied. If the problem is also convex, then the weighting approach is guaranteed to generate all of the nondominated solutions if all possible weights are explored, see Chankong and Haimes (1983). Of course, there is an infinite number of possible weights that can be assigned and there could be an infinite number of nondominated solutions.

ε-Constraint-Based Approach for Generating Nondominated Solutions. Another approach to finding nondominated solutions to a multiple objective mathematical programming problem is to convert the set of multiple objectives into a single objective by treating all but one of the objectives as inequality constraints. In this approach, one primary objective, $f_k(\boldsymbol{x})$, is selected to be minimized while the remaining objectives are converted into inequality constraints. Consider the following multiple objective programming problem:

$$\begin{aligned} \text{Minimize} \quad & f_k(\boldsymbol{x}) \\ \text{Subject to} \quad & g_j(\boldsymbol{x}) \leq 0 \; j = 1, 2, \ldots, m \\ & f_j(\boldsymbol{x}) \leq \varepsilon_i \; j = 1, 2, \ldots, p, i \neq k \end{aligned} \tag{3}$$

If the values of ε_i are chosen so that problem (3) has feasible solutions, then the solution is guaranteed to be nondominated. This means that at least some of the nondominated solutions can be discovered by solving problem (3) for specific feasible values for ε_i. In fact, unlike the weighting approach, it turns out that all of the nondominated solutions can be generated if the ε_i are varied over all possible feasible values, even for problems that are not convex.

Interactive Methods

The techniques that rely on a progressive articulation of preferences (interactive methods) follow a common pattern. The decision maker is presented with a subset of the non-dominated alternatives and is asked to provide some local preference information on these alternatives. This information allows the formulation of a single criterion subproblem, which is then solved. The new nondominated solution and the outcome are then presented to the decision maker to provide new local preference information. This process is repeated until the decision maker either converges toward a best-compromise solution, or terminates the process prior to reaching this point. The objective of these approaches is to find a satisfactory solution after

a reasonable number of iterations and within a reasonable amount of time.

Interactive methods differ in how the single objective optimization problem is formed, how the preference information is incorporated and obtained from the decision maker, and how information is provided to the decision maker. Interactive methods may involve direct interaction with an analyst or with a computer program with appropriate interface for the decision maker. In interactive methods, the decision maker may be required to provide one of the following types of information regarding the nondominated solutions or criteria, (Evans, Stuckman, & Mollaghasemi, 1991):

1. A ranking of nondominated solutions in the outcome space,
2. A readjustment of aspiration levels from one iteration to the next, or
3. Marginal rates of substitution between various criteria.

Interactive methods may be based on the existence of an underlying value function for the decision maker (often implicit), the ability to create reference points on which the algorithms can operate, the ability to classify objectives, or a combination of these approaches. The value function allows a decision maker to identify trade-off values and generate marginal rates of substitution. The following methods are representative of the kinds of interactive approaches that have been developed for multiple objective optimization. The first four assume the existence of a value function and the remaining methods use reference points and classification. The following paragraphs provide very brief, high level descriptions of the various approaches. Mollaghasemi and Pet-Edwards (1997) provide additional descriptions. Miettinen (1999) includes a much more detailed treatment of the methods and provides much of the background for the following descriptions.

Interactive Surrogate Worth Trade-off (ISWT) Method. The basic idea of the interactive surrogate worth trade-off (ISWT) method (Chankong & Haimes, 1978) is to maximize an underlying value function that is known implicitly. The algorithm begins with the ε-constraint method by having the analyst select one objective to optimize and provide bounds on the other objectives. The problem is solved and a Pareto optimal solution is presented to the decision maker. The opinions of the decision maker regarding the trade-off rates at the current solution are used to determine a new search direction. Specifically, the decision maker conducts a worth assessment using a specified worth scale to determine how (much) the decision maker would like to make a trade-off between the primary response and each secondary response, where the value of the primary response decreases by the value of the Lagrange Multiplier for a one unit increase in value of the secondary response. The worth values are used to update the right-hand-side of the secondary response and then the problem is re-optimized. The process continues until the decision maker is satisfied with the solution.

Geoffrion-Dyer-Feinberg (GDF) Method. The Geoffrion-Dyer-Feinberg (GDF) method (Geoffrion, Dyer, & Feinberg, 1972) also assumes the existence of a decision maker value function and the process of the method is similar to the ISWT method, although the computational approach is different. The GDF method maximizes a value function and requires the decision maker to identify the reference function and then specify marginal rates of substitution between this function and the other objectives at the current solution point. The marginal rates of substitution are used to specify the direction of steepest ascent for the value function. The decision maker also helps to determine the step size. The optimization is conducted iteratively with the decision maker choosing the preferred solution among each set of solutions until the decision maker chooses to stop. The GDF method is one of the most well-known interactive methods.

Sequential Proxy Optimization Technique (SPOT) Method. The sequential proxy optimization technique (SPOT) method (Sakawa, 1982) is also based on maximizing the decision maker's underlying value function. SPOT includes some of the ideas of ISWT and GDF. The algorithm begins with the ε-constraint method like the ISWT method. With that solution, like the GDF method, the decision maker must specify the marginal rates of substitution. Then, unlike GDF where the decision maker is involved in determining a step size for the next iteration, a proxy function is generated by solving a series of ε-constraint problems and step size is determined from those results. The optimization is conducted iteratively until the decision maker is satisfied.

Interactive Goal Programming. Dyer (1972) introduced the concept of interactive goal

programming to provide a linkage between goal programming and interactive strategies that had been suggested for the optimization of multiple criteria optimization problems. The method assumes that the decision maker can specify a goal for each objective. It further assumes that the decision maker has an underlying utility function and can provide appropriate trade-off weights. These weights are used in a one-sided goal-programming problem to generate a new solution. The optimization is conducted iteratively until the decision maker chooses to stop.

Tchebycheff Method. The Tchebycheff method (Steuer & Choo, 1983) does not assume the existence of a decision maker value function and requires much less information from the decision maker. The algorithm starts by computing the utopian objective vector, an infeasible point that is slightly perturbed from the ideal objective vector that corresponds to the point where all objectives are at their minimum value. The algorithm proceeds by minimizing the maximum weighted distance of the function from the utopian objective vector using the weighted Tchebycheff metric. The bounds on the weights are tightened to reduce the number of Pareto optimal solutions generated. The decision maker chooses a most preferred objective vector among a subset of the generated ones at each iteration until a final solution is chosen.

STEM (Step Method). The step method (STEM) (Benayoun, de Montgolfier, Tergny, & Laritchev, 1971) is one of the first interactive methods developed for multiple objective optimization. While STEM has some elements similar to the Tchebycheff method, it focuses on identifying satisfactory solutions rather than optimizing an underlying value function. The method is described in more detail below, but follows the following framework. The decision maker must be able to indicate functions that have acceptable values and those that have values that are too high. The weighted Tchebycheff metric is used to generate solutions. Then the decision maker is asked to determine which of the objectives are satisfactory and relax the upper bounds on those objectives. The method repeats until the decision maker is satisfied for all objectives.

Reference Point Method. The reference point method (Wierzbicki, 1982) is based on the decision maker specifying a reference point of aspiration levels that are reasonable or desirable to the decision maker. The reference point method begins by providing the decision maker with some information that provides a range of the Pareto optimal set based on lower and upper bounds on the objectives. The decision maker specifies a reference point from which the achievement function is minimized. The solutions are provided to the decision maker and if one of the solutions is satisfactory, the process stops. If not, a new reference point is determined and the process continues. The reference point idea has been subsequently incorporated in other interactive methods. For example, the reference direction approach (Korhonen & Laasko, 1986) projects the vector from the current iteration point to the reference point. This provides the decision maker with more information to determine the next direction and also provides a wider part of the weakly Pareto optimal set to review. The reference direction method minimizes the computational effort by having the decision maker determine the number of steps to be taken in the reference direction, minimizing the number of alternatives that the decision maker will review. This method may be facilitated by computer graphic representations of the Pareto optimal curve. Korhonen (1987) developed a general software package to apply this visual interactive graphic technique to multiple criteria problems in general.

GUESS Method. The GUESS method (Buchanan, 1997) is a relatively simple method that assumes that lower bounds (ideal objective vector) and upper bounds (nadir objective vector) can be computed and are available. The decision maker specifies a reference point (a guess) as well as any additional upper or lower bounds to the objective functions. Then a function representing the maximum weighted deviation from the nadir objective vector is optimized with equal proportional achievement and the solution is presented to the decision maker. If the solution is satisfactory the process stops otherwise it is repeated with the decision maker specifying new bounds and reference points.

Satisficing Trade-Off Method (STOM). The satisficing trade-off method (STOM), based on satisficing decision making, incorporates ideas from STEM, the reference point method, and GUESS (Nakayama, 1989). STOM begins by optimizing a scalarizing function and providing the solution to the decision maker. Different kinds of scalarizing functions (and weighting schemes) can be used, but require that the utopian objective vector is known and

available. The decision maker reviews each objective and labels them as unacceptable, acceptable with the ability to relax, or acceptable as is. The decision maker provides aspiration levels for the objectives to be improved. Then the modified scalarizing function is minimized and the process is repeated.

Light Beam Search Method. The light beam search method has the decision maker identify bounds for each objective, as well as to specify indifference thresholds (Jaszkiewicz & Slowiński, 1994). This incorporates features and reference point methods with concepts from multiple attribute decision analysis. The decision maker is asked to specify indifference thresholds and preference thresholds that are used to establish preference relations between alternative pairs of objective vectors. The achievement function is minimized and the solution, as well as Pareto optimal neighbors, are presented to the decision maker. If one of the alternatives is satisfactory, then the process is stopped. Otherwise, the decision maker can revise reference points or thresholds and the function is re-optimized. This method allows the decision maker to save preferred solutions, explore other directions, and then select among the preferred solutions.

The preceding general descriptions of the interactive methods have not rigorously described the assumptions and characteristics for the objectives and constraint functions in the multiple objective problem. The methods described above were developed for cases where the functions were differentiable and many required other kinds of smooth properties. In real-world optimization problems, one may encounter nondifferentiable functions. For noninteractive approaches, use of some smoothing approaches may make a nondifferentiable problem solvable. For interactive approaches, however, there are limits since trade-off weights, for example, require twice continuous differentiability.

NIMBUS. To address these complicated real-world problems, Miettinen (1994) developed the Nondifferentiable Interactive Bundle-based optimization System (NIMBUS). The NIMBUS algorithm includes two versions: one uses a vector subproblem, and the other uses a scalar subproblem. They differ with respect to the handling of the information provided by the decision maker. The NIMBUS method requires that at each solution point the decision maker determines whether each objective should be decreased freely, decreased to a certain bound, satisfactory, increased to a certain bound, or changed freely. These classifications provide more freedom in developing subsequent solutions. The function is minimized and the process repeats with input from the decision maker until the decision maker chooses a preferred solution. When the vector version is used, the multiobjective proximal bundle (MPB) method is required for the optimization. For the scalar version, any efficient nondifferentiable optimization approach may be used.

NIMBUS is suitable for both differentiable and nondifferentiable multiple objective and single objective optimization problems subject to both linear and nonlinear constraints with bounds on the variables. Miettinen and Mäkelä (2006) have developed a Web-based computer implementation of the method. See Table 1 for a summary of representative interactive methods.

STEM (Step Method)

The following description of STEM, the step method, serves to illustrate an interactive method in more detail. See Benayoun et al. (1971) for a more complete description of this method. The original formulation of the method was designed for maximizing Multiple Objective Linear Programming problems. Although the method has been generalized for nonlinear problems, the original intent is illustrated with an MLOP problem in the following description that is based on Mollaghasemi and Pet-Edwards (1997).

STEM, the step method, is an interactive method that can be used to identify the best compromise solution for multiple objective mathematical programming problems. STEM starts by converting the multiple objective problem into a series of single objective problems. Assume, without loss of generality, that it is desirable to maximize p separate linear objective functions, $f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_p(\boldsymbol{x})$, where $\boldsymbol{x}$ is an n-dimensional vector of decision variables and

$$f_k(\boldsymbol{x}) = \sum_{j=1}^{n} c_{jk} x_j \tag{4}$$

Suppose also that the decision variable values are constrained by $\boldsymbol{x} \in \boldsymbol{X}$, where $\boldsymbol{X}$ is a set of feasible solutions.

Interactive Multiple Objective Mathematical Programming, Table 1 Summary of representative interactive methods

Interactive method	Features	User inputs and actions
Interactive Surrogate Worth Tradeoff Method (ISWT)	Assumed underlying utility function; ε-constraint based approach	Primary objective and limits on other objectives; worth function; trade-off rates
Geoffrion-Dyer-Feinberg (GDF)	Assumed underlying utility function; Frank and Wolfe gradient method	Reference function; trade-off values among objectives; step size; best local solution
Sequential Proxy Optimization Technique (SPOT) Method	Assumed underlying utility function; ε-constraint based approach	Marginal rates of substitution; best local solution
Interactive Goal Programming	Assumed underlying utility function; scalar optimization	Target values on the goals (aspiration levels); Marginal rates of substitution
Tchebycheff Method	Computed weighting vector; weighting method	Best local solution at each iteration
STEM (Step Method)	Classification of objectives (2 classes); weighting method	Objectives to relax and amount of acceptable relaxation
Reference Point Method	Reference point; aspiration levels; perturbation of reference points	Reference point; assess perturbed solutions
Reference Direction Approach	Reference point; works best for MOLP problems	Reference point; most preferred solutions
GUESS Method	Reference point; maximize minimum distance from nadir (weighted)	Reference point (guess); assessment of solution and adjustment of reference point
Satisficing Trade-Off Method (STOM)	Classification of objectives (3 classes); weighting method	Classification of objectives; aspiration levels
Light Beam Search Method	Reference point; outranking	Best and worst values of objectives; indifference, preference, and veto thresholds
Nondifferentiable Interactive BUndle-based optimization System (NIMBUS)	Classification of objectives (5 classes); bundle method	Classification of objectives; aspiration levels

STEM begins by solving p single objective problems separately, as shown in (5):

$$\begin{aligned} &\text{Maximize } f_k(\boldsymbol{x}) = \sum_{j=1}^{n} c_{jk} x_j \\ &\text{Subject to } \boldsymbol{x} \in \boldsymbol{X} \text{ for } k = 1, 2, \ldots, p \end{aligned} \tag{5}$$

The solution to problem (5), $\boldsymbol{x}^k$, results in the maximum value of $f_k(\boldsymbol{x})$ which is represented by $f_k^M(\boldsymbol{x}^k)$. Note that this solution is always a nondominated solution. The values of the remaining objectives at $\boldsymbol{x}^k$ are denoted by $f_k^i(\boldsymbol{x}^k)$ for $i = 1, 2, \ldots, p$ and $k \neq i$. By solving the p optimization problems, a $p \times p$ payoff matrix is then constructed. The objective function values associated with the solutions to problem (5), $f_k^M(\boldsymbol{x}^k)$, represent the ideal solutions. They are placed as the diagonal elements of the payoff matrix and the remaining elements of the payoff matrix, $f_k^i(\boldsymbol{x}^k)$, correspond to the values of the remaining objective functions when the optimal solution, $\boldsymbol{x}^k$, to problem (5), is substituted into the objective functions.

The diagonal elements of the matrix give an outcome associated with an ideal solution. Unfortunately, due to the conflicting nature of the objectives, an ideal solution usually does not exist. However, the payoff matrix provides the decision maker with a better understanding of the system's multiple response surface.

The next step involves identifying the nondominated solution with the least deviation from the ideal solution. This is accomplished by solving the following problem:

$$\begin{aligned} &\text{Minimize} && d \\ &\text{Subject to} && \pi_k\left(f_k^M - f_k(\boldsymbol{x})\right) \leq d, \quad k = 1, 2, \ldots, p \\ &&& \boldsymbol{x} \in \boldsymbol{X}, \quad d > 0 \end{aligned} \tag{6}$$

where

d = maximum deviation of an objective from the ideal solution, and

π_k = relative weight of deviation defined as

$$\pi_k = \left(\frac{a_k}{\sum_{i=1}^{p} a_i} \right)$$

with

$$a_k = \begin{cases} \left[\frac{f_k^M - f_k^m}{f_k^M} \right] \sqrt{\sum_{j=1}^{n} C_{jk}^2} & \text{if } f_k > 0 \\ \left[\frac{f_k^m - f_k^M}{f_k^m} \right] \sqrt{\sum_{j=1}^{n} C_{jk}^2} & \text{if } f_k \leq 0 \end{cases}$$

And $f_k^M (f_k^m)$ is the maximum (minimum) of each column in the payoff matrix. It is evident that the value of a given weight, π_k, is dependent upon the deviation of the objective from its ideal solution. That is, the greater this deviation, the larger the magnitude of π_k.

The decision maker is then presented with the solution to problem (6) (i.e., the solution that results in the least deviation from the ideal solution.) The decision maker must then identify the satisfactory and unsatisfactory objectives and also indicate which objectives in the current solution can be decreased to achieve an improvement in the unsatisfactory objectives. The constraint set in problem (5) is then modified using this information generating a new ideal point and the iterations continue until the decision maker is satisfied with a solution.

STEM has been successfully used in a number of practical applications. For example, Loucks (1977) described how the method was used in a water resources planning project in North Africa. The aim of the project was to aid government officials in choosing the best compromise among three conflicting objectives: maximize water yield, maximize yield reliability, and minimize total cost. The solution yielded a single plan for each irrigation area.

See

- ▶ Decision Analysis
- ▶ Multiobjective Programming
- ▶ Multiple Criteria Decision Making
- ▶ Pareto-Optimal Solution

References

Benayoun, R., de Montgolfier, J., Tergny, J., & Laritchev, O. (1971). Linear programming and multiple objective functions: STEP method (STEM). *Mathematical Programming, 1*, 366–375.

Branke, J., Deb, K., Miettinen, K., & Slowiński, R. (Eds.). (2008). *Multiobjective optimization: Interactive and evolving approaches*. Berlin/Heidelburg: Springer-Verlag.

Buchanan, J. T. (1997). A naïve approach for solving MCDM problems: The GUESS method. *Journal of the Operational Research Society, 48*, 202–206.

Chankong, V., & Haimes, Y. Y. (1978). The interactive surrogate worth trade-off (ISWT) method for multiobjective decision making. In S. Zionts (Ed.), *Multicriteria problem solving* (pp. 42–67). Berlin/Heidelberg: Springer-Verlag.

Chankong, V., & Haimes, Y. Y. (1983). *Multiobjective decision making: Theory and methodology*. New York: Elsevier/North-Holland.

Dyer, J. S. (1972). Interactive goal programming. *Management Science, 19*, 62–70.

Evans, G. W. (1984). An overview of techniques for solving multiobjective mathematical programs. *Management Science, 30*, 1268–1282.

Evans, G. W., Stuckman, B., & Mollaghasemi, M. (1991). Multiple response simulation optimization. In *Proceedings of 1991 winter simulation conference*, Phoenix, Arizona. pp 894–900.

Geoffrion, A. M., Dyer, J. S., & Feinberg, A. (1972). An interactive approach for multicriterion optimization, with an application to the operation of an academic department. *Management Science, 19*, 357–368.

Jaszkiewicz, A., & Slowiński, R. (1994). The light beam search over a non-dominated surface of a multiple objective programming problem. In G. H. Tzeng, H. F. Wand, U. P. Wen, & P. L. Yu (Eds.), *Multiple criteria decision making – Proceedings of the tenth international conference*, Springer-Verlag, New York, pp 87–99.

Korhonen, P. (1987). VIG–a visual interactive support system for multiple criteria decision making. *Belgian Journal of Operations Research, Statistics, and Computer Science, 27*, 3–15.

Korhonen, P., & Laasko, J. (1986). A visual interactive approach for solving the multiple criteria problem. *European Journal of Operational Research, 24*, 277–287.

Loucks, D. P. (1977). An application of interactive multiobjective water resources planning. *Interfaces, 8*(1), 70–75.

Mäkelä, M. M. (1993). Issues of implementing a Fortran subroutine package NSOLIB for nonsmooth optimization, Report 5/1993, University of Jyväskylä, Department of Mathematics, Laboratory of Scientific Computing, Jyväskylä.

Miettinen, K. M. (1994). On the methodology of multiobjective optimization with applications, Doctoral thesis, Report 60, University of Jyväskylä, Department of Mathematics, Jyväskylä.

Miettinen, K. M. (1999). *Nonlinear multiobjective optimization*. Boston/London/Dordrecht: Kluwer Academic.

Miettinen, K. M., & Mäkelä, M. M. (2006). Synchronous approach in interactive multiobjective optimization. *European Journal of Operational Research, 170*, 909–922.
Mollaghasemi, M., & Pet-Edwards, J. (1997). *Making multiple objective decisions*. Los Alamitos, CA: IEEE Computer Society Press.
Nakayama, H. (1989). Sensitivity and trade-off analysis in multiobjective programming. In A. Lewandowski & I. Stanchev (Eds.), *Methodology and software for interactive decision support* (Lecture notes in economics and mathematical systems, Vol. 337, pp. 86–93). Berlin: Springer-Verlag.
Sakawa, M. (1982). Interactive multiobjective decision making by the sequential proxy optimization technique: SPOT. *European Journal of Operational Research, 9*, 386–396.
Steuer, R. E. (1986). *Multiple criteria optimization: Theory, computation, and application*. New York: Wiley.
Steuer, R. E., & Choo, E.-U. (1983). An interactive weighted Tchebycheff procedure for multiple objective programming. *Mathematical Programming, 26*, 326–344.
White, D. J. (1990). A bibliography on the applications of mathematical programming multiple-objective methods. *Journal of the Operational Research Society, 41*, 669–691.
Wierzbicki, A. P. (1982). A mathematical basis for satisficing decision making. *Mathematical Modelling, 3*, 391–405.
Zopounidis, C., & Pardalos, P. M. (2010). *Handbook of multicriteria analysis*. Berlin/Heidelberg: Springer-Verlag.

Interchange Heuristic

A type of local improvement heuristic.

See

► Heuristics

Interfering Float

Float which is shared among the activities on a chain or path in a project network, that is, all the activities on the chain have the same float.

See

► Network Planning

Interior Point

In a constrained optimization problem, an interior point is a solution point that is not on the boundary of the solution space S. If S is defined by the set of constraints $\{g_i(\boldsymbol{x}) \leq 0\}$, then $\boldsymbol{x}^0$ in S is an interior point if $g_i(\boldsymbol{x}) < 0$ for all i.

Interior-Point Methods for Conic-Linear Optimization

Tamás Terlaky[1] and Paul T. Boggs[2,3]
[1]Lehigh University, Bethlehem, PA, USA
[2]Sandia National Laboratories, Livermore, CA, USA
[3]National Institute of Standards and Technology, Gaithersburg, MD, USA

Introduction

Even with the success of the simplex method for linear programming (LP), there was from the earliest days of operations research a desire to create an algorithm for solving LP problems that proceeded on a path through the polytope rather than around its perimeter. Interior point methods (IPMs) were first developed in 1950s, analyzed and first implemented in the 1960s. At that time the conclusion was made that IPMs were not competitive with other algorithms, especially with simplex methods. The continuous effort to find a polynomial algorithm for LP problems led to the revitalization of IPMs. In 1984 Karmarkar first proved the polynomial complexity of an IPM, which led to the "Interior Point Revolution" (Wright 2004) in mathematical programming. In this article the motivation for desiring an interior path, the concept of the complexity of solving LP problems, a brief history of the developments in the area, and the research state of the art are discussed, including generalizations to nonlinear problems.

Background

The LP problem in standard form is

$$\begin{array}{ll} \text{minimiz e}_x & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \\ & \boldsymbol{x} \geq \boldsymbol{0}, \end{array} \qquad (\boldsymbol{LP})$$

where $\boldsymbol{c}, \boldsymbol{x} \in R^n$, $\boldsymbol{b} \in R^m$, and $\boldsymbol{A} \in R^{m \times n}$. It will be assumed that the feasible region for the problem **(LP)** has a strictly feasible point, that is, a point $\boldsymbol{x}^0$ such that $\boldsymbol{A}\boldsymbol{x}^0 = \boldsymbol{b}$ and $\boldsymbol{x}^0 > 0$ (i.e., each component of $\boldsymbol{x}^0$ is strictly positive). The simplex method proceeds on a path from vertex to vertex on the boundary of this region, a process that could require many steps to go around a multifaceted feasible region, although in actual practice the method is generally quite efficient. Intuitively, however, a more direct path through the interior of the region is appealing since there exists the possibility of moving through the polytope in very few steps.

A formal analysis of the complexity of the simplex method remained elusive until the famous result of Klee and Minty (1972), who showed with a simple example that the worst case complexity of some variants of the simplex method is exponential. Their example is a slightly out-of-kilter cube (in n dimensions) in which all $2n$ vertices can be visited by a simplex method, i.e., starting at the origin, there is a path through all of the vertices such that the objective function is decreased at each step. It was immediately recognized, however, that no practical simplex method would use this path; thus there was a desire to explain the efficiency of practical simplex methods. Later analyses showed that a simplex method could expect linear performance, thus partially explaining its behavior (Borgwardt 1987; Goldfarb and Todd 1989).

The first algorithm for LP that was proven to have a worst-case polynomial complexity is the ellipsoid algorithm of Khachiyan (1979). Assuming that the optimal set is nonempty, Khachiyan's algorithm first constructs an ellipsoid that is large enough to contain the optimal set. At subsequent iterations, that ellipsoid is shrunk so that the center of the ellipsoid is the solution to the problem **(LP)**, or after a polynomial number of steps, evidence for the nonexistence of optimal solution is derived. For his method, Khachiyan proved that the complexity is $O(n^4L)$, where L is the number of bits necessary to specify the problem. Unfortunately, the algorithm also seemed to have an expected performance of similar complexity, and was quickly shown to be noncompetitive in practice. Note that Khachiyan's algorithm is not an interior-point method.

Interior-point methods (IPMs) seek to approach the optimal solution through a sequence of points that are always strictly feasible. Such methods have been known for a long time, but for reasons explained below were not considered to be effective. One of the earliest IPMs is the barrier method originally proposed in the 1950s by Frish (1954). When applied to problems with only inequality constraints, it is typically used in conjunction with the sequential unconstrained minimization technique that is described more generally in Fiacco and McCormick (1968). In the primal barrier method for problem **(LP)**, the following log-barrier function problem with equality constraints is formed:

$$\begin{array}{ll} B(\boldsymbol{x}, \mu) = & \boldsymbol{c}^T\boldsymbol{x} - \mu \sum_{i=1}^{m} \log x_i \\ \text{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \end{array}$$

where μ is a positive parameter. Given a positive value of μ and a strictly feasible point $\boldsymbol{x}^0$, the equality constrained barrier problem

$$\begin{array}{ll} \text{minimize}_x & B(\boldsymbol{x}, \mu) \\ \text{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \end{array} \qquad (\textbf{BP})$$

is solved approximately, i.e., a vector $\boldsymbol{x}^1$ is calculated that satisfies the equality constraints and is close to the true minimum of the barrier function. (Note that minimizing a strictly convex function with linear equality constraints is not difficult). Clearly, $\boldsymbol{x}^1$ will remain strictly feasible since the log-barrier function becomes infinite at the boundary of the feasible region. The parameter μ is reduced and **(BP)** is solved again using $\boldsymbol{x}^1$ as the initial starting point. It can be shown that if μ is reduced to zero, for example, by setting $\mu = \frac{\mu}{2}$ at the beginning of each iteration, the resulting sequence $\{\boldsymbol{x}^i\}$ will converge to $\boldsymbol{x}^*$, an optimal solution to **(LP)**. The set of the minimizers $\{\boldsymbol{x}(\mu) \mid \mu > 0\}$ of the barrier function $B(\boldsymbol{x}, \mu)$ gives a smooth analytic curve, the so-called central path (Sonnevend 1985).

Another interior-point approach, called the method of centers was suggested by Huard (1967). This method is initiated with a strictly feasible point $\boldsymbol{x}^0$

and computes the so-called analytic center of the polytope formed by the intersection of the original polytope and the half space of points corresponding to an objective function value less than $\boldsymbol{c}^T\boldsymbol{x}^0$. The analytic center of this bounded polytope is defined to be the maximum of the function

$$C(\boldsymbol{x}, \boldsymbol{c}^T\boldsymbol{x}^0) = (\boldsymbol{c}^T\boldsymbol{x}^0 - \boldsymbol{c}^T\boldsymbol{x}) \prod_{i=0}^{n} \boldsymbol{x}_i.$$

The function $C(\boldsymbol{x}, \boldsymbol{c}^T\boldsymbol{x}^0)$ is clearly zero on the boundary and positive in that part of the interior of the polytope that corresponds to lower values than $\boldsymbol{c}^T\boldsymbol{x}^0$ of $\boldsymbol{c}^T\boldsymbol{x}$, and thus has a maximum, say $\boldsymbol{x}^1$. The level set is then redefined using $\boldsymbol{x}^1$ in place of $\boldsymbol{x}^0$, and the process repeated. It can be shown that the sequence $\{\boldsymbol{x}^i\}$ converges to an optimal solution $\boldsymbol{x}^*$. Moreover, the set of the minimizers $\{\boldsymbol{x}(\gamma)\}$ of the function $C(\boldsymbol{x}, \gamma)$, where γ runs from the maximum until the minimum of $\boldsymbol{c}^T\boldsymbol{x}$ on the feasible set coincides with the central path.

Early IPMs also include that of Dikin (1967). This method begins each iteration by scaling the variables so that the current, strictly feasible point is transformed to the vector of all ones, a point well away from the boundary in the affinely scaled space. A steepest descent step is then applied to this scaled problem, and the resulting point is transformed back to the original space to obtain the next iterate. The advantage of this idea is that the steepest descent step in the original space can be extremely short if the current iterate is close to the boundary, whereas long steps are always possible in the transformed space. This method is known as the affine scaling algorithm. Modern variants of log-barrier methods and the methods of centers enjoy polynomial complexity, while the polynomiality of Dikin's affine scaling method is still open. There is strong belief that primal or dual affine scaling methods are not polynomial, but variants of primal-dual affine scaling methods enjoy polynomial complexity.

All these early methods relate to Karmarkar's method, as discussed below. All of them were tried and compared with the simplex method in the 1970s, but none was seen to be competitive for two principal reasons. First, almost all IPMs (see the next section) require at each step the solution to a linear system of equations of the form

$$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{D}\boldsymbol{A}\boldsymbol{u} = \boldsymbol{r}, \tag{1}$$

where $\boldsymbol{u}$ and $\boldsymbol{r}$ are n-vectors and $\boldsymbol{D}$ is an appropriate positive definite diagonal matrix. It was not until the 1970s that there were sufficiently powerful linear algebra routines that were able to explore the sparsity structure of $\boldsymbol{A}^T\boldsymbol{D}\boldsymbol{A}$ to solve such systems efficiently. Second, IPMs tend to outperform simplex methods on large problems that were well beyond the capabilities of the computers of the 1960s. Third, IPMs typically need more memory and need better floating point calculations than simplex methods. While IPMs were put aside in the 1970s, significant advances in numerical linear algebra were made and, of course, in computational capacity and speed. Interest in IPMs was then revitalized by the announcement of Karmarkar (1984) that he had developed an IPM that had provable polynomial complexity and was competitive with the simplex method. In fact he claimed a factor of 100 speed-up, compared with the state-of-the-art simplex solver MPSX. Karmarkar's procedure begins with a strictly feasible point and then embeds the problem **(LP)** in a space of one dimension higher, in which the feasible point is the center of the higher dimensional polytope. As in the affine scaling algorithm, a "good" step can then be taken in this space and the new point projected back to the original space to obtain the next iterate. Karmarkar's method and its relatives, including the affine scaling algorithm and barrier methods, were studied intensively. Since then, thousands of papers have been written on both the theoretical and computational aspects of IPMs for LP and on the extension of these ideas to quadratic and more general nonlinear programming problems. IPMs opened a new age in the theory, implementations and applications of mathematical optimization, mathematical programming.

IPMs for LP

In the theoretical arena, there has been considerable interest in improving the bound on the number of iterations required to solve an LP problem. It was observed early that IPMs could be cast in such a way as they follow a continuous path, or trajectory, the central path from an initial interior feasible point to

an optimal solution. The central path enjoys many appealing properties. Much analysis of these trajectories and of algorithms based on following the central trajectory has been performed. Several versions of the algorithms described below have also been extensively analyzed. The best theoretical results for these methods demonstrate a complexity that is $O(\sqrt{n}L)$ steps with a quadratic asymptotic rate of convergence. The computationally most successful IPMs for solving an LP problem are based on using a primal-dual formulation and applying Newton's method to the system of equations arising from the barrier method, i.e., by perturbing the optimality conditions. Specifically, the dual problem to **(LP)** is

$$\begin{aligned} \text{maximize}_{y,s} \quad & \boldsymbol{b}^T\boldsymbol{y} \\ \text{subject to } & \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c} \qquad (\mathbf{DP}) \\ & \boldsymbol{s} \geq 0, \end{aligned}$$

where $\boldsymbol{y} \in R^m$ and $\boldsymbol{s} \in R^n$. By the duality theorem of **LP**, at an optimal solution pair the duality gap is zero, thus

$$\boldsymbol{x}|^T\boldsymbol{s} = \boldsymbol{c}^T\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{y} = 0.$$

The optimality conditions for the primal-dual problem can thus be formulated as

$$\begin{aligned} \boldsymbol{A}\boldsymbol{x} \quad & = \boldsymbol{b}, \quad \boldsymbol{x} \geq 0, \\ \boldsymbol{A}^T\boldsymbol{y} + \boldsymbol{s} \quad & = \boldsymbol{c}, \quad \boldsymbol{s} \geq 0, \\ \boldsymbol{X}\boldsymbol{S}\boldsymbol{e} \quad & = 0, \end{aligned}$$

where $\boldsymbol{X}$ denotes the diagonal matrix with x_i as its i^{th} diagonal element. The last conditions are called the complementarity conditions, because they require that at least one of the complementary pair of variables (x_i, s_i) be zero. By perturbing the complementarity conditions, the system

$$\begin{aligned} \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \quad & \boldsymbol{x} \geq 0, \\ \boldsymbol{A}^T\boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c}, \quad & \boldsymbol{s} \geq 0, \qquad (\mathbf{CP}) \\ \boldsymbol{X}\boldsymbol{S}\boldsymbol{e} = \mu\boldsymbol{e}, & \end{aligned}$$

is obtained, where $\boldsymbol{e}$ denotes the vector with all coordinates equal to one. This perturbed system **(CP)** coincides with the Karush-Kuhn-Tucker (first order) optimality conditions of the primal $\mathrm{B}(\boldsymbol{x}, \mu)$, the dual $\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} + \mu\sum_{\mathrm{i}=1}^{\mathrm{n}} \log \mathrm{s_i}$ and the primal-dual $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{s} - \mu\sum_{\mathrm{i}=1}^{\mathrm{n}} \log \mathrm{x_i s_i}$ barrier functions as well, so it can be concluded that the set of solutions $\{\boldsymbol{x}(\mu)|\mu > 0\}$ and $\{(\boldsymbol{y}(\mu), \boldsymbol{s}(\mu))|\mu > 0\}$ is the primal and dual central path, respectively.

The Newton Step

Given an interior feasible point $\boldsymbol{x} > 0, \boldsymbol{y}, \boldsymbol{s} > 0$, a Newton step can be made to solve the system **(CP)**. The goal is to compute the displacements $(\Delta\boldsymbol{x}, \Delta\boldsymbol{y}, \Delta\boldsymbol{s})$ such that

$$\begin{aligned} \boldsymbol{A}(\boldsymbol{x} + \Delta\boldsymbol{x}) &= \boldsymbol{b}, \\ \boldsymbol{A}^T(\boldsymbol{y} + \Delta\boldsymbol{y}) + (\boldsymbol{s} + \Delta\boldsymbol{s}) &= \boldsymbol{c}, \\ (\boldsymbol{X} + \Delta\boldsymbol{X})(\boldsymbol{S} + \Delta\boldsymbol{S})\boldsymbol{e} &= \mu\boldsymbol{e}. \end{aligned}$$

By neglecting the second-order term in the last set of equations and using that $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s})$ is interior feasible, the Newton equation system is obtained:

$$\begin{aligned} &\boldsymbol{A}\Delta\boldsymbol{x} = 0, \\ &\boldsymbol{A}^T\Delta\boldsymbol{y} + \Delta\boldsymbol{s} = 0, \\ &\boldsymbol{X}\Delta\boldsymbol{S}\boldsymbol{e} + \boldsymbol{S}\Delta\boldsymbol{X}\boldsymbol{e} = \mu\boldsymbol{e} - \boldsymbol{X}\boldsymbol{S}\boldsymbol{e}. \end{aligned}$$

Since the matrix $\boldsymbol{A}$ has full rank, the Newton system has a unique solution, which can be obtained by first expressing

$$\begin{aligned} \Delta\boldsymbol{s} &= \boldsymbol{X}^{-1}(\mu\boldsymbol{e} - \boldsymbol{X}\boldsymbol{S}\boldsymbol{e}) - \boldsymbol{X}^{-1}\boldsymbol{S}\Delta\boldsymbol{X}\boldsymbol{e} \\ &= \mu\boldsymbol{X}^{-1}\boldsymbol{e} - \boldsymbol{S}\boldsymbol{e} - \boldsymbol{X}^{-1}\boldsymbol{S}\Delta\boldsymbol{X}\boldsymbol{e} \end{aligned}$$

from the last equations, which leads to the so-called augmented system:

$$\begin{pmatrix} 0 & \boldsymbol{A} \\ \boldsymbol{A}^T & -\boldsymbol{X}^{-1}\boldsymbol{S} \end{pmatrix} \begin{pmatrix} \Delta\boldsymbol{y} \\ \Delta\boldsymbol{x} \end{pmatrix} = \begin{pmatrix} 0 \\ \boldsymbol{S}\boldsymbol{e} - \mu\boldsymbol{X}^{-1}\boldsymbol{e} \end{pmatrix}.$$

This system has a symmetric indefinite coefficient matrix. Expressing Δx as a function of Δy

$$\Delta\boldsymbol{x} = \boldsymbol{S}^{-1}\boldsymbol{X}\boldsymbol{A}^T\Delta\boldsymbol{y} + \boldsymbol{x} - \mu\boldsymbol{S}^{-1}\boldsymbol{e},$$

then the so-called normal equation system

$$(\boldsymbol{AXS}^{-1}\boldsymbol{A}^T)\Delta\boldsymbol{y} = \boldsymbol{x} - \mu\boldsymbol{S}^{-1}\boldsymbol{e}$$

is obtained, where the coefficient matrix is symmetric and positive definite. Highly efficient sparse matrix techniques are available to solve either the augmented or the normal equation system. Having the displacement vectors, an appropriate step-length $\alpha > 0$ needs to be determined to get the new iterates:

$$\begin{aligned}\boldsymbol{x} &:= \boldsymbol{x} + \alpha\Delta\boldsymbol{x},\\ \boldsymbol{y} &:= \boldsymbol{y} + \alpha\Delta\boldsymbol{y},\\ \boldsymbol{s} &:= \boldsymbol{s} + \alpha\Delta\boldsymbol{s}.\end{aligned}$$

Centrality Measures

To determine the appropriate step-size, the deviation of the iterates from the central path needs to be measured. Besides the barrier functions themselves, various centrality measures have been developed. Two such measures are presented here.

Observe that on the central path all the coordinates of the vector $\boldsymbol{XSe}$ are equal. This observation indicates that the proximity measure

$$\Delta_c(\boldsymbol{x},\boldsymbol{s}) := \frac{\max_i(\boldsymbol{x}_i,\boldsymbol{s}_i)}{\min_i(\boldsymbol{x}_i,\boldsymbol{s}_i)},$$

which equals to one on the central path, is an appropriate measure of centrality. The use of this simple measure leads to $O(nL)$ complexity, what is $O(\sqrt{n})$ worse than the best known iteration complexity to date. Due to its simplicity, this centrality measure is frequently used in practice. Another proximity measure can be defined as follows:

$$\delta_0(\boldsymbol{x},\boldsymbol{s},\mu) := \frac{1}{2}\left\|\left(\frac{\boldsymbol{XS}}{\mu}\right)^{\frac{1}{2}}\boldsymbol{e} - \left(\frac{\boldsymbol{XS}}{\mu}\right)^{-\frac{1}{2}}\boldsymbol{e}\right\|.$$

This centrality measure assumes the value zero on the central path. By using this proximity, polynomial IPMs are designed with iteration complexity $O(nL)$ and $O(\sqrt{n}L)$ as well.

Generic Interior Point Newton Algorithm.
Inputs:
proximity parameter κ;
accuracy parameter $\varepsilon > 0$;
variable damping factor α;
update parameter $0 < \theta < 1$;
$(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0)$, $\mu^0 \le 1$ subject to $(\boldsymbol{x}^0,\boldsymbol{s}^0) > 0$ and $\delta(\boldsymbol{x}^0,\boldsymbol{s}^0;\mu^0) \le \kappa$.
begin
  $\boldsymbol{x} := \boldsymbol{x}^0$; $\boldsymbol{y} := \boldsymbol{y}^0$; $\boldsymbol{s} := \boldsymbol{s}^0$; $\mu := \mu^0$;
  while $n\mu \ge \varepsilon$ **do**
  begin
    $\mu := (1-\theta)\mu$;
    while $\delta(\boldsymbol{x},\boldsymbol{s};\mu) \ge \kappa$
      $\boldsymbol{x} := \boldsymbol{x} + \alpha\Delta\boldsymbol{x}$;
      $\boldsymbol{y} := \boldsymbol{y} + \alpha\Delta\boldsymbol{y}$;
      $\boldsymbol{s} := \boldsymbol{s} + \alpha\Delta\boldsymbol{s}$;
    end
  end
end

The following crucial issues remain: How to get an initial interior point that satisfies the initialization requirements; how to choose the centrality parameter κ; how to update μ; and how to damp the Newton step, when needed. Initialization strategies will be discussed in the next section; at this moment it is assumed that an initial interior point, with $\mu = 1$, on the central path is given. The following two parameter choices allow polynomial complexity proof.

(1) **primal-dual log-barrier algorithm with full Newton steps**

This IPM enjoys the best complexity known to date. The following parameter choices are made:

- $\delta(\boldsymbol{x},\boldsymbol{s},\mu) := \delta_0(\boldsymbol{x},\boldsymbol{s},\mu)$;
- $\mu^0 := 1$;
- $\theta := \frac{1}{2\sqrt{n}}$;
- $\kappa = \frac{1}{\sqrt{2}}$;
- $(\Delta\boldsymbol{x},\Delta\boldsymbol{y},\Delta\boldsymbol{s})$ is the Newton step;
- $\alpha = 1$.

Theorem 1. Theorem II.52 in Roos, Terlaky and Vial (1997) *With the given parameter set, the full Newton step algorithm requires not more than*

$$\left\lceil 2\sqrt{n}\log\frac{n}{\varepsilon}\right\rceil$$

iterations to produce a feasible solution $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s})$ *such that* $\Delta_0(\boldsymbol{x},\boldsymbol{s},\mu) \le \kappa$ *and* $n\mu \le \varepsilon$.

(2) **large-update primal-dual log-barrier algorithm**

The following parameter choices are made:

- $\delta(\boldsymbol{x},\boldsymbol{s},\mu) := \delta_0(\boldsymbol{x},\boldsymbol{s},\mu)$;

- $\mu^0 := 1$;
- $0 < \theta < \frac{n}{n+\sqrt{n}}$;
- $\kappa = \frac{\sqrt{R}}{2\sqrt{1+\sqrt{R}}}$, where $R = \frac{\theta\sqrt{n}}{1-\theta}$;
- $(\Delta \boldsymbol{x}, \Delta \boldsymbol{y}, \Delta \boldsymbol{s})$ is the Newton step;
- α is the result of a line search, when along the search direction the primal-dual log-barrier function

$$\boldsymbol{x}^T \boldsymbol{s} - n \sum_{i=1}^{n} \log \boldsymbol{x}_i \boldsymbol{s}_i$$

is minimized.

Theorem 2. Theorem II.74 in Roos, Terlaky and Vial (1997) *With the given parameter set, the large update primal-dual log-barrier algorithm requires not more than*

$$\left\lceil \frac{1}{\theta} \left\lceil 2 \left(1 + \sqrt{\frac{\theta\sqrt{n}}{1-\theta}} \right)^4 \right\rceil \log \frac{n}{\varepsilon} \right\rceil$$

iterations to produce a feasible solution $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s})$ *such that* $\delta_0(\boldsymbol{x}, \boldsymbol{s}, \mu) \leq \kappa$ *and* $n\mu \leq \varepsilon$.

With the choice $\theta = \frac{1}{2}$, the total complexity becomes $O\left(n \log \frac{n}{\varepsilon}\right)$, while the choice $\theta = \frac{K}{\sqrt{n}}$, with any fixed positive value K gives $O\left(\sqrt{n} \log \frac{n}{\varepsilon}\right)$ complexity.

Hundreds of polynomial time IPMs have been developed since 1984, including projective methods; predictor-corrector methods; small- and large-update methods; higher-order methods that are based on higher than first-order approximation of the central path, volumetric barrier methods, self-regular IPMs. Several variants enjoy quadratic or superlinear convergence to an optimal solution.

On Finding an Exact Solution

IPMs follow the central path, and the iterates converge to a maximally complementary solution. However, IPMs not only converge to a solution but also allow identification of exact solutions after a finite number of iterations. This is because the size of the variables can be bounded along, or close to, the central path. An absolute lower bound can be given for the variables that converge to a positive value, while an upper bound, depending on the parameter μ, can be given for the variables that converge to zero. If μ is small enough, then the algorithm stops. The "small" variables can be rounded-off to zero, while the "large" variables can be modified to to get an an exact strictly complementary solution pair. This is a strongly polynomial rounding procedure, see Roos, Terlaky and Vial (1997) and Ye (1997).

The rounding procedure provides an exact strictly complementary solution. When an optimal basis is needed then from an optimal solution pair, an optimal basis can be obtained in strongly polynomial time. The optimal basis identification procedure is due to Megiddo; it can be found in the books by Terlaky (1996) and Roos, Terlaky and Vial (1997). This basis identification algorithm is successfully implemented in various commercial packages.

Initialization

IPMs need an interior feasible point to start with. "Interior" essentially means that having a feasible solution for which all the inequalities hold with strict inequality. Further, in the polynomial IPM variants given above, it is required that the initial solution is close to the central path. To find such solutions the following two methods were developed.

Infeasible IPMs: An infeasible interior point for both the primal and dual LP problem can easily be selected. Choose $\boldsymbol{x}^0 = \boldsymbol{e}$, $\boldsymbol{y}^0 = 0$ and $\boldsymbol{s}^0 = \boldsymbol{e}$, where $\boldsymbol{e}^T = (1, \cdots, 1)$. Then clearly $\boldsymbol{x} > 0$ and $\boldsymbol{s} > 0$, i.e., they are interior points of the positive orthant, but they are infeasible, because usually the primal $\boldsymbol{r}_p = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{e}$ and dual $\boldsymbol{r}_d = \boldsymbol{c} - \boldsymbol{s}^0 - \boldsymbol{A}^T\boldsymbol{y}^0 = \boldsymbol{c} - \boldsymbol{e} - \boldsymbol{A}^T 0 \neq 0$ residuals are not zero. Nevertheless, starting from such an infeasible interior point, the working horse of IPMs, the Newton process, can be launched. In this case the Newton system becomes

$$\begin{aligned} \boldsymbol{A}\Delta\boldsymbol{x} &= \boldsymbol{r}_p, \\ \boldsymbol{A}^T\Delta\boldsymbol{y} + \Delta\boldsymbol{s} &= \boldsymbol{r}_d, \\ \boldsymbol{X}\Delta\boldsymbol{S}\boldsymbol{e} + \boldsymbol{S}\Delta\boldsymbol{X}\boldsymbol{e} &= \mu\boldsymbol{e} - \boldsymbol{X}\boldsymbol{S}\boldsymbol{e}. \end{aligned}$$

This differs from the original Newton system just in the right hand side of the first two equations. Since the coefficient matrix remains the same, the new Newton system can be solved the same way as in the feasible case. During infeasible IPMs the primal and dual residual and the complementarity gap is reduced to

zero simultaneously. Infeasible IPMs allow polynomial complexity proofs. However, the worst case complexity of infeasible IPMs is $O(nL)$ iterations, a factor of $O(\sqrt{n})$ worse than the complexity of feasible IPMs.

Initialization by embedding: Another theoretically and for infeasible or unbounded problems also practically more sound initialization procedure is the self-dual embedding strategy, originally proposed by Ye, Todd and Mizuno (see the books Ye 1997; Jansen 1997; and Roos, Terlaky and Vial, 1997). The roots of this approach can be recognized in Goldman and Tucker's homogeneous self-dual model. By putting the primal and dual constraints together, requiring that the dual objective value be at least as large as the primal one, and finally by homogenizing the system, the Goldman–Tucker model is obtained:

$$\begin{array}{rrrrl} \boldsymbol{A}\boldsymbol{x} & & -\boldsymbol{b}\tau & = 0, & \boldsymbol{x} \geq 0, \quad \tau \geq 0, \\ -\boldsymbol{A}^T\boldsymbol{y} & -\boldsymbol{s} & +\boldsymbol{c}\tau & = 0, & \boldsymbol{s} \geq 0, \\ +\boldsymbol{b}^T\boldsymbol{y} & -\boldsymbol{c}^T\boldsymbol{x} & & -\rho = 0, & \rho \geq 0. \end{array}$$

Any solution of this system where $\tau > 1$ gives an optimal solution to the LP problem; moreover if this system has no solution with $\tau > 0$, then either the primal or the dual LP is infeasible. Due to the Weak Duality Theorem, this problem cannot have an interior feasible solution, but allows the following embedding with a perfectly centered interior feasible solution.

Let $\boldsymbol{y}^0 = 0, \boldsymbol{x}^0 = \boldsymbol{e}, \boldsymbol{s}^0 = \boldsymbol{e}, \tau^0 = 1, \rho^0 = 1, \ \vartheta^0 = 1$ be the initial value of the variables and let $\bar{\boldsymbol{b}} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{e}$, $\bar{\boldsymbol{c}} = \boldsymbol{c} - \boldsymbol{e}$, $\gamma = \boldsymbol{c}^T\boldsymbol{e} + 1$, $\beta = \gamma - \bar{\boldsymbol{c}}^T\boldsymbol{e} = \boldsymbol{e}^T\boldsymbol{e} + 1$. For the embedding problem **(SP)**

$$\begin{array}{l} \min \beta\vartheta \\ \text{Subject to} \\ \begin{array}{rrrrrl} \boldsymbol{A}\boldsymbol{x} & & -\boldsymbol{b}\tau & +\bar{\boldsymbol{b}}\vartheta & & = 0, \boldsymbol{x} \geq 0, \quad \tau \geq 0, \\ -\boldsymbol{A}^T\boldsymbol{y} & -\boldsymbol{s} & +\boldsymbol{c}\tau & -\bar{\boldsymbol{c}}\vartheta & & = 0, \boldsymbol{s} \geq 0 \\ +\boldsymbol{b}^T\boldsymbol{y} & -\boldsymbol{c}^T\boldsymbol{x} & & +\gamma\vartheta & -\rho & = 0, \rho \geq 0, \\ -\bar{\boldsymbol{b}}^T\boldsymbol{y} & +\bar{\boldsymbol{c}}^T\boldsymbol{x} & -\gamma\vartheta & & & = -\beta, \end{array} \end{array}$$

the following statements hold:

- the given initial point is interior feasible;
- it is on the central path of the embedding problem with $\mu = 1$;
- this the embedding problem can be solved by any feasible IPM;
- **(SP)** is self dual, thus its optimal value is zero, hence at optimum $\vartheta = 0$;
- a solution of **(SP)** with $\vartheta = 0, \tau > 0$ gives an optimal solution pair for LP;
- a solution of **(SP)** with $\vartheta = 0, \rho > 0$ gives evidence of primal or dual infeasibility of LP.

When problem **(SP)** is solved by any feasible IPM, the linear algebra can be organized so that an iteration costs hardly any more computational effort than an iteration at the original problem. Further, IPMs provide a strictly complementary solution, thus $\tau\rho = 0$ and $\tau + \rho > 0$ holds for the solution produced by IPMs. As a consequence, it suffices to solve **(SP)** to solve the original problem **LP**. Finally, note that the worst case complexity of IPMs applied to **(SP)** is the same as their complexity when applied to the original problem, because the embedding problem has $m + 2n + 3$ variables, thus the iteration complexity is $O(\sqrt{m + 2n + 3}L = O(\sqrt{n}L)$.

Initialization by embedding not only allows initializing IPMs while preserving the best worse-case complexity, but also provides the most robust method to detect infeasibility of either the primal or the dual LP problem. This is due to the fact that infeasiblity is detected by convergence to an optimal solution of the embedding problem **(SP)**. This is in sharp contrast to the divergence of the iterates when an infeasible IPM is applied to an infeasible or unbounded problem.

Barrier Approaches

Various extensions of interior point methods were developed in the past decades. It is natural to replace the logarithmic barrier $(-\ln t)$ by other barrier functions. Nesterov and Nemirovskii (1994) introduced the so-called universal barrier function that allows solution of any smooth convex optimization problem in a polynomial number of iterations. However, such an algorithm is not necessarily polynomial time, because the iterations might not be possible to compute in polynomial time. Copositive optimization (see Bomze et al. 2000), where a linear objective function is optimized over the intersection of an affine subspace and the cone of copositive matrices is a convex optimization problem where the universal barrier approach yields an algorithm for which the number of iterations is polynomial, but a single Newton step cannot be made in polynomial time. Nesterov and Nemirovskii (1994) have also

developed the theory of self-concordant barrier functions that allowed development of polynomial time algorithms for large classes of convex optimization problems.

Other approaches include the volumetric barrier method that in a cutting plane framework allows solution of LP problems in polynomial time. When a polynomial time separation oracle is available, the complexity of the volumetric barrier cutting plane IPM is independent of the number of constraints, its complexity depending only on the dimension of the problem. The significance of this result is that a large set of combinatorial optimization problems, previously solvable in polynomial time only by the ellipsoid method, can be solved with better complexity, and so IPMs completely supersede the ellipsoid method.

Self-regular barrier functions, by Peng, Roos and Terlaky (2002), allow design of IPMs that operate in a large neighborhood of the central path, while having almost the same complexity as small-update IPMs.

IPMs Versus Simplex Methods

An extensive comparison of IPMs and simplex methods can be found in Illés and Terlaky (2002). Some important aspects are reviewed here.

Efficiency: IPMs have been extremely successful in solving some very large linear programs, but they do not completely replace the simplex method. The best algorithm is, of course, problem-dependent, but generally speaking the IPMs perform better on larger problems and on problems that allow efficient exploitation of the numerical linear algebra. Specifically, as noted above, if the structure of $\boldsymbol{A}^T\boldsymbol{D}^2\boldsymbol{A}$ can be exploited to solve either the augmented or the normal equation system quickly, then IPMs have an advantage. An example of such an $\boldsymbol{A}$ matrix arises in multi-period resource planning problems where $\boldsymbol{A}$ has a staircase structure. The matrix $\boldsymbol{A}^T\boldsymbol{D}^2\boldsymbol{A}$ is then block diagonal and can usually be factored efficiently. IPMs also perform better on (highly) degenerate problems that often arise in large-scale applications, because degeneracy hardly effects the performance of IPMs.

Basic solution versus strictly complementary solution: IPMs by nature generate a maximally complementary optimal solution pair, while simplex-based solvers generate an optimal basis. Although users are used to basic solutions, there are many practical situations when a strictly complementary solution is desirable. In these situations simplex methods are clearly outperformed by IPMs, because finding a strictly complementary solution from an optimal basis solution is not easier than solving the original problem.

On the other hand, when an optimal basis is needed, then as discussed earlier, the identification of an optimal basis can be made in strongly polynomial time. Such algorithms are efficiently implemented in state-of-the-art software packages. So it can be concluded, that even in this case, the choice between IPMs and simplex methods should be made on the basis of their ability to solve the original problem efficiently, i.e., on the basis of which is better able to exploit the sparsity structure of the problem.

Sensitivity analysis: Post-optimality analysis has tremendous importance in practice. Linear programming software reports sensitivity analysis based on the obtained optimal basis, which answers the questions:

Over what range of the parameter values does the obtained optimal basis remain optimal, and how does the optimal function value change with this optimal basis solution?

However, typically the user would prefer to know the answers to the following questions:

What is the rate of change (shadow price, reduced cost) of the optimal objective value when a parameter changes, and for what intervals does this rate remain valid?

To answer this question correctly requires finding the possibly different left- and right-hand derivatives of the optimal value function. For this purpose the solution of some smaller linear problems is needed. Both an optimal basic solution and a strictly complementary solution are appropriate to set up those subproblems, which can again be solved either by simplex or IPM solvers. A thorough treatment of correct sensitivity analysis can be found in Jansen (1997) and in Roos, Terlaky and Vial (1997).

IPMs, Klee-Minty Examples and Complexity Bounds

This section reviews some cases when the central path exhibits extreme behavior.

Klee-Minty Examples for IPMs

The Klee-Minty cube (1972) for which simplex algorithms may take an exponential number,

i.e., $2^n - 1$ pivot steps, is given in the following form, where the convention $x_0 = 0$ is used, and τ is a small positive factor by which the unit cube $[0,1]^n$ is perturbed.

$$\min \; x_n$$

$$\text{s.t. } \tau x_{k-1} \le x_k \le 1 - \tau x_{k-1} \text{ for } k = 1, \ldots, n.$$

This optimization problem has $N = 2n$ constraints and n variables. The set of feasible solutions is a perturbed n-dimensional cube. Starting from the vertex $(0, \ldots, 0, 1)^T$, simplex methods may visit all the vertices of the feasible set.

Although IPMs are polynomial time algorithms, the complexity of central-path following IPMs depend on the number of inequalities in the problem and the condition number, or input length, see e.g., Roos, Terlaky and Vial (2006). Deza et al. (2006) show that in any dimension, the central path of an LP problem, where the feasible set is the Klee-Minty cube, follows the simplex path. More precisely, by adding exponentially many redundant constraints that are parallel to the facets passing through the optimal vertex of the Klee-Minty cube, the central path can be forced to visit a predefined arbitrary small neighborhood of all the vertices of the Klee-Minty cube in the same order as simplex methods do. In other words, this central path has $2^n - 2$ almost-90° turns. All redundant hyperplanes are at the same distance from the Klee-Minty cube, and the number of inequalities in dimension n is $N = O(n^2 2^{6n})$. In subsequent papers the number of redundant inequalities are reduced significantly. By decaying geometrically the distances of the redundant constraints to the corresponding facets, Deza, Nematollahi and Terlaky (2008) show that the number of the inequalities N can be reduced to $O(n^3 2^{2n})$ and that after $O(\sqrt{N}n)$ iterations, a standard rounding procedure allows identification of the optimal solution. This results tighten the gap between iteration-complexity lower and upper bounds.

The tightest result is given by Nematollahi and Terlaky (2008). The redundant constraints are placed parallel to the coordinate hyperplanes at geometrically decaying distances, so only $N = O(n2^{2n})$ redundant inequalities are needed to force the central path to follow the simplex path of the n-dimensional Klee-Minty cube, yielding an $O(n\frac{3}{2}2^n)$ iteration-complexity upper bound. The iteration complexity lower bound remains $\Omega(2^n)$, which follows from the fact that the central path follows the simplex path arbitrarily close. As a result, the gap between the iteration complexity upper and lower bounds is almost closed, because the lower bound for the number of iterations is $\Omega(\sqrt{\frac{N}{\ln N}})$, while the upper bound is $O(\sqrt{N}\ln N)$.

The Tight Klee-Minty Construction and Complexity Bounds

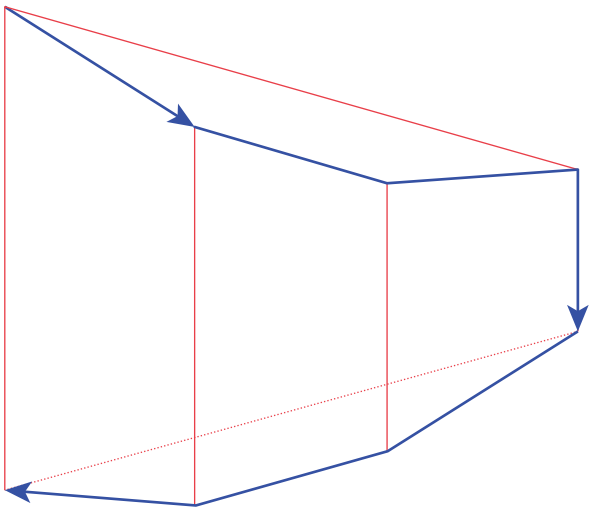
To force the central path to follow the simplex path, redundant constraints are introduced that are given by the inequalities $d_k + x_k \ge 0\}$, for $k = 1, \ldots, n$, where d_k is the distance to the respective coordinate plane. These redundant constraints are repeated h_k times, where the specific values of h_k are given in the sequel. While adding redundant constraints do not change the set of feasible solutions, the analytic center and the central path change. The redundant Klee-Minty example of Nematollahi and Terlaky (2008) is given as:

$$\begin{aligned} &\min && x_n \\ &\text{s.t.} \\ &&& \tau x_{k-1} \le x_k \le 1 - \tau x_{k-1} \text{ for k} = 1, \ldots, \text{n}, \\ &0 \le d_k + x_k \text{ repeated h}_\text{k} \text{ times, for k} = 1, \ldots, \text{n}. \end{aligned}$$

For this example the following parameters are chosen (Fig. 1):

$$\begin{aligned} &\tau = \tfrac{n}{2(n+1)}, \; \delta \le \tfrac{1}{4(n+1)}, \\ &d = \left(\tfrac{1}{\sqrt{\tau^{n-1}}}, \tfrac{1}{\sqrt{\tau^{n-2}}}, \ldots, \tfrac{1}{\sqrt{\tau}}, 0\right), \\ &h = \left(\left\lfloor \tfrac{4(1+\sqrt{\tau^{n-1}})}{\sqrt{\tau^{n-1}}\delta} \right\rfloor, \left\lfloor \tfrac{4(1+\sqrt{\tau^{n-2}})(2+\sqrt{\tau^{n-1}})}{\tau\sqrt{\tau^{n-2}}\delta} \right\rfloor, \right. \\ &\qquad \left. \ldots, \left\lfloor \tfrac{4(1+\sqrt{\tau})\prod_{i=2}^{n-1}(2+\sqrt{\tau^i})}{\tau^{n-2}\sqrt{\tau}\delta} \right\rfloor, \left\lfloor \tfrac{4\prod_{i=1}^{n-1}(2+\sqrt{\tau^i})}{\tau^{n-1}\delta} \right\rfloor \right). \end{aligned}$$

Theorem 3. *For the given redundant Klee-Minty example, the iteration-complexity lower and upper bounds for central-path-following interior point methods are* $O(2^n)$ *and* $O(n\frac{3}{2}2^n)$, *respectively. These bounds expressed in terms of the number of inequalities are* $O(\sqrt{\frac{N}{\ln N}})$ *and* $O(\sqrt{N}\ln N)$, *respectively. The gap between the lower and upper bounds is* $O(\ln^2 \text{N})$.



Interior-Point Methods for Conic-Linear Optimization, Fig. 1 The Klee-Minty 3-cube and the simplex path that is traced by the central path

Curvature and Conjectures

Deza, Terlaky and Zinchenko (2008, 2009) present a construction where all the N constraints are non-redundant, and the central path makes $N-4$ sharp turns. For this non-redundant example, the input length and the condition number of the problem grow as the number of inequalities grows. They also relate IPMs to the Hirsch Conjecture, a claim presented by W.M. Hirsch to Dantzig in a letter in 1957 (cf. Dantzig 1963), which asserts a linear upper bound for the diameter of polytopes. Although the conjecture in its original form was disproved by Santos (2010), its weaker variants are still open. Analogous conjectures for the central path curvature are presented by Deza, Terlaky and Zinchenko (2008, 2009), substantiating the relationships and presenting partial results.

Extensions

IPMs were designed for nonlinear problems already in the sixties, see e.g., Fiacco and McCormick (1968). Since then, IPMs have been generalized to large classes of smooth, convex optimization problems. Some polynomially solvable classes are summarize here.

QP, LCP

Convex quadratic optimization problems are solvable in polynomial time by IPMs. The complexity results are analogous to the case of LP. There are two differences: typically the step length is shorter in QP than in LP, but this just adds a constant factor in the complexity estimates. The other, more important difference is that no strictly complementary solution exists in general for QP problems. The consequence is that identifying an exact solution needs more computational effort, and the analysis gets also significantly more involved (see Illés et al. 2000).

Linear complementarity problems (LCPs) are natural generalizations of LP and QP. The solvability of an LCP

$$-\boldsymbol{M}\boldsymbol{x}+\boldsymbol{s}=\boldsymbol{q}, \qquad \boldsymbol{x}\geq 0, \boldsymbol{s}\geq 0, \qquad x_i s_i = 0\ \forall i$$

depends on the properties of the coefficient matrix $\boldsymbol{M}$. The largest polynomially solvable class is the class of LCPs with $P_*(\kappa)$ matrices, for all $\kappa \geq 0$. A matrix is $\boldsymbol{M}$ is a $P_*(\kappa)$ matrix if for all $x \in R^n$ the inequality

$$(1+4\kappa)\sum_{\boldsymbol{x}_i(M\boldsymbol{x})_i>0}\boldsymbol{x}_i(M\boldsymbol{x})_i+\sum_{\boldsymbol{x}_i(M\boldsymbol{x})_i<0}\boldsymbol{x}_i(M\boldsymbol{x})_i\geq 0$$

holds. Clearly $P_*(0)$ matrices are positive semidefinite. The union of the classes $P_*(\kappa)$ for all $\kappa \geq 0$ is the class of P_* matrices, which coincides with the class of sufficient matrices. It is known that LCPs with sufficient matrices are solvable with pivoting methods, thus there is no discrepancy (except worst-case complexity) between the solvability of LCPs by pivot and IPMs. The book of Kojima, Megiddo, Noma and Yoshise (1991) is devoted to IPMs for LCPs.

Conic Linear Optimization Problems

Conic linear optimization (CLO) problems are obtained when the nonnegativity constraints, i.e., the requirement that the variables of an LP problem are in the polyhedral cone of the nonnegative vectors, are replaced by the requirement that the variables belong to a convex cone. A primal-dual pair of CLO problems is given as:

$$\begin{array}{llll} (P)\ \min & \boldsymbol{c}^T\boldsymbol{x} & (D)\ \max & \boldsymbol{b}^T\boldsymbol{y} \\ \text{s.t.} & \boldsymbol{A}\boldsymbol{x}-\boldsymbol{b} \in \mathcal{C}_1 & \text{s.t.} & \boldsymbol{c}-\boldsymbol{A}^T\boldsymbol{y} \in \mathcal{C}_\in^* \\ & \boldsymbol{x} \in \mathcal{C}_2 & & \boldsymbol{y} \in \mathcal{C}_\infty^*, \end{array}$$

where $\boldsymbol{b}, \boldsymbol{y} \in R^m$, $\boldsymbol{c}, \boldsymbol{x} \in R^n$, $\boldsymbol{A}: m \times n$ matrix, C_1, C_2 are convex cones and $C_i^* = \{\boldsymbol{s} \in R^n : \boldsymbol{x}^T\boldsymbol{s} \geq 0, \forall \boldsymbol{x} \in C_i\}$ are the dual cones for $i = 1, 2$.

Large classes of CLO problems are solvable by IPMs in polynomial time. The best known classes of CLO problems are the classes of second-order cone optimization and semidefinite optimization problems. In a second-order cone optimization problem the cones $\mathcal{C}_1$ and $\mathcal{C}_2$ are direct products of second order and linear cones. The second order cone, or ice cream cone, in dimension n is given by

$$S_2^n := \left\{ \boldsymbol{x} \in R^n \; : \; \sqrt{\sum_{i=1}^{n-1} x_i^2} \leq x_n \right\}.$$

Second-order cone optimization problems are studied in Nesterov and Nemirovskii (1994); Vandenberghe and Boyd (1996); and Andersen, Roos, and T. Terlaky (2003).

Another important class of CLO problems, semidefinite optimization (SDO), which has numerous important applications, particularly in control and combinatorics, is discussed in more detail here (see e.g., Nesterov and Nemirovskii 1994; Vandenberghe and Boyd 1996; de Klerk 2002). Let $C, \boldsymbol{A}_i$ for $i = 1, \ldots, n$ be given symmetric matrices and $\boldsymbol{b} \in R^m$. Let X be denote the symmetric matrix of variables. The primal problem of semidefinite optimization can be given as

$$\begin{array}{lll} \text{minimize}_{\,X} & \mathbf{Tr}(CX) & \\ \text{subject to} & \mathbf{Tr}(\boldsymbol{A}_i X) = \boldsymbol{b}_i, & i = 1, \ldots, n, \quad \text{(CLP)} \\ & \boldsymbol{X} \succcurlyeq 0, & \end{array}$$

where $\mathbf{Tr}(\cdot)$ indicates the trace of the given matrix and the positive semidefiniteness of the matrix $\boldsymbol{X}$ is denoted by $\boldsymbol{X} \succcurlyeq 0$. Analogous to LP, the dual problem can be given as

$$\begin{array}{lll} \text{maximize}_{(y,S)} & \boldsymbol{b}^{\mathrm{T}} \boldsymbol{y} & \\ \text{subject to} & \sum_{\mathrm{i}=1}^{\mathrm{n}} \boldsymbol{A}_{\mathrm{i}} \boldsymbol{y}_{\mathrm{i}} + \boldsymbol{S} = \boldsymbol{C}, & \quad \text{(CLD)} \\ & \boldsymbol{S} \succcurlyeq 0. & \end{array}$$

The interior point condition for SDO reads as follows: a feasible solution $\boldsymbol{X}$ and $(\boldsymbol{y}, \boldsymbol{S})$ exists where the matrices $\boldsymbol{X}$ and $\boldsymbol{S}$ are positive definite. The design and analysis of IPMs for SDO are analogous to the case of LP, with two major differences.

First, the duality theory of SDO is weaker. There are SDO problems where both the primal and the dual problem admit an optimal solution, but the duality gap is nonzero; or where the optimal value either from the primal or from the dual side is not attained; or where either of the problems is weakly infeasible. Special techniques have been developed to deal with these pathological cases, see Wolkowicz, Saigal, and Vandenberghe (2000).

The second difference comes from the calculation of the search direction. When the interior point condition holds, the central path is well defined. Again, it is defined as the set of solutions of the perturbed optimality conditions:

$$\begin{array}{lll} \mathbf{Tr}(\boldsymbol{A}_{\mathbf{i}} \boldsymbol{X}) & = \boldsymbol{b}_{\mathbf{i}}, & i = 1, \ldots, n, \boldsymbol{X} \succcurlyeq 0, \\ \sum_{i=1}^{n} \boldsymbol{A}_i \boldsymbol{y}_i + \boldsymbol{S} & = \boldsymbol{C}, & \boldsymbol{S} \succcurlyeq 0, \quad \text{(CCP)} \\ \boldsymbol{X}\boldsymbol{S} & = \mu \boldsymbol{I}. & \end{array}$$

When Newton's method is applied to this system to calculate the search directions, the system

$$\begin{array}{ll} \mathbf{Tr}(\boldsymbol{A}_i \Delta \boldsymbol{X}) & = 0, \quad i = 1, \ldots, n, \\ \sum_{i=1}^{n} \boldsymbol{A}_i \Delta \boldsymbol{y}_i + \Delta \boldsymbol{S} & = 0, \\ \boldsymbol{X}\Delta \boldsymbol{S} + (\Delta \boldsymbol{X})\boldsymbol{S} & = \mu \boldsymbol{I} - \boldsymbol{X}\boldsymbol{S} \end{array}$$

is obtained. When $(\Delta \boldsymbol{X}, \Delta \boldsymbol{y}, \Delta \boldsymbol{S})$ is the solution of this Newton system, then $\Delta \boldsymbol{S}$ is symmetric; however $\Delta \boldsymbol{X}$ is symmetric if and only if $\boldsymbol{X}\boldsymbol{S}$ is a multiple of the unit matrix. As a consequence, additional tools are needed to symmetrize the Newton system. For details the reader is referred to Wolkowicz, Saigal, and Vandenberghe (2000), and Todd (1999).

Several codes have been developed to solve SDO problems. Traditionally nonlinear optimization software has required a functional description of the set of feasible solution, hence not allowing incorporation of conic constraints.

Smooth Structured Nonlinear Programming Problems

Smooth nonlinear programming (NLP) problems are also solvable in polynomial time by using IPMs when a smoothness condition is satisfied. Let a set of convex

functions $f_0(\boldsymbol{x}), f_1(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x}) : R^n \to R$ be given. Further, assume that the NLP problem

$$\begin{aligned} &\text{minimize}_{\mathrm{x}} \quad \mathrm{f}_0(\boldsymbol{x}) \\ &\text{subject to} \quad \mathrm{f}_{\mathrm{i}}(\boldsymbol{x}) \leq 0, \qquad \mathrm{i} = 1, \ldots, \mathrm{n}, \end{aligned}$$

satisfies the interior point condition, i.e., admits a solution for which all constraints hold with strict inequality and its level sets are bounded. Let

$$\phi_\mu(\boldsymbol{x}) := f_0(\boldsymbol{x}) - \mu \sum_{i=1}^{m} \log(-f_i(\boldsymbol{x}))$$

be the log-barrier function. The NLP problem satisfies the smoothness condition – called κ self-concordant – if there exists a $\kappa \geq 0$ for which the inequality

$$|\nabla^3 \phi_\mu(\boldsymbol{x})[h,h,h]| \leq 2\kappa \left\{\nabla^2 \phi_\mu(\boldsymbol{x})[h,h]\right\}^{\frac{3}{2}}$$

holds for any $\boldsymbol{x}$ in the domain of ϕ_μ and for any $h \in R^n$. Here $\nabla^2 \phi_\mu(\boldsymbol{x})[h,h]$ and $\nabla^3 \phi_\mu(\boldsymbol{x})[h,h,h]$ denote the second- and third-order directional derivatives of $\phi_\mu(\boldsymbol{x})$ in the direction $h \in R^n$, respectively.

When the self-concordancy condition is satisfied, IPMs applied to NLP admit polynomial complexity proofs (see den Hertog 1994; Nesterov and Nemirovskii 1994). Implementations of IPMs for NLP problems include the KNITRO package Byrd, Nocedal, and Waltz (2006), and the IPOPT package Wächter (2002).

Concluding Remarks

IPMs allow polynomial-time solution of large classes of smooth convex optimization problems, where new efficiently solvable problem classes such as semidefinite programming and second-order cone optimization have been identified. IPMs are efficient not only in theory, but also in computational practice, often being the only option for solving large-scale structured problems. This has contributed heavily to efficiency improvements in optimization software on the order of 10^6 or more (Bixby 2002). Core theory, such as duality theory and sensitivity analysis, has been rejuvenated (Roos et al. 2006; Koltai and Terlaky 2000; Ghaffari Hadigheh et al. 2007).

IPMs have spread to all areas of optimization. For example, novel robust optimization methodology developed by Ben-Tal and Nemirovskii (2001) has opened new opportunities to solve important problem classes, including problems in truss-topology design, signal processing, VLSI design and robust and intensity modulated radiation therapy treatment (Chu et al. 2008; Craig et al. 2008).

More details on IPMs for LP problems can be found in Gonzaga (1991a,1991b,1992), Goldfarb and Todd (1989), Roos and Terlaky (1997), Roos, Terlaky and Vial (1997), Terlaky (1996), Ye (1997), Wright (1996) and Wright (2004); for more general convex and nonlinear problems, see den Hertog (1994), Nesterov and Nemirovskii (1994), and Wolkowicz, Saigal, and Vandenberghe (2000).

See

- Barrier Functions and their Modifications
- Computational Complexity
- Duality Theorem
- Hirsch Conjecture
- Large-Scale Systems
- Linear Programming
- Newton's Method
- Nonlinear Programming
- Optimization
- Quadratic Programming
- Simplex Method (Algorithm)

References

Andersen, E. D., Roos, C., & Terlaky, T. (2003). On implementing a primal–dual interior–point method for conic quadratic optimization. *Mathematical Programming, 95*(2), 249–277.

Ben-Tal, A., & Nemirovski, A. (2001). *Lectures on modern convex optimization: Analysis, algorithms, and engineering applications* (MPS-SIAM series on optimization). Philadelphia, PA: SIAM.

Bixby, R. E. (2002). Solving real-world linear programs: A decade and more of progress. *Operations Research, 50*(1), 3–15.

Bomze, I. M., Duerr, M., de Klerk, E., Roos, C., Quist, A. J., & Terlaky, T. (2000). On copositive programming and standard quadratic optimization problems. *Journal of Global Optimization, 18*(2), 301–320.

Borgwardt, K. H. (1987). *The simplex method: A probabilistic analysis, algorithms and combinatorics* (Vol. 1). Berlin: Springer.

Byrd, R., Nocedal, J., & Waltz, R. (2006). KNITRO: An integrated package for nonlinear optimization. In G. Di Pillo & M. Roma (Eds.), *Large-scale nonlinear optimization* (Nonconvex optimization and its applications, Vol. 83, pp. 35–59). Berlin: Springer.

Chu, M., Zinchenko, Y., Henderson, S. G., & Sharpe, M. B. (2008). Robust optimization for intensity modulated radiation therapy treatment planning under uncertainty. *Physics in Medicine and Biology, 53*, 3231–3250.

Craig, T., Sharpe, M. B., Terlaky, T., & Zinchenko, Y. (2008). Controlling the dose distribution with gEUD-type constraints within the convex IMRTP framework. *Physics in Medicine and Biology, 53*, 3231–3250.

de Klerk, E. (2002). *Aspects of semidefinite programming: Interior point algorithms and selected applications*. Dordrecht, The Netherlands: Kluwer.

den Hertog, D. (1994). *Interior point approach to linear, quadratic and convex programming*. Dordrecht, The Netherlands: Kluwer.

Deza, A., Nematollahi, E., Peyghami, R., & Terlaky, T. (2006). The central path visits all the vertices of the Klee-Minty cube. *Optimization Methods and Software, 21*, 851–865.

Deza, A., Nematollahi, E., & Terlaky, T. (2008). How good are interior point methods? Klee-Minty cubes tighten iteration-complexity bounds. *Mathematical Programming, 113*, 1–14.

Deza, A., Terlaky, T., & Zinchenko, Y. (2008). Polytopes and arrangements: Diameter and curvature. *Operations Research Letters, 36*, 215–222.

Deza, A., Terlaky, T., & Zinchenko, Y. (2009). The continuous *d*-step conjecture for polytopes. *Discrete and Computational Geometry, 41*, 318–327.

Dikin, I. I. (1967). Iterative solution of problems of linear and quadratic programming. *Soviet Mathematics Doklady, 8*, 674–675.

Fiacco, A. V., & McCormick, G. P. (1968). *Nonlinear programming: Sequential unconstrained minimization techniques*. New York: John Wiley.

Frish, K. R. (1954). Principles of linear programming – the double gradient form of the logarithmic potential method. *Memorandum*, Institute of Economics, University of Oslo, Oslo, Norway.

Ghaffari Hadigheh, A. R., Romanko, O., & Terlaky, T. (2007). Sensitivity analysis in convex quadratic optimization: Simultaneous perturbation of the objective and right-hand-side vectors. *Algorithmic Operations Research, 2*(2), 4–111.

Goldfarb, D., & Todd, M. J. (1989). Linear programming. In G. L. Nemhauser, A. H. G. Rinnooy Kan, & M. J. Todd (Eds.), *Optimization* (pp. 73–170). Amsterdam/New York: North Holland.

Gonzaga, C. C. (1991a). Large-steps path-following methods for linear programming, part I: Barrier function method. *SIAM Journal on Optimization, 1*, 268–279.

Gonzaga, C. C. (1991b). Large-steps path-following methods for linear programming, part II: Potential reduction method. *SIAM Journal on Optimization, 1*, 280–292.

Gonzaga, C. C. (1992). Path following methods for linear programming. *SIAM Review, 34*, 167–224.

Huard, P. (1967). Resolution of mathematical programming with nonlinear constraints by the method of centres. In J. Abadie (Ed.), *Nonlinear programming* (pp. 209–219). Amsterdam: North Holland.

Illés, T., Peng, J., Roos, C., & Terlaky, T. (2000). A strongly polynomial rounding scheme in interior point methods for $P_*(\kappa)$ linear complementarity problems. *SIAM Journal on Optimization, 11*(2), 320–340.

Illés, T., & Terlaky, T. (2002). Pivot versus interior point methods: Pros and cons. *European Journal of Operational Research, 140*(2), 6–26.

Jansen, B. (1997). *Interior point techniques in ptimization. Complexity, sensitivity and algorithms*. Dordrecht, The Netherlands: Kluwer.

Karmarkar, N. K. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica, 4*, 373–395.

Khachiyan, L. G. (1979). A polynomial algorithm in linear programming. *Translated in Soviet Mathematics Doklady, 20*, 191–194.

Klee, V. & Minty, G. J. (1972). How good is the simplex algorithm. In O. Shisha (Ed.), *Inequalities III* (pp. 159–175). Academic Press.

Kojima, M., Megiddo, N., Noma, T., & Yoshise, A. (1991). *A unified approach to interior point al − go − rithms for linear complementarity problems* (Lecture notes in computer science, Vol. 538). Berlin, Germany: Springer.

Koltai, T., & Terlaky, T. (2000). The difference between managerial and mathematical interpretation of sensitivity analysis results in linear programming. *International Journal of Production Economics, 65*, 257–274.

Nematollahi, E., & Terlaky, T. (2008). A simpler and tighter redundant Klee-Minty construction. *Optimization Letters, 2*(3), 403–414.

Nesterov, Y. E., & Nemirovskii, A. S. (1994). *Interior point polynomial methods in convex programming: Theory and algorithms*. Philadelphia: SIAM.

Peng, J., Roos, C., & Terlaky, T. (2002). *Self-regularity: A new paradigm for primal-dual interior-point algorithms*. Princeton, NJ: Princeton University Press.

Roos, C., & Terlaky, T. (1997). Advances in linear optimization. In M. DellAmico, F. Maffioli, & S. Martello (Eds.), *Annotated bibliography in combinatorial optimization, Chap − ter 7*. New York: John Wiley & Sons.

Roos, C., Terlaky, G. J., & Vial J. -Ph. (1997). *Interior point methods for linear optimization*. (New York: Springer, 2nd ed., 2006). (Roos, C., Terlaky, T., Vial, J. -Ph. (1997). *Theory and algorithms for linear optimization: An interior point approach*. Chichester, UK: John Wiley & Sons).

Santos, F. (2010). A counterexample to the Hirsch conjecture. arXiv:1006.2814.

Sonnevend, G. y. (1985). An 'analytic center' for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming. In A. Prékopa, J. Szelezsán, & B. Strazicky (Eds.), *System modeling and optimization : Proceedings of the 12th IFIP-Conference held in Budapest, Hungary, September 1985. Lecture notes in control and information sciences* (Vol. 84, pp. 866–876). Berlin, West–Germany: Springer Verlag, 1986.

Terlaky, T. (Ed.). (1996). *Interior point methods in mathematical programming*. Dordrecht, The Netherlands: Kluwer.

Todd, M. (1999). A study of search directions in primal-dual interior-point methods for semidefinite programming. *Optimization Methods and Software, 11*, 1–46.

Vandenberghe, L., & Boyd, S. (1996). Semidefinite programming. *SIAM Review, 38*, 49–95.

Wächter, A. (2002). *An interior point algorithm for large-scale nonlinear optimization with applications in process engineering*. Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA, USA.
Wächter, A., & Biegler, L. T. (2006). On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming, 106*(1), 25–57.
Wolkowicz, H., Saigal, R., Vandenberghe, L. (Eds.) (2000). *Handbook of semidefinite programming: Theory, algorithms, and applications*. Kluwer A.P.C.
Wright, S. J. (1996). *Primal-dual interior-point methods*. Philadelphia: SIAM.
Wright, M. H. (2004). The interior-point revolution in optimization: History, recent developments, and lasting consequences. *Bulletin (New Series) of the American Mathematical Society*, 42(1): 39–56.
Ye, Y. (1997). *Interior point algorithms*. New York: John Wiley & Sons.

International Federation of Operational Research Societies (IFORS)

Graham K. Rand
Lancaster University, Lancaster, UK

IFORS, the International Federation of Operational Research Societies, is an international society whose members are national operational research societies. IFORS was founded in 1959 following the first international OR conference that was held in Oxford, 1957 (Rand 2000).

IFORS was formed by three societies: The U.S. Operations Research Society of America (ORSA), the U.K. Operational Research Society (ORS), and the French Société Française de Recherche Opérationelle (SOFRO). The Statutes of IFORS (Anonymous 1959) set out the purpose of the Federation: "The development of operational research as a unified science and its advancement in all nations of the world." Perhaps the most striking aspect of the Statutes is the provision that in all formal votes taken by the IFORS Board, the voting strength of each member society is in proportion to the square root of its membership. As of this writing, 50 national societies belong to IFORS, with a collective individual membership of about 30,000. See Rand (2001) and del Rosario and Rand (2010) for fuller information about IFORS' history.

IFORS publications includes the proceedings of its triennial conferences from 1957 to 1990; the abstracting journal, *International Abstracts in Operations Research* (*IAOR*), started in 1961 and now published on the Internet as well as in print; and, since 1993, the journal, *International Transactions in Operational Research* (*ITOR*), that publishes selected conference papers, special issues focused on current OR topics, as well as international perspectives of OR.

Discussions at the sixth IFORS Conference in Dublin in 1972 led to the creation, in 1976, of EURO, The Association of European Operational Research Societies within IFORS; there are now 31 member societies in EURO, including South Africa, Egypt and Israel. In 1982, the Association of Latin American OR Societies (ALIO) was established; there are now eight member societies in ALIO, including two members of EURO: Spain and Portugal. The Association of Asian-Pacific OR Societies within IFORS (APORS) came into being in 1985; there are now 10 member societies in APORS. When, in 1987, the IFORS constitution was changed, NORAM, the Association of North American OR Societies within IFORS, composed of the OR Societies in Canada and the USA, was created solely so that a Vice-President would be able to represent North America.

References

Anonymous. (1959). The International Federation of Operational Research Societies. *Operations Research*, *7*, B36–B41.
del Rosario, E. A., & Rand, G. K. (2010). IFORS: 50 at 50. *Boletín de Estadística e Investigación Operativa*, *26*(1), 84–96.
Rand, G. K. (2000). IFORS: The formative years. *International Transactions in Operational Research, 7*, 101–107.
Rand, G. K. (2001). Forty years of IFORS. *International Transactions in Operational Research, 8*, 611–623.

International Institute for Applied Systems Analysis (IIASA)

The International Institute of Applied Systems Analysis (IIASA) is a nongovernmental research institution located in Laxenburg, Austria. IIASA was founded in 1972 on the initiative of the academies of science or equivalent institutions of 12 nations. As of 2012, the following countries are national member organizations:

Austria, Brazil, China, Egypt, Finland, Finland, Germany, India, Japan, Republic of Korea, Malaysia, Netherlands, Norway, Pakistan, Russian Federation, South Africa, Sweden, Ukraine and United States of America. The original motivation for the establishment of IIASA was to enable scientists from East and West to work together on problems of common concern. Although this is still an objective of the Institute, it has been broadened to encompass joint work by scientists from most countries. The goal of IIASA is "To conduct international and interdisciplinary scientific studies to provide timely and relevant information and options, addressing critical issues of global environmental, economic, and social change, for the benefit of the public, the scientific community, and national and international institutions" (IIASA *Agenda for the Third Decade*). Resident scientists at IIASA coordinate research projects, working in collaboration with worldwide networks of researchers, policymakers, and research organizations. IIASA has been instrumental in the development of global (world) models that are concerned with environmental, energy and other resource, economic and population issues.

See

- ▶ Environmental Systems Analysis
- ▶ Global Models

Intervention Model

- ▶ Time Series Analysis

Invariant Distribution

Another name for the stationary distribution. Also called invariant measure.

See

- ▶ Limiting Distribution
- ▶ Stationary Distribution
- ▶ Statistical Equilibrium

Inventory Modeling

Edward A. Silver[1] and David F. Pyke[2]
[1]University of Calgary, Calgary, Alberta, Canada
[2]University of San Diego, San Diego, CA, USA

Introduction

Supply chains have simultaneously become increasingly globalized and lean. As a result, costs have often decreased dramatically, but at the expense of increased complexity and risk. A virus outbreak in Asia can shut down a factory in Cleveland, and a volcano in Iceland can shutter automotive factories in Spain and Germany. These developments suggest that excellent inventory management and control are critical to effective management of supply chains. Managers need to understand the optimal amount of inventory to hold in stable situations, as well as in highly dynamic and uncertain environments. The tendency is often to hold too much inventory, and thus to avoid stockouts and the resulting fallout that lands on inventory managers. Companies, however, are also confronted with the cost of inventory in the form of reduced working capital that could be used for other profitable activities such as new product innovation or paying down debt. How should managers handle this tradeoff? This article provides an introduction to the answer to this question.

Standard inventory models, of the type introduced here, add significant value to many organizations. In a number of cases, however, the basic models must be adjusted to account for the complexities of the situation (Silver 2008). Tiwari and Gavirneni (2007) argue that inventory researchers should seek close connections with companies facing these complexities so that their research will address actual company needs. This is clearly true. Nevertheless, there are excellent examples of successful implementation of challenging inventory models, including:

i) A major appliance manufacturer demonstrates that it could reduce its service parts inventory stocked in service vehicles from $7 million to $3 million (Gorman and Ahire 2006).
ii) A division of John Deere with sales of $3 billion improved on-time shipments from 63% to 92%

while reducing or avoiding inventory costs by $890 million (Troyer et al. 2005).

Inventory decisions can often interact with decisions in other areas of the organization. Examples include: i) preventive maintenance (determination of inventory levels of spare parts); ii) marketing (effects of pricing and promotion on demand, hence stock requirements and, in the opposite direction, more effective inventory management can reduce customer response time, hence stimulating increased demand), iii) quality assurance (higher quality levels reduce the need for buffer or safety stocks), iv) production scheduling (provision of supporting raw materials and supplies), and v) finance (interest rates that vary with the level of assets held by the firm (Buzacott and Zhang 2004). The models that deal with these complexities are not explicitly presented here. Rather, the intention is to provide an introduction to inventory models per se, thus providing the reader with an overview of the general subject area.

In the next section a listing is provided of the generic reasons why organizations carry inventories. This is followed by a discussion of the types of costs that are relevant in the development and use of inventory models. Then, the subsequent four sections deal with illustrative models. These are followed by a general classification scheme. The coverage of the topic area concludes with a discussion of the important possibility of changing some of the parameters or constraints (givens) in inventory models. More detailed treatments of inventory modeling include Graves, Rinnooy Kan and Zipkin (1993), Silver, Pyke, and Peterson (1998), Zipkin (2000), and Axsäter (2010).

Reasons for Carrying Inventories

There are six generic reasons for organizations to carry inventories. Most situations involve a mix of these reasons, but each is discussed separately to emphasize the associated rationale:

i) *Cycle stock* — when the demand pattern is level and known and there is no uncertainty in supply it still may make sense to not have the replenishment inflow precisely match the steady outflow. There may be physical limits on replenishment sizes (e.g., batch container sizes in chemical processes), major fixed costs associated with each replenishment action, or quantity discounts on purchase price and/or transportation costs. Full truck load shipments, for example, are significantly less expensive than less-than-truckload shipments. Each of these reasons leads to the repeated (or cyclic) use of a significant replenishment size.

ii) *Congestion stock* — even when the reasons for holding cycle stock are not present and there is still no uncertainty in supply or demand, it may be necessary to have inventories of items when they are produced on the same piece of equipment and it takes an appreciable amount of time to change over from production of one item to another. One has to produce more than the immediate needs of an item because the congestion on the equipment prevents producing that item again for an appreciable amount of time.

iii) *Buffer or safety stock* — when there is uncertainty in demand and/or supply and the required customer response time is lower than the time necessary to acquire/produce the demanded goods, it is necessary to have extra stock on hand to ensure an adequate level of customer service. Note that the customer can be internal to the organization; for example, spare parts needed to repair equipment.

iv) *Pipeline stock* — if an item has to be moved an appreciable distance before being delivered to the customer, then there will be stock in the pipeline. More generally, if units must go through a process (transportation is a special case) that requires a non-negligible amount of time, then there will be associated pipeline stock equal to the throughput rate multiplied by the process time per unit.

v) *Anticipation stock* — where factors such as demand levels, raw material availability or raw material prices are expected to change appreciably with time it may make sense to build up (and deplete) inventory levels in anticipation of these changes.

vi) *Decoupling stock* — in a multi-echelon situation (or multistage process) stock may be used to permit the separation of decision making at the different levels or echelons. For example, decoupling inventory allows decentralized decision making at branch warehouses without every decision at a branch having an immediate impact on, say, the central warehouse or factory.

I

As discussed below, it may be more appropriate to eliminate the underlying causes or reasons for carrying inventories, rather than simply accepting them within the modeling and inventory control environment.

The Categories of Inventory: Related Costs

The costs associated with inventory management are often not easy to estimate in practice. Moreover, only so-called relevant costs, that is, those that can be influenced by inventory management decisions, should be considered. In particular, care must be taken with respect to overhead costs that are often not affected by inventory decisions., see (Silver et al. 1998).

Five different categories of costs are considered.

a) *Costs of the Material Itself*: This so-called unit variable cost (raw material plus any value added or material handling) is denoted by v, with dimension of dollars/unit. These costs are relevant only insofar as they are affected by the size of the replenishments used. If there are no quantity discounts in acquisition cost (including the transportation component), then over a given time period (such as a year), the costs of the material will be a constant, independent of the replenishment sizes used. Specifically, if the unit variable cost is independent of the replenishment sizes used, then the total cost of the material is Dv per year, where D is the demand (or usage) rate in units/year.

b) *Fixed Cost of Each Replenishment Action*: The fixed cost of a replenishment action is denoted by A, that is, the cost component that is independent of the size of the replenishment. In a production context, A is often referred to as the setup or change-over cost. In a retail or distribution environment, A includes the costs of order forms, postage, authorization, receiving, inspection, and handling of vendor invoices.

c) *The Costs of Having Material in Inventory (Inventory Carrying Costs)*: The common way of modeling the costs of having material in inventory is as

$$\text{Cost/year} = \bar{I}vr \tag{1}$$

where $\bar{I}$ is the *average* inventory, in convenient units of the item under consideration, v is the unit variable cost in \$/unit, and the carrying charge, r, is the cost in dollars of carrying one dollar of inventory for one year. The latter encompasses out-of-pocket expenses (e.g., insurance, taxes, operating the warehouse, etc.) and the lost opportunity of having capital tied up in the stock (e.g., it could be invested elsewhere or used to pay off debt). Some models use the symbol $h = vr$ to represent the cost per unit in inventory per year.

d) *The Costs of Insufficient Stock in the Short Run*: If the stock is inadequate to meet pending demand, then two types of costs may be incurred – those associated with stockouts (lost sales, backorders, loss of goodwill, downtime of equipment, etc.) and those of emergency actions to avoid stockouts (e.g., expediting, use of an emergency high-cost supplier, etc.). There is no universally appropriate way of modeling such costs as a function of the occurrence and magnitude of the shortage. Possibilities include a fixed cost per stockout occasion, a cost proportional to the number of units short, and so on.

In lieu of assigning a cost of insufficient stock, many organizations impose a service constraint on the inventory policy. Again, there are a wide variety of possible service measures. Two of the more common ones are a specified probability of no stockout prior to the receipt of each replenishment, and a specified fraction of the demand to be routinely met from stock. The latter is known as the fill rate.

e) *The Costs of the Inventory Control System*: Many models are concerned with minimizing the total of two or more of the preceding four cost categories. There is a fifth category, however, that should be considered in selecting among different inventory control systems, namely the costs of administering the system itself. These include the costs of acquiring and updating the data (e.g., demand rate, measure of demand variability, cost parameters, etc.) required for the operation of the system and its associated decision rules. They also include the cost of numerical calculations, although these are likely quite low in most instances, and the cost training and other aspects of implementation.

The Economic Order Quantity (Wilson Lot-Size)

The Economic Order Quantity (EOQ) is one of the earliest results developed in inventory modeling

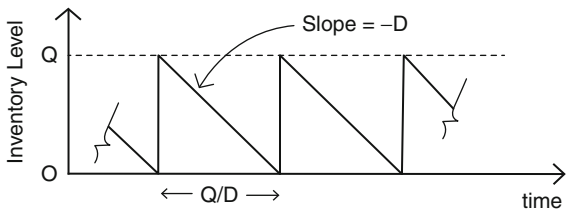
(*Harris* 1913). It addresses the issue of how much inventory to replenish under very stable conditions where there is a significant fixed cost (A) per replenishment. Thus, the economic order quantity (EOQ) is concerned with cycle stock. Strictly speaking, it is based on a number of rather severe assumptions, but it is still an important result for two reasons: (1) the costs tend to be insensitive to some of the assumptions; and (2) many of the assumptions can be relaxed leading to somewhat more complicated results, but the EOQ or an obvious variation thereof often still plays a central role.

Assumptions — There are nine underlying assumptions:

1. the demand rate is constant and known,
2. there are no restrictions on the size of the replenishment quantity (including that it need not be an integer number of units),
3. there are no quantity discounts,
4. the cost factors do not change appreciably with time,
5. each inventory item is treated independently of others (i.e., it is chosen to ignore any possible benefits of coordination),
6. the replenishment lead time (the time interval from when it is decided to place a replenishment order until the moment that the associated material is on the shelf ready to satisfy demand) has a known value,
7. the entire replenishment arrives at the same time (unlike in a production context where there may be a gradual buildup of stock),
8. no shortages are permitted,
9. the planning horizon is very long; that is, the parameters will continue at the current values well into the future.

Derivation of the EOQ — Under the above set of assumptions, there is no uncertainty and no parameters change appreciably with time. Therefore, it is appropriate to restrict attention to a policy of ordering the same quantity Q (in units) over and over again with each replenishment arriving just as the on-hand inventory goes to zero. Thus, each order is placed exactly a lead time before the replenishment arrives. The resulting pattern of inventory versus time is shown in Fig. 1, where the slope is the negative of the demand rate, D, in units/year.

From the set of assumptions it follows that there are only two categories of relevant costs (relevant



Inventory Modeling, Fig. 1 Inventory Level Versus Time

in the sense that they will be affected by the choice of Q), namely the fixed costs of replenishments and the inventory carrying costs. The total relevant costs per year as a function of the order quantity are given by

$$TRC(Q) = \frac{AD}{Q} + \frac{Qvr}{2}. \tag{2}$$

The first term is the product of the fixed cost per replenishment and the number of replenishments per year, while the second term comes from Equation (1) and the fact that $\bar{I}$ for the triangles of Fig. 1 is $Q/2$.

Setting $dTRC(Q)/dQ = 0$ leads to the optimum Q-value as

$$\text{EOQ} = \sqrt{\frac{2AD}{vr}}. \tag{3}$$

Moreover,

$$\frac{d^2TRC(Q) =}{d^2Q} = \frac{2AD}{Q^3} > 0$$

for any $Q > 0$, so that the minimizing value of Q is indeed found.

Remarks

i) At the EOQ value, one can show that the two components of $TRC(Q)$ in Equation (2) are equal. (This is a very special property of the nature of these two components. In general, when one is minimizing the sum of two cost functions of a variable, the two cost functions are not equal at the minimizing value of the controllable variable, but at that point it is known that the slopes of the two component functions are equal and opposite in sign.)

ii) The EOQ expressed as a time supply is

$$\frac{EOQ}{D} = \sqrt{\frac{2A}{Dvr}}. \tag{4}$$

Many organizations have tended to use a very simple decision rule, namely the same time supply replenishment quantity for a broad range of items. For example, they may set Q = six weeks of supply for all items. Equation (4) shows that this is inappropriate in that any of A, D and v are likely to vary between items.

iii) An important relaxation of the EOQ model is to permit quantity discounts.

As an example, the so-called all units discount situation is where the unit variable cost

$$v = \begin{cases} v_0 & \text{if } Q < Q_b \\ v_1 & \text{if } Q \geq Q_b \end{cases}$$

where Q_b is the breakpoint order quantity and $v_1 < v_0$. Under such circumstances, it can be shown that the best order quantity must be at one of three positions: the EOQ using v_0, Q_b, or the EOQ using v_1 (Silver et al. 1998).

iv) Another important extension is where the demand pattern is still known but varies with time. Trended demand would be an example. In addition, with the use of Manufacturing Resources Planning (MRP), the demand pattern may be lumpy. In these situations, it no longer follows that repetitive use of the same Q value is appropriate; hence it is inadequate to look at average costs in a typical year and an exact analysis becomes much more complicated. There is an extensive literature on this so-called lot-sizing problem, see (Silver et al. 1998).

An Illustrative Model for the Case of Congestion Stock

Here, a group of n items (numbered $i = 1, 2, \ldots, n$) satisfying all but two of the EOQ assumptions is considered. Specifically the items are produced on the same piece of equipment (i.e., coordination is necessary) and there is a gradual buildup of the stock of the item being replenished (m_i units/year for item i). Furthermore, it is assumed that a so-called cyclic production schedule is used, that is, item 1, then item 2 are produced,. . ., then item n and returned to item 1 to begin a new cycle. There may be idle time in each cycle, as appropriate. Item i has parameters D_i, A_i, v_i and m_i and Q_i is defined to be its replenishment quantity. Moreover, assume that there is a setup time of τ_i at the beginning of the replenishment of item i. The single decision variable is the duration of each cycle, T, in years. The associated replenishment quantities are given by

$$Q_i = D_i T \quad i = 1,\ 2,\ \ldots,\ n. \tag{5}$$

Production of item i begins just as its inventory level is depleted, that is, the setup must be commenced τ_i before that moment. Production of i continues for $D_i\, T/m_i$ units of time and the inventory reaches a maximum level of $Q_i\,(1 - D_i/m_i)$, not Q_i, because usage at rate D_i continues during the production. Thus the average inventory level of item i is

$$\bar{I}_i = \frac{D_i T}{2}(1 - D_i/m_i) \tag{6}$$

The total relevant costs per year are

$$TRC(T) = \sum_{i=1}^{n} \frac{A_i}{T} + \sum_{i=1}^{n} \frac{D_i T}{2}(1 - D_i/m_i)v_i r. \tag{7}$$

One wishes to minimize this expression but subject to having adequate capacity, namely

$$\sum_{i=1}^{n} \left(\tau_i + \frac{D_i T}{m_i}\right) \leq T$$

or

$$T \geq \frac{\Sigma \tau_i}{1 - \Sigma D_i/m_i}. \tag{8}$$

Again, one can show that $\mathrm{d}^2\, TRC(T)/\mathrm{d}T^{\,2} > 0$, so that setting $\mathrm{d}TRC(T)/\mathrm{d}T = 0$ will give a minimum of $TRC(T)$. This turns out to be where

$$T_{\text{opt}} = \sqrt{\frac{2\Sigma A_i}{r\Sigma D_i v_i (1 - D_i/m_i)}}. \tag{9}$$

Because of the convexity of $TRC(T)$, T_{opt} is used if it satisfies condition (8), otherwise T is set equal to the right-hand side of (8).

A more complicated problem to analyze is where not every item is produced on each cycle. Rather, now let $Q_i = k_i D_i T$ where $k_i = 1, 2, 3, \ldots$, see Chapter 11 in Silver, Pyke and Peterson (1998).

The Newsvendor (or Single-Period) Problem

The Newsvendor problem applies to situations in which the selling season is quite brief relative to the replenishment lead time. Therefore, there is only one opportunity to purchase the item in advance of actual demand. Seasonal goods, holiday items, newspapers, fashion apparel, and even consumer electronics fit in this category. Indeed, in today's world, more and more products share the characteristics that demand a newsvendor solution. Typically, some type of forecasting model is used to forecast demand in the period of interest, and there is an associated probability distribution of forecast errors or equivalently of actual demand that will result. Let the continuous probability distribution of demand x in the period of interest be denoted by $f(x)$ with a cumulative distribution

$$F(x) = \int_0^x f(y)dy. \tag{10}$$

The decision to be made is how large a quantity, Q, of the item to have available to meet demand in the period. Suppose that there is a shortage (or underage) cost of c_u for each unit of demand not satisfied (i.e., when $Q < x$), and an overage cost of c_o for each unit of stock that is not demanded, that is, remaining at the end of the period (when $Q > x$).

A marginal is used, as opposed to a total, cost argument. Specifically consider the Qth unit made available. It will save an underage cost anytime $x \geq Q$. The probability of this event is $1 - F(Q)$. Hence, the expected marginal cost savings of the Qth unit are

$$EMS(Q) = c_u[1 - F(Q)]. \tag{11}$$

The Qth unit will incur an avoidable cost of c_o if demand turns out to be less than Q. Therefore, the expected marginal cost increase of the Qth unit is

$$EMI(Q) = c_o F(Q) \tag{12}$$

It can be argued, that for optimality, one would want to stop with the Q value where

$$EMS(Q) = EMI(Q)$$

or, using Equations (11) and (12), the best Q, denoted by Q^*, must satisfy

$$F(Q^*) = \frac{c_u}{c_u + c_o}. \tag{13}$$

Note that (13) is a general result for any continuous distribution of demand.

Many extensions of this problem have been solved. A multi–item version with a budget constraint on the total amount that can be spent on the set of items can be found in Silver, Pyke and Peterson (1998, pp. 393–396. Pricing and quantity decisions are reviewed and extended in Petruzzi and Dada (1999), and situations with demand substitution are analyzed in Netessine and Rudi (2003). Numerous additional extensions exist.

An Illustration of Dealing with Uncertain Demand in an On-Going Situation

In contrast with the previous section, consider the case where demand continues on indefinitely so that unused material can be kept in stock until it is used up by future demand. Examples include apparel that is not fashion-oriented and non-perishable staple food items. These items typically have a non-zero replenishment lead time. The combination of random demand and a non-zero lead time forces a more careful definition on what is meant by inventory level. In fact, there are at least four different definitions:

i) On-hand stock — material physically present.
ii) Backorders — unsatisfied demand that will be met when stock becomes available.
iii) Net stock = (On-hand) — (Backorders)
iv) Inventory position = (On-hand)

$$+\begin{pmatrix}\text{On-order}\\ \text{from supplier}\end{pmatrix} - \begin{pmatrix}\text{Backordered}\\ \text{customer demands}\end{pmatrix}$$

Reordering decisions are based on this last quantity.

Common Individual Item Control Systems: When demand is uncertain, there are really three decision variables regarding the inventory management of a particular item at a specific location, namely,

i) how often to review the status of the item (continuous review, sometimes called transactions reporting, versus periodic review and, if the latter, what review interval (R) to use),
ii) when to initiate a replenishment, and
iii) how much to replenish.

The three most common individual item control policies are

i) (s, Q) — continuous review ($R = 0$) with an order for a fixed quantity Q being placed when the inventory position drops to the reorder point s or lower,
ii) (R, S) — every R units of time enough is ordered to raise the inventory position to the order–up–to–level S, and
iii) (R, s, S) — every R units of time a review is made. If the inventory position is at s or lower, enough is ordered to raise it to S.

It should be emphasized that here one deals with a so-called independent demand situation, where the demand for the item under consideration is not a function of replenishment decisions for other items. In particular, where an item is a component of another item, its demand is dependent on the demand for the latter item, and the control procedures of MRP may be more appropriate than any of the above control policies. Furthermore, if the item shares a common resource, such as a transportation container or production equipment, it cannot be considered in isolation of the other items that share the resource.

Selecting s in an (s, Q) System: In an (s, Q) system, the order quantity, Q, is often chosen using the EOQ or perhaps a constrained order quantity, such as a truckload. Here it is focused on choosing the reorder point, s. It is illustrated for the case of normally distributed demand during a constant lead time (of duration L) and for a particular service constraint, namely, where there is a specified probability P of no stockout during each lead time. [A variety of other combinations of control policy, demand distribution and service measure/shortage costing method can be treated (Brown 1982; Hax and Candea 1984; Nahmias 2008; Silver et al. 1998; Zipkin 2000; *Axsäter* 2010)].

It is assumed that each replenishment is triggered when the inventory position is exactly at the level of s. (If the inventory position falls below s before a replenishment is triggered, due to large transactions, the amount below the reorder point is called an undershoot, and the mathematics becomes substantially more complex). Letting $f(x)$ be the general probability distribution of the lead time demand x, then the probability of no stockout must satisfy

$$P = \int_{-\infty}^{s} f(x)dx \tag{14}$$

For the special case of $f(x)$ being normally distributed (with mean μ_L and standard deviation σ_L), letting

$$s = \mu_L + k\sigma_L, \tag{15}$$

then a substitution of $u = (x - \mu_L)/\sigma_L$ in Equation (14) leads to

$$P = \Phi(k) \tag{16}$$

where $\Phi(k) = \int_{-\infty}^{k} \phi(u)du$ is the unit normal distribution function and $\phi(u)$ is the probability density function of that distribution.

In summary, the procedure is as follows: The specified value of P gives $\Phi(k)$ from (16). Then using the Excel function NORMSINV($\Phi(k)$), or a table lookup, find the associated k value. Then s is determined from equation (15).

The above analysis was based on a constant lead time L. If the lead time is variable, a more refined analysis is required. For example, if one can assume that L and D are independent random variables, then it can be shown that $E(x) = E(L)E(D)$ and $\sigma_x = \sqrt{E(L)\,\mathrm{var}(D) + [E(D)]^2\,\mathrm{var}(L)}$, where x, with mean $E(x)$ and standard deviation σ_x, is the total demand in a replenishment lead time, in units; L, with mean $E(L)$ and variance var (L), is the length of a lead time (L is the number of unit time periods, that is, just a dimensionless number); and D, with mean $E(D)$ and variance var (D), is the demand, in units,

in a unit time period. In this case, the $E(x)$ and σ_x quantities found here should be used in place of μ_L and σ_L in the above decision rule.

Note that the above choice of s is independent of Q. Other service measures/shortage costing methods lead to a decision rule for s that depends upon Q. Such a dependence was ignored in the derivation of the EOQ. There are optimization procedures for *simultaneously* choosing values of Q and s (Hadley and Whitin 1963; Naddor 1966; Nahmias 2008; Silver et al. 1998).

The Wide Variety of Possible Inventory Models

There are a large number of structural parameters that can take on two or more values in actual inventory systems. In principle, each combination of these parameters leads to a different inventory model. In this section, most of the important parameters are listed (many of the possible combinations have been modeled in the literature but often tailor–made adaptations or approximations are needed to accurately model the inventory problem of a given organization (Silver 1981; Silver 2008; Tiwari and Gavirneni 2007):

Nature of Demand

- deterministic vs. probabilistic (in the latter case, known versus uncertain probability distribution)
- stationary vs. varying with time (e.g., seasonality)
- influenced by on–hand inventory or not
- consumables vs returnables/repairables
- independent of, vs. dependent on, replenishment decisions of other items

Time Horizon

- single period vs. multiperiod
- discrete vs. continuous time
- use of discounting or not

Supply Issues

- quantity discounts (economies of scale)
- minimum order size or fixed batch size
- supply available or not in certain periods
- fixed or random lead time
- orders can cross in time or not
- random yield (acceptable quantity received is not the same as that ordered)
- capacity restrictions
- two or more suppliers used for the same item (i.e., possible order-splitting to help cope with either random lead times or random yield)

Time-Dependent Parameters (Other than Demand)

- inflation
- one-time special prices
- lead time varies with time
- capacity varies with time

What Happens Under a Stockout Situation

- lost sales vs. backorders vs mix of these

Shelf-Life Considerations

- obsolescence
- perishability (deteriorating inventory)

Single vs. Multiple Items

- group budget or space constraint
- coordinated control (or joint replenishment) because of common supplier, mode of transport or production equipment
- substitutable or complementary items

Single vs. Multiechelon

- in multiechelon (multistage) (Schwarz 1981; Sherbrooke 2004), serial vs. convergent (e.g., assembly) vs. divergent (e.g., distribution)

Knowledge of Status of Stock (and Other Parameter Values)

- known exactly or not
- continuously vs. at discrete points in time

Challenging the Underlying Assumptions and Parameter Values

Traditionally, the underlying assumptions and the values of the parameters, discussed earlier, have been accepted as givens in inventory modeling and associated inventory control. There are a number of

innovations that challenge this perspective. First, the philosophy of continuous improvement (an aspect of the philosophy of Just-in-Time, or JIT) argues that parameters such as the setup cost, the replenishment lead time and so on can be changed, often with much more substantial benefits than simply optimizing subject to the given parameter values. Another way of saying this is that it may be better to at least partially eliminate the causes of inventories rather than just choosing the best inventory level (Silver 1992). Second, the Theory of Constraints perspective is that one should identify the key constraint (e.g., a bottleneck operation) that is preventing better performance and concentrate improvement efforts on reducing the impact of this constraint (Goldratt and Cox 1986). Third, the spread of electronic and internet ordering has significantly reduced both the cost of communication and the fixed cost of orders, although substantial costs remain. Fourth, globalization has encouraged companies to consider moving operations across the world, sometimes increasing the replenishment lead time from, say, one week to eight weeks. Fluctuations in oil prices and currency exchange rates have caused companies to reconsider offshoring, moving their sources of supply back to their domestic markets. Finally, other innovations in supply chain management, such as vendor managed inventory (VMI), have improved communications and therefore forecast accuracy. These changes do not necessarily change the appropriate choice of models, but they do demand that managers should update model parameters and re-optimize, or even change models to more accurately capture the new situation.

See

- ▶ Economic Order Quantity Model Extensions
- ▶ Hierarchical Production Planning
- ▶ Just-in-Time (JIT) Manufacturing
- ▶ Logistics and Supply Chain Management
- ▶ Operations Management
- ▶ Production Management
- ▶ Scheduling and Sequencing
- ▶ Supply Chain Management
- ▶ Theory of Constraints

References

Axsäter, S. (2010). *Inventory control* (2nd ed.). Boston: Kluwer.

Brown, R. G. (1982). *Advanced service parts inventory control* (2nd ed.). Norwich, VT: Materials Management Systems.

Buzacott, J. A., & Zhang, R. Q. (2004). Inventory management with asset-based financing. *Management Science, 50*(9), 1274–1290.

Goldratt, E. M., & Cox, J. (1986). *The goal* (Rev. Ed.). Croton-on-Hudson, NY: North River Press.

Gorman, M. F., & Ahire, S. (2006). A major appliance manufacturer rethinks its inventory policies for service vehicles. *Interfaces, 36*(5), 407–419.

Graves, S. C., Rinnooy Kan, A. H. G., & Zipkin, P. H. (Eds.). (1993). *Logistics of production and inventory* (Handbooks in operations research and management science, Vol. 4). Amsterdam: North-Holland.

Hadley, G., & Whitin, T. (1963). *Analysis of inventory systems*. Englewood Cliffs, NJ: Prentice-Hall.

Harris, F. W. (1913). How many parts to make at once. *Factory, the Magazine of Management*, 10(2), 135–6 and 152. (Reprinted in *Operations Research*, 38(6), 947–950).

Hax, A. C., & Candea, D. (1984). *Production and inventory management*. Englewood Cliffs, NJ: Prentice-Hall.

Naddor, E. (1966). *Inventory systems*. New York: John Wiley.

Nahmias, S. (2008). *Production and operations analysis* (3rd ed.). Chicago: Irwin.

Netessine, S., & Rudi, N. (2003). Centralized and competitive inventory models with demand substitution. *Operations Research, 51*(2), 329–335.

Petruzzi, N. C., & Dada, M. (1999). Pricing and the newsvendor problem: A review with extensions. *Operations Research, 47*(2), 183–189.

Schwarz, L. B. (Ed.). (1981). *Multi-level production/inventory control systems: Theory and practice* (Studies in the Management Sciences, Vol. 16). Amsterdam: North-Holland.

Sherbrooke, C. S. (2004). *Optimal inventory modeling of systems multi-echelon techniques*. Boston: Kluwer.

Silver, E. A. (1981). Operations research in inventory management: A review and critique. *Operations Research, 29*, 628–645.

Silver, E. A. (1992). Changing the givens in model-ling inventory problems: The example of just-in-time systems. *International Journal of Production Economics, 26*, 347–351.

Silver, E. A. (2008). Inventory management: An overview, canadian publications, practical applications, and suggestions for future research. *Infor, 46*(1), 15–27.

Silver, E. A., Pyke, D. F., & Peterson, R. (1998). *Inventory management and production planning and scheduling* (3rd ed.). New York: John Wiley.

Tiwari, V., & Gavirneni, S. (2007). ASP, the art and science of practice: Recoupling inventory control research and practice: guidelines for achieving synergy. *Interfaces, 37*(2), 176–186.

Troyer, L., Smith, J., Marshall, S., Yaniv, E., Tayur, S., Barkman, M., et al. (2005). Improving asset management and order fulfillment at Deere & Company's C&CE Division. *Interfaces, 35*(1), 76–87.
Zipkin, P. H. (2000). *Foundations of inventory management*. Boston: McGraw-Hill.

Inverse Matrix

For a square $m \times m$ matrix $\boldsymbol{A}$, the inverse matrix $\boldsymbol{A}^{-1}$ is also an $m \times m$ matrix such that $\boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{I} = \boldsymbol{A}\boldsymbol{A}^{-1}$, where $\boldsymbol{I}$ is the identity matrix. If a matrix has an inverse, then its inverse is unique and the matrix is said to be nonsingular. If an inverse does not exist, the matrix is said to be singular. A nonsingular matrix has a nonzero value for its determinant; a singular matrix has a determinant value equal to zero.

See

► Matrices and Matrix Algebra

Inverse Transform Method

In stochastic or Monte Carlo simulation, a method for sampling from a given probability distribution by using random numbers transformed by the inverse of the cumulative distribution function.

See

► Monte Carlo Simulation
► Random Number Generators
► Random Variates
► Simulation of Stochastic Discrete-Event Systems

IP

Integer Programming.

See

► Integer and Combinatorial Optimization

IPA

Infinitesimal Perturbation Analysis.

See

► Perturbation Analysis

IS

Information systems.

See

► Information Systems and Database Design in OR/MS

Isomorphic Graph

Graphs that have identical structure.

ISOP 9000 Standard

► Quality Control

Isoquant

For a function $f(x)$, the graph or contour $f(x) = C$, where C is a constant, is called an isoquant. If $f(x)$ is a profit (cost) function, then the isoquant is termed an isoprofit (isocost) line.

Iteration

The cycle of steps of an algorithm is called an iteration. For example, in the simplex algorithm for solving linear-programming problems, one iteration is given concisely by the steps: (1) select a nonbasic variable to replace a basic variable, (2) determine the inverse of the new feasible basis, and (3) determine if the new basic feasible solution is optimal.

ITS

► Intelligent Transportation Systems

IVHS

Intelligent vehicle-highway system.

See

► Traffic Analysis

J

Jackson Network

A collection of multi-server queueing systems or nodes with exponential service and Markovian or memoryless probabilistic routing of departures from one node to the others. If there are customers arriving from outside the network to individual nodes in Poisson streams, the network is said to open; otherwise it is closed. All customers who arrive from outside to an open network must eventually leave after receiving service at one or more systems within the network.

See

► Networks of Queues
► Queueing Theory

JIT

► Just-in-Time (JIT) Manufacturing

Job Shop Scheduling

Luis C. Rabelo[1] and Albert Jones[2]
[1]University of Central Florida, Orlando, FL, USA
[2]National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA

Introduction

In the United States there are approximately 62,000 factories producing metal fabricated parts. These parts end up in a wide variety of products sold here and abroad. These factories employ roughly 1.5 million people and ship close to $247 billion worth of products every year. The vast majority of these factories are what are called job shops, meaning that the flow of raw and unfinished goods through them is completely random. Over the years, the behavior and performance of these job shops have been the focus of considerable attention in the operations research (OR) literature. Research papers on topics such as factory layout, inventory control, process control, production scheduling, and resource utilization can be found in almost every issue of every OR journal. The most popular of these topics is production (often referred to as job shop) scheduling. Job shop scheduling can be thought of as the allocation of resources over a specified time to perform a predetermined collection of tasks. Job shop scheduling has received this large amount of attention, because it has the potential to dramatically decrease costs and increase throughput, thereby, profits.

A large number of approaches to the modeling and solution of these job shop scheduling problems have been reported in the OR literature, with varying degrees of success. These approaches revolve around a series of technological advances that have occurred mainly since the 1960s. These include mathematical programming, dispatching rules, expert systems, neural networks, support vector machines, agents, genetic algorithms, particle warm optimization, and inductive learning. In this article, an evolutionary view is taken in describing how these technologies have been applied to job shop scheduling problems. To do this, a few of the most important contributions in each of these technology areas and trends are discussed.

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Mathematical Techniques

Mathematical programming has been applied extensively to job shop scheduling problems. Problems have been formulated using integer programming (Balas 1965, 1967), mixed-integer programming (Balas 1969, 1970), and dynamic programming (Srinivasan 1971). Until recently, the use of these approaches has been limited because scheduling problems belong to the class of *NP*-complete problems. To overcome these deficiencies, a group of researchers began to decompose the scheduling problem into a number of subproblems, proposing a number of techniques to solve them. In addition, new solution techniques, more powerful heuristics, and the computational power of modern computers have enabled these approaches to be used on larger problems. Still, difficulties in the formulation of material flow constraints as mathematical inequalities and the development of generalized software solutions have limited the use of these approaches.

Decomposition strategies—Davis and Jones (1988) proposed a methodology based on the decomposition of mathematical programming problems that used both Benders-type (Benders 1960) and Dantzig/Wolfe-type (Dantzig and Wolfe 1960) decompositions. The methodology was part of closed-loop, real-time, two-level hierarchical shop floor control system. The top-level scheduler (i.e., the supremal) specified the earliest start time and the latest finish time for each job. The lower level scheduling modules (i.e., the infimals) would refine these limit times for each job by detailed sequencing of all operations. A multicriteria objective function was specified that included tardiness, throughput, and process utilization costs. The decomposition was achieved by first reordering the constraints of the original problem to generate a block angular form, then transforming that block angular form into a hierarchical tree structure. In general, *N* subproblems would result plus a constraint set that contained partial members of each of the subproblems. The latter are termed coupling constraints, and included precedence relations and material handling. The supremal unit explicitly considered the coupling constraints, while the infimal units considered their individual decoupled constraint sets. The authors pointed out that the inherent stochastic nature of job shops and the presence of multiple, but often conflicting, objectives made it difficult to express the coupling constraints using exact mathematical relationships. This made it almost impossible to develop a general solution methodology. To overcome this, a real-time simulation methodology was proposed by Davis and Jones (1988) to solve the supremal and infimal problems.

Gershwin (1989) used the notion of temporal decomposition to propose a mathematical programming framework for analysis of production planning and scheduling. This framework can be characterized as hierarchical and multi-layer. The problem formulations to control events at higher layers ignored the details of the variations of events occurring at lower layers. The problem formulations at the lower layers view the events at the higher layers as static, discrete events. Scheduling is actually carried out in bottom three layers so that the production requirements imposed by the planning layers can be met. First, a hedging point is found by solving a dynamic programming problem. This hedging point is the number of excess goods that should be produced to compensate for future equipment failures. This hedging point is used to formulate a linear-programming problem to determine instantaneous production rates. These rates are then used to determine the actual schedule (which parts to make and when). Other approaches have been proposed for generating schedules.

Enumerative techniques and Lagrangian relaxation—Two popular solution techniques for integer-programming problems are branch-and-bound and Lagrangian relaxation. Branch-and-bound is an enumerative technique (Agin 1966; Lawler and Wood 1966). A summary of branching is as follows, (Morton and Pentico 1993): "The basic idea of branching is to conceptualize the problem as a decision tree. Each decision choice point–a node-corresponds to a partial solution. From each node, there grow a number of new branches, one for each possible decision. This branching process continues until leaf nodes, that cannot branch any further, are reached. These leaf nodes are solutions to the scheduling problem."

Although efficient bounding and pruning procedures have been developed to speed up the search, this is still a very computational intensive procedure for solving large scheduling problems. If the integer constraint is the main problem, then why not remove that constraint? A technique called

Lagrangian relaxation, does just that (Shapiro 1979). Lagrangian relaxation solves integer-programming problems by omitting specific integer-valued constraints and adding the corresponding costs (due to these omissions and/or relaxations) to the objective function. As with branch and bound, Lagrangian relaxation is computationally expensive for large scheduling problems.

Model-Based Optimization—Model-Based Optimization (MBO) is an optimization approach that uses mathematical expressions (e.g., constraints and inequalities) to model scheduling problems as mixed integer (non) linear programs (MINLP), (Zentner et al. 1994). A set of methods such as linear programming, branch-and-bound, and decomposition techniques are used to search the scenario space of solutions. Due to the advances in computer technologies, the computation times are becoming very practical. These approaches are being enhanced by the development of English-like scheduling and high-level graphical interfaces. The scheduling languages support the developing of the mathematical formulations with minimum intervention from the user.

Dispatching rules—Dispatching rules have been applied consistently to scheduling problems. They are procedures designed to provide good solutions to complex problems in real-time. The term dispatching rule, scheduling rule, sequencing rule, or heuristic are often used synonymously (Panwalker and Iskander 1977; Blackstone et al. 1982; Baker 1974). Dispatching rules have been classified mainly according to the performance criteria for which they have been developed. Wu (1987) categorized dispatching rules into several classes. Class 1 contains simple priority rules, which are based on information related to the jobs. Sub-classes are based on the particular piece of information used. Example classes include those based on processing times (such as shortest processing time, SPT), due dates (such as earliest due date, EDD), slack (such as minimum slack, MINSLACK), and arrival times (such as first-in first-out, FIFO). Class 2 consists of combinations of rules from class one. The particular rule that is implemented can now depend on the situation that exists on the shop floor. A typical example of a rule in this class is, for example, SPT until the queue length exceeds 5, then switch to FIFO. This prohibits jobs with large processing times from staying in the queue for long periods. Class 3 contains rules that are commonly referred to as Weight Priority Indexes. The idea here is to use more than one piece of information about the jobs to determine the schedule. Pieces of information are assigned weights to reflect their relative importance. Usually, an objective function $f(x)$ is defined. For example,

$$f(x) = \mathbf{weight}_1 * \mathbf{Processing\ Time\ of\ Job}\,(x) + \mathbf{weight}_2 * (\mathbf{Current\ Time} - \mathbf{Due\ Date\ of\ Job}(x)).$$

Then, any time a new sequence is needed, the function f(x) is evaluated for each job x in the queue. The jobs are ranked based on this evaluation.

The performance of a large number of these rules has been studied extensively using simulation techniques (Montazer and Van Wassenhove 1990). These studies have been aimed at answering the question: If you want to optimize a particular performance criterion, which rule should you choose? Most of the early work concentrated on the shortest processing time rule (SPT). Conway and Maxwell (1967) were the first to study the SPT rule and its variations. They found that, although some individual jobs could experience prohibitively long flow times, the SPT rule minimized the mean flow time for all jobs. They also showed that SPT was the best choice for optimizing the mean value of other basic measures such as waiting time and system utilization. Many similar investigations have been carried out to determine the dispatching rule which optimizes a wide range of job-related (such as due date and tardiness) and shop-related (such as throughput and utilization) performance measures. This problem of selecting the best dispatching rule for a given performance measure has been a very active area of research. The research, however, has been expanded to include the possibility of switching rules to address an important problem: error recovery. Two early efforts to address error recovery were conducted by Bean and Birge (1986) and Saleh (1988). Both developed heuristic rules to smooth-out disruptions to the original schedule, thereby creating a match-up with that schedule. Bean and Birge (1986) based their heuristic on Turnpike Theory (McKenzie 1976) to optimize a generalized cost function. Saleh showed how to minimize duration of the disruption by switching the objective function from mean flow time to makespan based on disjunctive graphs (Adams et al. 1988).

Composite dispatching rules are found to be performing much better than simple dispatching rules (Binh and Cing 2005). In addition, their performance depends on the state of the system. Data mining has been used as a to find these composite dispatching rules. For instance, Shahzad and Mebarki (2008) have provide a methodology using data mining to identify a rule-set by exploring the patterns in the solution set obtained by an optimization module based on Tabu Search, a very efficient meta-heuristic. The rule-set approximates the output of the optimization module when incorporated in a simulation model of the system. The C5.0 algorithm (Quinlan 1992) is used as a data mining algorithm for the induction of rule-set.

Artificial Intelligence (AI) Techniques

Starting in the early 1980s, a series of new technologies were applied to job shop scheduling problems. They fall under the general title of artificial intelligence (AI) techniques and include expert systems, knowledge-based systems, and several search techniques. Expert and knowledge-based systems were quite prevalent in the early and mid-1980s. They have four main advantages. First, and perhaps most important, they use both quantitative and qualitative knowledge in the decision-making process. Second, they are capable of generating heuristics that are significantly more complex than the simple dispatching rules described above. Third, the selection of the best heuristic can be based on information about the entire job shop including the current jobs, expected new jobs, and the current status of resources, material transporters, inventory, and personnel. Fourth, they capture complex relationships in elegant new data structures and contain special techniques for powerful manipulation of the information in these data structures. There are, however, serious disadvantages. They can be time consuming to build and verify, as well as difficult to maintain and change. Moreover, since they generate only feasible solutions, it is rarely possible to tell how close that solution is to the optimal solution. Finally, since they are tied directly to the system they were built to manage, there is no such thing as a generic AI system.

Expert/knowledge-based systems—Expert and knowledge-based systems consist of two parts: a knowledge base and inference engine to operate on that knowledge base. Formalizations of the knowledge that human experts use — rules, procedures, heuristics, and other types of abstractions — are captured in the knowledge base. Three types of knowledge are usually included: procedural, declarative, and meta. Procedural knowledge is domain specific problem solving knowledge. Declarative knowledge provides the input data defining the problem domain. Meta knowledge is knowledge about how to use the procedural and declarative knowledge to actually solve the problem. Several data structures have been utilized to represent the knowledge in the knowledge base including semantic nets, frames, scripts, predicate calculus, and production rules. The inference engine selects a strategy to apply to the knowledge bases to solve the problem at hand. It can be forward chaining (data driven) or backward chaining (goal driven).

ISIS (Fox 1983) was the first major expert system aimed specifically at job shop scheduling problems. ISIS used a constraint-directed reasoning approach with three constraint categories: organizational goals, physical limitations, and causal restrictions. Organizational goals considered objective functions based on due-date and work-in-progress. Physical limitations referred to situations where a resource had limited processing capability. Procedural constraints and resource requirements were typical examples of the third category. Several issues with respect to constraints were considered, such as constraints in conflict, importance of a constraint, interactions of constraints, constraint generation and constraint obligation. ISIS used a three level, hierarchical, constraint-directed search. Orders were selected at level 1. A capacity analysis was performed at level 2 to determine the availability of the resources required by the order. Detailed scheduling was performed at level 3. ISIS also provided for the capability to interactively construct and alter schedules. In this capacity, ISIS utilized its constraint knowledge to maintain the consistency of the schedule and to identify scheduling decisions that would result in poorly satisfied constraints. Other examples of expert/knowledge-based scheduling systems developed are MPECS (Multi-Pass Expert Control System, Wysk et al. 1986) and OPIS (Opportunistic Intelligent Scheduler, Smith 1995).

Distributed AI: *Agents*—Due to the limited knowledge and the problem solving ability of a single expert or knowledge based system, these AI

approaches have difficulty solving large scheduling problems, as well. To address this, AI researchers have developed distributed scheduling system approaches (Parunak et al. 1985). They have done this by an application of the well-known divide and conquer approach. This requires a problem decomposition technique, such as those described above, and the development of different expert/knowledge-based systems that can cooperate to solve the overall problem (Zhang and Zhang 1995). The AI community's answer is the agent paradigm. An agent is a unique software process operating asynchronously with other agents. Agents are complete knowledge-based systems by themselves. The set of agents in a system may be heterogeneous with respect to long-term knowledge, solution/evaluation criteria, or goals, as well as languages, algorithms, hardware requirements. Integrating agents selected from a library creates a multi-agent system.

For example, one such multi-agent system could involve two types of agents: tasks and resources. Each task agent might be responsible for scheduling a certain class of tasks such as material handling, machining, or inspection on those resources capable of performing those tasks. This can be done using any performance measure related to tasks, such as minimize tardiness, and any solution technique. Each resource agent might be responsible for a single resource or a class of resources. Task agents must send their resource requests to the appropriate resource agent, along with the set of operations to be performed by that resource (Daouas et al. 1995). Upon receipt of such a request, the resource agent must generate a new schedule using its own performance measures, such as maximize utilization, which includes this request. The resource agent will use the results to decide whether to accept this new request or not. To avoid the situation, where no resource will accept a request, coordination mechanisms need to be developed. No general guidelines for the design and implementation of this coordination are available. Thus, the debates (pros and cons) about centralized vs. decentralized approaches to job shop scheduling continue. The agents formalism may provide an answer to these debates.

Artificial neural networks — Neural networks, also called connectionist or distributed parallel processing models, have been studied for many years in an attempt to mirror the learning and prediction abilities of human beings. Neural network models are distinguished by network topology, node characteristics, and training or learning rules. An example of a three-layer, feed-forward neural network is shown in Fig. 1.

Supervised learning neural networks—Through exposure to historical data, supervised learning neural networks attempt to capture the desired relationships between inputs and the outputs. Back-propagation is the most popular and widely used supervised training procedure. Back-propagation (Rumelhart et al. 1986; Werbos 1995) applies the gradient-descent technique in the feed-forward network to change a collection of weights so that some cost function can be minimized. The cost function, which is only dependent on weights ($\boldsymbol{W}$) and training patterns, is defined by:

$$C(W) = \frac{1}{2}\sum_{ij}(T_{ij} - O_{ij}) \tag{1}$$

where $\boldsymbol{T}$ is the target value, $\boldsymbol{O}$ is the output of the network, i is an output node, and j is the training pattern.

After the network propagates the input values to the output layer, the error between the desired output and actual output will be back-propagated to the previous layer. In the hidden layers, the error for each node is computed by the weighted sum of errors in the next layer's nodes. In a three-layered network, the next layer means the output layer. The activation function is usually a sigmoid function with the weights modified according to

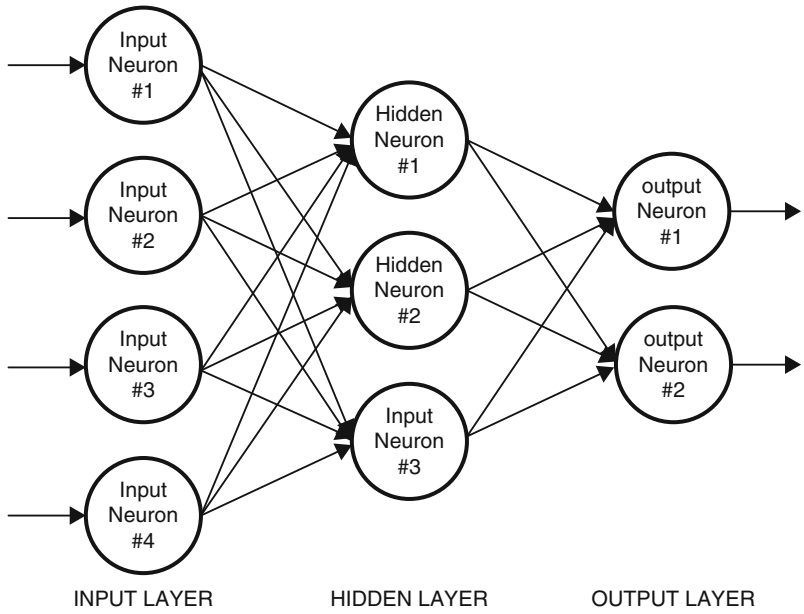
$$\Delta W_{ij} = \eta X_j(1 - X_j)(T_j - X_j)X_i \tag{2}$$

or

$$\Delta W_{ij} = \eta X_j(1 - X_j)\left(\sum_k \delta_\kappa\ W_{jk}\right)X_i \tag{3}$$

where W_{ij} is weight from node i to node j (e.g., neuron), η is the learning rate, X_j is the output of node j, T_j is the target value of node j, and δ_k is the error function of node k.

If j is in the output layer, Equation (2) is used; if j is the hidden layers, (3) is used. The weights are updated to reduce the cost function at each step. The Job shop scheduling temporal reinforcement learning in process continues until the error between



Job Shop Scheduling, Fig. 1 An example of a three-layer, feed-forward neural network

the predicted and the actual outputs is smaller than some predetermined tolerance.

Rabelo (1990) was the first to use back propagation neural nets to solve temporal reinforcement learning, in job shop scheduling problems with several job types, exhibiting different arrival patterns, process plans, precedence sequences and batch sizes. Training examples were generated to train the neural network to select the correct characterization of the manufacturing environments suitable for various scheduling policies and the chosen performance criteria. In order to generate training samples, a performance simulation of the dispatching rules available for the manufacturing system was carried out. The neural networks were trained for problems involving 3, 4, 5, 8, 10, and 20 machines. To carry out this training, a special, input-feature space was developed. This space contained both job characteristics (such as types, number of jobs in each type, routings, due dates, and processing times) and shop characteristics (such as number of machines and their capacities). The output of the neural network represented the relative ranking of the available dispatching rules for that specific scheduling problem and the selected performance criteria. The neural networks were tested in numerous problems and their performance (in terms of minimizing Mean Tardiness) was always better than each single dispatching rule (25% to 50%).

Relaxation models — Neural networks based on job shop scheduling relaxation models in are defined by energy functions. They are preassembled systems that relax from input to output along a predefined energy contour. Hopfield neural networks (Hopfield and Tank 1985) are a classical example of a relaxation model that has been used to solve some classic, textbook scheduling problems (Foo and Takefuji 1988). Two-dimensional Hopfield networks were used to solve 4-job, 3-machine problems and 10-job, 10-machine problems (*Zhou et al.* 1990). They were extended in (Lo and Bavarian 1991) to 3 dimensions to represent jobs ($i = 1, \ldots, I$), machines $j = 1, \ldots, J$), and time ($m = 1, \ldots, M$). In each case, the objective was to minimize the makespan, total time to complete all jobs, which is defined as

$$E = \frac{1}{2} \sum_{j=1} \sum_{i=l} \sum_{m=l} (v_{ijm})(m + T_{ij} - 1) \qquad (4)$$

where v_{ijm} is the output (1 or 0) of neuron ijm, and T_{ij} is the time required by the jth resource (e.g., machine) to complete the ith job.

Due to a large number of variables involved in generating a feasible schedule, these approaches tend to be computationally inefficient and frequently generate infeasible solutions. Consequently, they have not been used to solve realistic scheduling problems.

Temporal reinforcement learning — It was noted above that supervised learning neural networks attempt to capture the desired relationships between inputs and the outputs through exposure to training patterns. However, for some problems, the desired response may not always be available during the time of learning. When, the desired response is obtained, changes to the neural network are performed by assessing penalties for the scheduling actions previously decided by the neural network. As summarized by Tesauro (1992), "In the simplest form of this paradigm, the learning system passively observes a temporal sequence of input states that eventually leads to a final reinforcement or reward signal (usually a scalar). The learning system's task in this case is to predict expected reward given an observation of an input state or sequence of input states. The system may also be set up so that it can generate control signals that influence the sequence of states."

For scheduling, the learning task is to produce a scheduling action that will lead to minimizing (or maximizing) the performance measure (e.g., makespan, tardiness) based on the state of the system (e.g., inventories, machine status, routings, due dates, layouts). Several procedures have been developed to train neural networks when the desired response is not available during the time of learning. Rabelo et al. (1994) utilized a procedure developed by Watkins (1989), denominated Q-learning, to implement a scheduling system to solve dynamic job shop scheduling problems. The scheduling system was able to follow trends in the shop floor and select a dispatching rule that provided the maximum reward according to performance measures based on tardiness and flow time.

Support Vector Machines – Support Vector Machines (SVMs) are algorithms in machine learning based on advances in statistical learning theory. Advances in statistical learning theory, Vapnik-Chervonenkis (VC) theory, (Vapnik 1994; Vapnik 1995) explain that it is critical to constrain the class of functions that the learning machine can generate to one with a capacity that is appropriate for the available training data. Burges (1998) states, "There is a remarkable family of bounds governing the relation between the capacity of a learning machine and its performance. The theory grew out of considerations of under what circumstances, and how quickly, the mean of some empirical quantity converges uniformly, as the number of data points increases, to the true mean (that which would be calculated from an infinite amount of data)." Therefore, to design efficient learning algorithms, a class of functions whose capacity can be computed is essential. SVMs are based on the class of hyperplanes

$$\langle \boldsymbol{w} \cdot \boldsymbol{x} \rangle + \boldsymbol{b} = 0$$

where $\boldsymbol{w}$ are the weights, $\boldsymbol{x}$ is an N-dimensional input vector, and $\boldsymbol{b}$ is a numeric parameter. This class of hyperplanes corresponds to decision functions of the

$$f_{\omega,b}(\boldsymbol{x}) = \text{sign}(\langle \boldsymbol{w} \cdot \boldsymbol{x} \rangle + b)$$

The maximum margin hyperplane is defined as the one with the maximal margin of separation between the classes has the lowest capacity. The instances that are closest to the maximum margin hyperplane are called support vectors (SVs). Points that are not SVs have no influence (Burges 1998) and they may be eliminated without affecting the decision function. SVMs has been applied for the dynamic dispatching rule selection classifier (Shiuea 2009). The proposed SVM classifier using the data-mining-based approach yields a better system performance than heuristic individual dispatching rules under various performance criteria over a long period.

Neighborhood search methods—Neighborhood search methods are very popular. Neighborhood search methods provide good solutions and offer possibilities to be enhanced when combined with other heuristics. Wilkerson and Irwin (1971) developed one of the first neighborhood procedures. This method iteratively added small changes (perturbations) to an initial schedule, that is obtained by any heuristic. Conceptually similar to hill climbing, these techniques continue to perturb and evaluate schedules until there is no improvement in the objective function. When this happens, the procedure is ended. Popular techniques that belong to this family include tabu search, simulated annealing, and genetic algorithms. Each of these has its own perturbation methods, stopping rules, and methods for avoiding local optimum.

Tabu search—The basic idea of Tabu search (Glover 1989, 1990) is to explore the search space of all feasible scheduling solutions by a sequence of

moves. A move from one schedule to another schedule is made by evaluating all candidates and choosing the best available, just like gradient-based techniques. Some moves are classified as tabu, i.e., they are forbidden, because they either trap the search at a local optimum, or they lead to cycling (repeating part of the search). These moves are put onto the tabu list, which is built up from the history of moves used during the search. These tabu moves force exploration of the search space until the old solution area (e.g., local optimum) is left behind. Another key element is that of freeing the search by a short term memory function that provides strategic forgetting. Tabu search methods have been evolving to more advanced frameworks that includes longer term memory mechanisms. These advanced frameworks are sometimes referred as Adaptive Memory Programming (AMP), (Glover 1996).

Tabu search methods have been applied successfully to scheduling problems and as solvers of mixed integer-programming problems. Glover (1996) showed some specialized implementations of tabu search methods for job shop and flow shop scheduling problems, including a number of important cases where tabu search methods are superior to other approaches such as simulated annealing, genetic algorithms, and neural networks.

Simulated annealing — Simulated annealing is based on the analogy to the physical process of cooling and recrystallization of metals. The current state of the thermodynamic system is analogous to the current scheduling solution, the energy equation for the thermodynamic system is analogous to the objective function, and the ground state is analogous to the global optimum. In addition to the global energy J, there is a global temperature T, which is lowered as the iterations progress. Using this analogy, the technique randomly generates new schedules by sampling the probability distribution of the system (Kirkpatrick et al. 1983),

$$p_j \mu \ \exp[-T(\Delta J_{\text{best}} - \Delta J_j)/K], \tag{5}$$

where p_j represents the probability of making move j from among the neighborhood choices, ΔJ_{best} represents the improvement of the objective function for the best choice, and ΔJ_j represents the improvement for choice j, while K is a normalization factor. Since increases of energy can be accepted, the algorithm is able to escape local minima.

Simulated annealing has been applied effectively to job shop scheduling problems. Vakharia and Chang (1990) developed a scheduling system based on simulated annealing for manufacturing cells. Jeffcoat and Bulfin (1993) applied simulated annealing to a resource-constrained scheduling problem. Their computational results indicated that the simulated annealing procedure provided the best results in comparison with other neighborhood search procedures.

Variable Neighborhood Search – Variable Neighborhood Search (VNS) has shown an excellent capability to solve scheduling problems to optimal or near-optimal schedules. VNS can be categorized as a local search-based algorithm, armed with systematic neighborhood search structures. Roshanaei et al. (2009) introduced implementations of VNS that improve the notorious myopic behavior of local search-based metaheuristic algorithms by the means of several systematic insertion neighborhood search structures. Roshanaei et al. (2009) uses the Taillard's benchmark to evaluate the efficiency and effectiveness of their proposed VNS approach against some effective algorithms. The obtained results strongly support the high performance of VNS with respect to other well-known heuristic and metaheuristic algorithms.

Genetic algorithms — Genetic algorithms (GAs) are an optimization methodology based on a direct analogy to Darwinian natural selection and mutations in biological reproduction. In principle, genetic algorithms encode a parallel search through concept space, with each process attempting coarse-grain hill climbing (Goldberg 1988). Instances of a concept correspond to individuals of a species. Induced changes and recombinations of these concepts are tested against an evaluation function to see which ones will survive to the next generation.

Starkweather et al. (1993) were the first to use genetic algorithms to solve a dual-criterion job shop scheduling problem in a real production facility. The criteria were the minimization of average inventory in the plant and the minimization of the average waiting time for an order to be selected. These criteria are negatively correlated (the larger the inventory, the shorter the wait; the smaller the inventory, the longer the wait). To represent the production/shipping

optimization problem, a symbolic coding was used for each member (chromosome) of the population. In this scheme, customer orders are represented by discrete integers. Therefore, each member of the population is a permutation of customer orders. The GA used to solve this problem was based on blind recombinant operators. This recombination operator emphasizes information about the relative order of the elements in the permutation, because this impacts both inventory and waiting time. A single evaluation function (a weighted sum of the two criteria) was utilized to rank each member of the population. That ranking was based on an on-line simulation of the plant operations. This approach generated schedules that produced inventory levels and waiting times that were acceptable to the plant manager. In addition, the integration of the genetic algorithm with the on-line simulation made it possible to react to system dynamics.

The utilization of GA has become very dominant in job shop scheduling. As noted by Fan and Zhang (2010), 22% of the journal articles from 2000 to 2009 in job shop scheduling proposed the utilization of GAs. Another dimension of this use of GAs is the combination of GAs with other methodologies to improve performance. For instance, Thamilselvan and Balasubramanie (2009) produced very good results with GAs combined with TS to create a combinational heuristic denominated Genetic Tabu Search Algorithm (GTA). In general, GTA follows the traditional GA. for the selection process, however, GTA uses tabu search. In several problems with combinations of 5 jobs and 5 machines, GTA outperformed GAs and TSs. Even chaos theory has been used to help GAs to find optimal solutions. For example, Zhou et al. (2009) utilized chaos theory to support the development of a mechanism to avoid local minima.

Fuzzy logic — Fuzzy set theory has been utilized to develop hybrid scheduling approaches. Fuzzy set theory can be useful in modeling and solving job shop scheduling problems with uncertain processing times, constraints, and set-up times. These uncertainties can be represented by fuzzy numbers that are described by using the concept of an interval of confidence. These approaches are usually integrated with other methodologies (e.g., search procedures, constraint relaxation). For example, Slany (1994) stressed the imprecision of straightforward methods presented in the mathematical approaches and introduces a method known as fuzzy constraint relaxation, that is integrated with a knowledge-based scheduling system. This system was applied to a steel manufacturing plant. Grabot and Geneste (1994) used fuzzy logic principles to combine dispatching rules for multi-criteria problems. On the other hand, Krucky (1994) addressed the problem of minimizing setup times of a medium-to-high product mix production line using fuzzy logic. The heuristic, fuzzy logic based algorithm helps to determine how to minimize setup time by clustering assemblies into families of products that share the same setup by balancing a product's placement time between multiple-high-speed placement process steps. Tsujimura et al. (1993) presented a hybrid system that uses fuzzy set theory to model the processing times of a flow shop scheduling facility. Triangular Fuzzy Numbers (TFNs) are used to represent these processing times. Each job is defined by two TFNs, a lower bound and an upper bound. A branch and bound procedure is utilized to minimize makespan.

Particle Swarm Optimization–Particle Swarm Optimization (PSO) was invented in the mid-1990s by Kennedy and Eberhart (1995) as an alternative to genetic algorithms for solving job shop scheduling problems. PSO is based on a social simulation of the movement of flocks of birds. PSO performs a population-based search to optimize the objective function. The population is composed by a swarm of particles that represent potential solutions to the problem. These particles, that are a metaphor of birds in flocks, fly through the search space updating their positions and velocities based on the best experience of their own and the swarm. The swarm moves in the direction of "the region with the higher objective function value, and eventually all particles will gather around the point with the highest objective value" (Jones 2005). There are many application of PSO to job shop scheduling. One of the most prominent ones is the approach presented by Yen and Ivers (2009) that uses space division techniques.

Reactive Scheduling — Reactive scheduling is generally defined as the ability to revise or repair a complete schedule that has been overtaken by events on the shop floor (Zweben et al. 1995). Such events include rush orders, excessive delays, and broken resources. There are two approaches: reactive repair and the proactive adjustment. In reactive repair,

the scheduling system waits until an event has occurred before it attempts to recover from that event. The match-up techniques described earlier fall into this category. Proactive adjustment requires a capability to monitor the system continuously, predict the future evolution of the system, do contingency planning for likely events, and generate new schedules, all during the execution time of the current schedule. The work of Wysk et al. (1986) and Davis and Jones (1988) fall into this category. Other approaches utilize artificial intelligence and knowledge-based methodologies (Smith 1995). Most AI approaches propose a quasi-deterministic view of the system, that is, a stochastic system featuring implicit and/or explicit causal rules. The problem formulation used does not recognize the physical environment of the shop floor domain where interference not only leads to readjustment of schedules but also imposes physical actions to minimize them.

Learning in Scheduling — The first step in developing a knowledge base is knowledge acquisition. This in itself is a two-step process: get the knowledge from knowledge sources and store that knowledge in digital form. To extract knowledge from these two sources, the machine learning technique that learns from examples (data) becomes a promising tool. Inductive learning is a state classification process. If the state space is viewed as a hyperplane, the training data (consisting of conditions and decisions) can be represented as points on the hyperplane. The inductive learning algorithm seeks to draw lines on the hyperplane based on the training data to divide the plane into several areas within which the same decision (conclusion) will be made.

An algorithm that has been implemented in inductive aids and expert system shells is that developed by *Quinlan* (1986), called Iterative Dichotomister 3 or ID3. ID3 uses examples to induce production rules (e.g. IF ... THEN ...), which form a simple decision tree. Decision trees are one way to represent knowledge for the purpose of classification. The nodes in a decision tree correspond to attributes of the objects to be classified, and the arcs are alternative values for these attributes. The end nodes of the tree (leaves) indicate classes to which groups of objects belong. Each example is described by attributes and a resulting decision. To determine a good attribute to partition the objects into classes, entropy is employed to measure the information content of each attribute, and then rules are derived through a repetitive decomposition process that minimizes the overall entropy. The entropy value of attribute A_k can be defined as

$$H(A_k) = \sum_{j=l}^{M_k} P(a_{kj}) \left\{ - \sum_{i=1}^{N} P(c_i|a_{kj}) \log_2 P(c_i|a_{kj}) \right\} \tag{6}$$

where $H(A_k)$ is the entropy value of attribute A_k;$P(a_{kj})$ is the probability of attribute k being at its jth value; $P(c_i \mid a_{kj})$ is the probability that the class value is c_i when attribute k is at its jth value; M_k is the total number of values for attribute A_k; and N is the total number of different classes (outcomes).

The attribute with the minimum entropy value will be selected as a node in the decision tree to partition the objects. The arcs out of this node represent different values of this attribute. If all the objects in an arc belong to one class, the partition process stops. Otherwise, another attribute will be identified using entropy values to further partition the objects that belong to this arc. This partition process continues until all the objects in an arc are in the same class. Before applying this algorithm, all attributes that have continuous values need to be transformed to discrete values.

In the context of job shop scheduling, the attributes represent system status and the classes represent the dispatching rules. Very often, the attribute values are continuous. Yih (1990) proposed a trace-driven knowledge acquisition (TDKA) methodology to deal with continuous data and to avoid the problems occurring in verbally interviewing human experts. TDKA learns scheduling knowledge from expert schedulers without a dialogue with them. There are three steps in this approach. Step 1, an interactive simulator is developed to mimic the system of interest. The expert will interact with this simulator and make decisions. The entire decision-making process will be recorded in the simulator and can be repeated for later analysis. The series of system information and the corresponding decision collected is called a trace. Step 2 analyzes the trace and forms classification rules to partition the trace into groups. The partition process stops when most of the cases in

each group use the same dispatching rule (error rate is below the threshold defined by the knowledge engineer). Then, the decision rules are formed. Step 3 is to verify the generated rules. The resulting rule base is used to schedule jobs in the simulator. If it performs as well as or better than the expert, the process stops. Otherwise, the threshold value is increased, and the process returns to Step 2.

As the job shop operates over time, it is important to be able to modify the knowledge contained in these rule bases. Chiu (1994) looks at knowledge modification for job shop scheduling problems by a framework of dynamic scheduling schemes that explores routing flexibility and handles uncertainties. The proposed methodology includes three modules: discrete-event simulation, instance generation, and incremental induction. First, a simulation module is developed to implement the dynamic scheduling scheme, to generate training examples, and to evaluate the methodology. Second, in an instance-generation module, the searching of good training examples is successfully fulfilled by a genetic algorithm. Finally, in an incremental-induction module, a tolerance-based incremental learning algorithm is pro-posed to allow continuous learning and facilitate knowledge modification. This algorithm uses entropy values to select attributes to partition the examples where the attribute values are continuous. The tolerance is used to maintain the stability of the existing knowledge while the new example is introduced. The decision tree will not be reconstructed unless there is enough momentum from the new data, that is, the change of the entropy value becomes significant. The experimental results showed that the tolerance-based incremental learning algorithm cannot only reduce the frequency of modifications, but also enhances the generalization ability of the resulting decision tree in a distributed job shop environment.

Theory of Constraints

The Theory of Constraints (TOC) developed by Eliyahu Goldratt (1990, 1992) is the underlying philosophy for synchronized manufacturing. Goldratt (1990) defined synchronized manufacturing as any systematic method that attempts to move material quickly and smoothly through the production process in concert with market demand. A core concept of TOC is the idea that a few critical constraints exist. Goldratt contends that there is only one constraint in a system at any given time. As defined by Dettmer (1997), a constraint is "any element of a system or its environment that limits the output of the system." A constraint will prevent increases in throughput regardless of improvements made to the system. The best schedule is obtained by focusing on the planning and scheduling of these constraint operations. In essence, the constraint operations become the basis from which the entire schedule is derived. TOC has several important concepts and principles. Among them (Goldratt 1990; Goldratt 1992):

1. Systems function like chains.
2. The system optimum is not the sum of the local optima.
3. The effect-cause-effect method identifies constraints.
4. System constraints can be either physically or policy.
5. Inertia is the worst enemy of a process of ongoing improvement.
6. Throughput is the rate at which the entire system generates money through sales.
7. Inventory is all the money the system invests in things it intends to sell.
8. Operating expense is all the money the system spends turning inventory into throughput.

The general process of TOC is as follows (Goldratt 1990):

1. Identify the system's constraints.
2. Decide how to exploit the system's constraints.
3. Subordinate everything else to the above decision.
4. Elevate the system's constraints.
5. If in the previous steps a constraint have been broken, go back to Step 1, but do not allow inertia to cause a system constraint.

TOC has been successfully applied to scheduling problems. Its tools comprise five distinct logic trees, (see Dettmer 1997), are the Current Reality Tree, the Evaporating Cloud Diagram, the Future Reality Tree, the Prerequisite Tree, and the Transition Tree. These trees are tied to the Categories of Legitimate Reservation (that provide the logic to guide the construction of the trees). These tools have not only been used in production scheduling, but also in other enterprise functions such as marketing and sales.

Human-Guided Search

An approach to job shop scheduling is to leverage people's abilities in areas in which they currently outperform computers, Lesh et al. (2003). This will allow human guidance to steer a computer towards effective job shop schedules based on their knowledge of real-world constraints. Furthermore, users can better understand, justify, and modify schedules if they participate in their construction. Lesh et al. (2003) developed a prototype which "allows users to manually modify the current schedule, backtrack to previous schedules, and invoke, monitor, and halt a variety of search algorithms to find better schedules." This interactive scheduling approach provides solutions with the potential to revolutionize the utilization of optimization in actual production environments.

Concluding Remarks

Job shop scheduling problems fall into the class of *NP*-complete problems. They are most difficult to formulate and solve. Operations Research analysts have been pursuing solutions to these problems for many years, with varying degrees of success.

Job shop scheduling problems are among the most important found in manufacturing; they impact the ability of manufacturers to meet customer demands in an effective and profitable manner. They also impact the ability of autonomous systems to optimize their operations, the deployment of intelligent systems, the development of software, and the optimization of communications systems.

See

- ▶ Artificial Intelligence
- ▶ Computational Complexity
- ▶ Dantzig-Wolfe Decomposition Algorithm
- ▶ Decision Trees
- ▶ Expert Systems
- ▶ Flexible Manufacturing Systems
- ▶ Gantt Charts
- ▶ Genetic Algorithms
- ▶ Heuristics
- ▶ Hierarchical Production Planning
- ▶ Integer and Combinatorial Optimization
- ▶ Lagrangian Relaxation
- ▶ Linear Programming
- ▶ Neural Networks
- ▶ Operations Management
- ▶ Project Management
- ▶ Scheduling and Sequencing
- ▶ Simulated Annealing
- ▶ Tabu Search
- ▶ Theory of Constraints

References

Adams, J., Balas, E., & Zawack, D. (1988). The shifting bottleneck procedure for job shop scheduling. *Management Science, 34*, 391–401.

Agin, N. (1966). Optimum seeking with branch and bound. *Management Science, 13*, 176–185.

Baker, K. (1974). *Introduction to sequencing and scheduling*. New York: John Wiley.

Balas, E. (1965). An additive algorithm for solving linear programs with zero-one variables. *Operations Research, 13*, 517–546.

Balas, E. (1967). Discrete programming by the filter method. *Operations Research, 15*, 915–957.

Balas, E. (1969). Machine sequencing via disjunctive graphs: An implicit enumeration algorithm. *Operations Research, 17*, 1–10.

Balas, E. (1970). Machine sequencing: disjunctive graphs and degree-constrained subgraphs. *Naval Research Logistics Quarterly, 17*, 941–957.

Bean, J., & Birge, J. (1986). Match-up real-time scheduling. *NBS Special Publication, 724*, 197–212.

Benders, J. (1960). Partitioning procedures for solving mixed-variables mathematical programming problems. *Numersche Mathematik, 4*(3), 238–252.

Binh, N. & Cing, T. (2005). Evolving dispatching rules for solving the flexible job-shop problem. In *The 2005 IEEE congress on evolutionary computation* (Vol. 3, pp. 2848–2855).

Blackstone, J., Phillips, D., & Hogg, G. (1982). A state-of-the-art survey of dispatching rules for manufacturing job shop operations. *International Journal of Production Research, 20*(1), 27–45.

Burges, C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery, 2*(2), 121–167.

Conway, R., Maxwell, W., & Miller, L. W. (1967). *Theory of scheduling*. Reading, MA: Addison-Wesley.

Dantzig, G., & Wolfe, P. (1960). Decomposition principles for linear programs. *Naval Research Logistics Quarterly, 8*, 101–111.

Daouas, T., Ghedira, K., & Muller, J. (1995). Distributed flow shop scheduling problem versus local optimization. In *Proceedings of the First International Conference on Multi-Agent Systems*. Cambridge, MA: MIT Press.

Davis, L. (1985). Job shop scheduling with genetic algorithms. In *Proceedings of an International Conference on Genetic Algorithms and Their Applications*, Carnegie-Mellon University (pp. 136–140).

Davis, W., & Jones, A. (1988). A real-time production scheduler for a stochastic manufacturing environment. *International Journal of Computer Integrated Manufacturing, 1*(2), 101–112.

Dettmer, W. (1997). *Goldratt's theory of constraints: A systems approach to continuous improvement*. Milwaukee, WI: Quality Press.

Fan, K. & Zhang, R. (2010). An analysis of research in job shop scheduling problem (2000–2009). In *Proceedings of the 2110 IEEE Conference on Advanced Management Science* (pp. 282–288).

Foo, Y. & Takefuji, Y. (1988). Stochastic neural networks for solving job-shop scheduling: Part 2 — Architecture and simulations. In *Proceedings of the IEEE International Conference on Neural Networks* (pp. II283–II290). Piscataway, NJ: IEEE TAB.

Fox, M. (1983). Constraint-directed search: A case study of job shop scheduling. Ph.D. Dissertation, Carnegie-Mellon University, Pittsburgh, PA.

Gershwin, S. (1989). Hierarchical flow control: A framework for scheduling and planning discrete events in manufacturing systems. *Proceedings of IEEE Special Issue on Discrete Event Systems, 77*, 195–209.

Glover, F. (1989). Tabu search-Part I. *ORSA Journal on Computing, 1*, 190–206.

Glover, F. (1990). Tabu search-Part II. *ORSA Journal on Computing, 2*, 4–32.

Glover, F. (1996). Tabu search and adaptive memory programming — Advances, applications and challenges. In R. S. Barr, R. V. Helgason, & J. L. Kennington (Eds.), *Interfaces in computer science and operations research: Advances in meta-heuristics, optimization, and stochastic modeling technologies*. Dordrecht: Kluwer Academic.

Goldberg, D. (1988). *Genetic algorithms in search optimization and machine learning*. Menlo Park, CA: Addison-Wesley.

Goldberg, D. & Lingle, R. (1985). Alleles, loci, and the traveling salesman problem. In *Proceedings of the International Conference on Genetic Algorithms and Their Applications*, Carnegie Mellon University (pp. 162–164).

Goldratt, E. (1990). *Theory of constraints*. Great Barrington, MA: North River Press.

Goldratt, E. (1992). *The goal*. Great Barrington, MA: North River Press.

Grabot, B., & Geneste, L. (1994). Dispatching rules in scheduling: a fuzzy approach. *International Journal of Production Research, 32*, 903–915.

Hopfield, J., & Tank, D. (1985). Neural computation of decisions in optimization problems. *Biological Cybernetics, 52*, 141–152.

Jeffcoat, D., & Bulfin, R. (1993). Simulated annealing for resource-constrained scheduling. *European Journal of Operational Research, 70*, 43–51.

Jones, K. (2005). Comparison of genetic algorithm and particle swarm optimization. In *International conference on computer systems and technologies*, Technical University, Varna, Bulgaria.

Kennedy, J. & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks*, Perth, Australia.

Kirkpatrick, S., Gelatt, C., & Vecchi, M. (1983). Optimization by simulated annealing. *Science, 220*(4598), 671–680.

Krucky, J. (1994). Fuzzy family setup assignment and machine balancing. *Hewlett-Packard Journal, 45*, 51–64.

Lawler, E., & Wood, D. (1966). Branch and bound methods: A survey. *Operations Research, 14*, 699–719.

Lesh, N., Lopes, L., Marks, J., Mitzenmacher, M., & Schafer, G. (2003). *Human-guided search for jobshop scheduling*, technical report TR2002-43, Mitsubishi Electric Research Laboratories.

Lo, Z. & Bavarian, B. (1991). Scheduling with neural networks for flexible manufacturing systems. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Sacramento, CA (pp. 818–823).

McKenzie, L. (1976). Turnpike theory. *Econometrics, 44*, 841–864.

Montazer, M., & Van Wassenhove, L. (1990). Analysis of scheduling rules for an FMS. *International Journal of Production Research, 28*, 785–802.

Morton, E., & Pentico, D. (1993). *Heuristic scheduling systems*. New York: John Wiley.

Panwalker, S., & Iskander, W. (1977). A survey of scheduling rules. *Operations Research, 25*, 45–61.

Parunak, H., Irish, B., Kindrick, J., & Lozo, P. (1985). Fractal actors for distributed manufacturing control. In *Proceedings of the Second IEEE Conference on Artificial Intelligence Applications* (pp. 653–660).

Quinlan, J. (1986). Induction of decision trees. *Machine Learning, 1*, 81–106.

Quinlan, J. (1992). *C4. 5: Programs for machine learning*. New York: Morgan Kaufmann.

Rabelo, L. (1990). Hybrid artificial neural networks and knowledge-based expert systems approach to flexible manufacturing system scheduling. Ph.D. Dissertation, University of Missouri-Rolla.

Rabelo, L., Sahinoglu, M., & Avula, X. (1994). Flexible manufacturing systems scheduling using Q-Learning. In *Proceedings of the World Congress on Neural Networks*, San Diego, CA (pp. I378–I385).

Roshanaei, V., Naderi, B., Jolai, F., & Khalili, M. (2009). A variable neighborhood search for job shop scheduling with set-up times to minimize makespan. *Future Generation Computer Systems, 25*, 654–661.

Rumelhart, D., McClelland, J., & the PDP Research Group. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition* (Foundations, Vol. 1). Cambridge, MA: MIT Press.

Saleh, A. (1988). Real-time control of a flexible manufacturing cell. Ph.D. Dissertation, Lehigh University, Bethlehem, PA.

Shahzad, A. & Mebarki, N. (2008). Discovering dispatching rules for job shop scheduling problem through data mining. In *8th International Conference of Modeling and Simulation* - MOSIM'10 - May 10–12.

Shapiro, J. (1979). A survey of Lagrangian techniques for discrete optimization. *Annals Discrete Mathematics, 5*, 113–138.

Shiuea, Y. (2009). Data-mining-based dynamic dispatching rule selection mechanism for shop floor control systems using

a support vector machine approach. *International Journal of Production Research, 47*(13), 3669–3690.
Slany, W. (1994). Scheduling as a fuzzy multiple criteria optimization problem. CD-Technical Report 94/62, Technical University of Vienna.
Smith, S. (1995). OPIS: A methodology and architecture for reactive scheduling. In M. Zweben & M. Fox (Eds.), *Intelligent scheduling* (pp. 29–66). San Francisco, CA: Morgan Kaufman.
Srinivasan, V. (1971). A hybrid algorithm for the one machine sequencing problem to minimize total tardiness. *Naval Research Logistics Quarterly, 18*, 317–327.
Starkweather, T., Whitley, D., & Cookson, B. (1993). A genetic algorithm for scheduling with resource consumption. In *Proceedings of the Joint German/US Conference on Operations Research in Production Planning and Control* (pp. 567–583).
Starkweather, T., Whitley, D., Mathias, K., & Mc-Daniel, S. (1992). Sequence scheduling with genetic algorithms. In *Proceedings of the US/German Conference on New Directions for OR in Manufacturing* (pp. 130–148).
Tesauro, G. (1992). Practical issues in temporal difference learning. *Machine Learning, 8*, 257–277.
Thamilselvan, R., & Balasubramanie, P. (2009). Integrating genetic algorithm, tabu search approach for job shop scheduling. *International Journal of Computer Science and Information Security, 2*(1), 134–139.
Tsujimura, Y., Park, S., Chang, S., & Gen, M. (1993). An effective method for solving flow shop scheduling problems with fuzzy processing times. *Computers and Industrial Engineering, 25*, 239–242.
Vakharia, A., & Chang, Y. (1990). A simulated annealing approach to scheduling a manufacturing cell. *Naval Research Logistics, 37*, 559–577.
Vapnik, V. (1994). Principles of risk minimization for learning theory. In J. Moody, S. Hanson, & R. Lippmann (Eds.), *Advances in neural information processing system* (Vol. 4, 831–838). Morgan Kaufman.
Vapnik, V. (1995). *The nature of statistical learning theory*. Springer.
Watkins, C. (1989). Learning from delayed rewards. Ph.D. Dissertation, King's College, Cambridge, England.
Werbos, P. (1995). Neurocontrol and supervised learning: An overview and evaluation. In *Handbook of intelligent control: Neural, fuzzy, and adaptive approaches* (pp. 65–89). New York: Van Nostrand Reinhold.
Wilkerson, L., & Irwin, J. (1971). An improved algorithm for scheduling independent tasks. *AIIE Transactions, 3*, 239–245.
Wu, D. (1987). An expert systems approach for the control and scheduling of flexible manufacturing systems. Ph.D. Dissertation, Pennsylvania State University.
Wysk, R., Wu, D., & Yang, F. (1986). A multi-pass expert control system (MPECS) for flexible manufacturing systems. *NBS Special Publication, 724*, 251–278.
Yen, G., & Ivers, B. (2009). Job shop scheduling optimization through multiple independent particle swarms. *International Journal of Intelligent Computing and Cybernetics, 2*(1), 5–33.
Yih, Y. (1990). Trace-driven knowledge acquisition (TDKA) for rule-based real-time scheduling systems. *Journal of Intelligent Manufacturing, 1*, 217–230.
Zentner, M., Peony, J., Relates, G., & Gupta, N. (1994). Practical considerations in using model-based optimization for the scheduling and planning of batch/semi-continuous processes. *Journal of Process Control, 4*, 259–280.
Zhang, M. & Zhang, C. (1995). The consensus of uncertainties in distributed expert systems. In *Proceedings of the First International Conference on Multi-Agent Systems*. Cambridge, MA: MIT Press.
Zhou, D., Cherkassy, V., Baldwin, T., & Hong, D. (1990). Scaling neural networks for job shop scheduling. In *Proceedings of the International Conference on Neural Networks* (Vol. 3, pp. 889–894).
Zhou, Q., Cui, X., Wang, Z., & Yang, B. (2009). A hybrid optimization algorithm for the job-shop scheduling problem. In *Proceedings of the GEC'09*, June 12–14, Shanghai, China (pp. 757–763).
Zweben, M., Daunt, B., Davis, E., & Deale, M. (1995). Scheduling and rescheduling with iterative repair. In M. Zweben & M. Fox (Eds.), *Intelligent scheduling* (pp. 241–256). San Francisco, CA: Morgan Kaufman.

Johnson's Theorem

► Scheduling and Sequencing

Judgmental Bootstrapping

In judgmental bootstrapping, a forecaster's rules are inferred by regressing the forecasts against the inputs that were used to make the forecasts.

See

► Bootstrapping
► Forecasting

Just-in-Time (JIT) Manufacturing

A manufacturing philosophy focusing on the elimination of waste (non-value added activities) in the manufacturing process by the more timely sequencing of operations.

Most of the early ideas originated from work at Toyota Motor Company in Japan, with the kanban system being the most well-known innovation.

See

- ▶ Kanban
- ▶ Material Handling
- ▶ Production Management
- ▶ Pull System

References

Hopp, W. J., & Spearman, M. L. (2008). *Factory physics* (3rd ed.). McGraw-Hill.

K

Kanban

Japanese phrase literally meaning "sign board" referring to a Just-in-Time (JIT) pull production system made famous by Toyota, where (kanban) cards are used to signal that parts are needed.

See

- ▶ CONWIP
- ▶ Just-in-Time (JIT) Manufacturing
- ▶ Production Management
- ▶ Pull System

References

Hopp, W. J., & Spearman, M. L. (2008). *Factory physics* (3rd ed.). McGraw-Hill.

Karmarkar's Algorithm

An algorithm devised by N. Karmarkar (1984) for solving a linear-programming problem by generating a sequence of points lying in the strict interior of the problem's solution space which converges to an optimal solution. Karmarkar's algorithm, and its many variations, have been shown to be polynomial-time algorithms that solve large-scale linear-programming problems in a computationally efficient manner.

See

- ▶ Interior-Point Methods for Conic-Linear Optimization
- ▶ Polynomially Bounded (−Time) Algorithm (Polynomial Algorithm)

References

Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatoria, 4*(4), 373–395.

Karush-Kuhn-Tucker (KKT) Conditions

The Karush-Kuhn-Tucker (KKT) conditions are necessary conditions that a solution to a general nonlinear-programming problem must satisfy, provided that the problem constraints satisfy a regularity condition called constraint qualification. If the problem is one in which the constraint set (i.e., solution space) is convex and the maximizing (minimizing) objective function is concave (convex), the KKT conditions are sufficient. Applied to a linear-programming problem, the KKT conditions yield the complementary slackness conditions of the primal and dual problems.

See

- ▶ Calculus of Variations
- ▶ Nonlinear Programming
- ▶ Quadratic Programming

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

References

Karush, W. (1939). *Minima of functions of several variables with inequalities as side constraints*. M.Sc. Dissertation. Depatment of Mathematics, University of Chicago, Chicago, IL.

Kuhn, H. W., & Tucker, A. W. (1951). Nonlinear programming. In *Proceedings of 2nd Berkeley Symposium* (pp. 481–492). Berkeley: University of California Press.

Kendall's Notation

A shorthand notation of the form $A/S/c/K/Q$ used to describe queueing systems. The A refers to an acronym for the interarrival-time distribution, S the service-time distribution, c the number of parallel servers, $K(\geq c)$ the maximum allowable system size (assumed infinite if omitted), and Q the queue discipline. Common designations for A and S include: M for Markovian or exponential; E_k for k-Erlang; D for deterministic or constant; H_k for hyperexponential of order k; PH for phase-type; G for general. Common queue disciplines include first-in, first-out (FIFO), first-come, first-served (FCFS), last-come, first-served (LCFS), and processor sharing (PS). Sometimes another parameters is included between K and Q that gives the arrival population size, which is assumed infinite otherwise. For example, an $M/M/1$ queue generally refers to a FCFS single-server queue with a Poisson arrival process and i.i.d. exponentially distributed service times.

See

▶ Queueing Theory

Kilter Conditions

For the minimum cost flow network problem, the complentary slackness optimality conditions are called kilter conditions.

See

▶ Out-of-Kilter Algorithm

KKT Conditions

▶ Karush-Kuhn-Tucker (KKT) Conditions
▶ Nonlinear Programming
▶ Quadratic Programming

Klee-Minty Problem

The Klee-Minty problem is a linear-programming problem designed to demonstrate that a problem exists that would require the simplex algorithm to generate all extreme point solutions before finding the optimal. This problem demonstrated that, although the simplex algorithm (under a nondegeneracy assumption) would find an optimal solution in a finite number of iterations, the number of iterations can increase exponentially. Thus, the simplex method is not a polynomially bounded algorithm. One form of the Klee-Minty problem, which defines a slightly perturbed hypercube, is the following:

$$\begin{aligned}
&\text{Minimixe } -x_d \\
&\text{subject to } x_1 \geq 0 \\
&x_1 \leq 1 \\
&-\varepsilon x_1 + x_2 \geq 0 \\
&\varepsilon x_1 + x_2 \leq 1 \\
&\vdots \\
&-\varepsilon x_{d-1} + x_d \geq 0 \\
&\varepsilon x_{d-1} + x_d \leq 1 \\
&x_j \geq 0
\end{aligned}$$

with $0 < \varepsilon < 1/2$.

References

Klee, V., & Minty, G. (1972). How good is the simplex algorithm? In O. Shisha (Ed.), *Inequalities: III* (pp. 159–175). New York: Academic Press.

Knapsack Problem

The following optimization problem is called the knapsack problem:

$$\begin{aligned} \text{Maximize} \quad & c_1x_1 + c_2x_2 + \cdots + c_nx_n \\ \text{subject to} \quad & a_1x_1 + a_2x_2 + \cdots + a_nx_n \le b \end{aligned}$$

with each x_j equal to 0 or 1, with all (a_j, c_j, b) usually taken to be positive integers. The name is due to interpreting the problem as one in which a camper has a knapsack that can carry up to b pounds. The camper has a choice of packing up to n items, with $x_j = 1$ if the item is packed and $x_j = 0$ if the item is not packed. Item j weighs a_j pounds. Each item has a value c_j to the camper if it is packed. The camper wishes to choose that collection of items having the greatest total value subject to the weight condition. The knapsack problem arises in many applications such as selecting a set of projects and as a subproblem of other problems. It can be solved by dynamic programming or by integer-programming methods. If the x_j are ordered such that $c_1/a_1 \ge c_2/a_2 \ge \ldots \ge c_n/a_n$ and the integer restrictions on the variables are replaced by $0 \le x_j \le 1$, then an optimal solution to the relaxed problem is to just pack all the items starting with the first until the weight restriction is violated. The item that caused the violation is then chosen at a fractional value so that the total weight of the selected set is equal to b.

See

▶ Knapsack Problems with Nonlinearities

Knapsack Problems with Nonlinearities

Kurt M. Bretthauer[1,2] and Bala Shetty[1]
[1]Texas A&M University, College Station, TX, USA
[2]Indiana University Bloomington, Bloomington, IN, USA

Introduction

Nonlinear optimization problems with a single constraint are known as nonlinear knapsack problems or nonlinear resource allocation problems, and will be formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^{n} f_i(x_i) \qquad \text{(IP)} \\ \text{s.t.} \quad & \sum_{i=1}^{n} g_i(x_i) \le b \\ & l_i \le x_i \le u_i, \quad i = 1, \ldots, n \\ & x_i \text{ integer}, \; i = 1, \ldots, n \end{aligned}$$

The continuous variable version of the problem also will be considered, and will be denoted as problem (CP). Assume $x_i \varepsilon \Re$ for $i = 1, \ldots, n$; $f_i(x_i)$ and $g_i(x_i)$ for $i = 1, \ldots, n$ are differentiable functions on $\Re$; b is a constant; l_i and u_i for $i = 1, \ldots, n$ are lower and upper bounds on the variables; and the feasible region is nonempty and bounded.

Practical applications of the nonlinear knapsack problem abound in financial models (Mathur et al. 1983), production and inventory management (Bretthauer et al. 1994), stratified sampling (Bretthauer et al. 1999; Cochran 1963), and the optimal design of queueing network models in manufacturing (Bitran and Tirupati 1989; Bretthauer 1996), health care, and computer systems (Gerla and Kleinrock 1977). The fact that the problem has one constraint allows efficient solution methods to be developed that take advantage of its special structure. Solution methods have been developed for the following versions of the problem: (1) continuous and integer variables, (2) constraints of the form $\sum_{i=1}^{n} x_i = b$ and $\sum_{i=1}^{n} g_i(x_i) = b$, (3) convex and nonconvex functions, and (4) additional specially structured constraints.

Continuous Convex Problems

Two basic approaches for solving the continuous problem (CP) are multiplier search methods and relaxation or variable pegging methods. Assume all objective and constraint functions are convex. Bretthauer and Shetty (1995) presented a multiplier search algorithm that solves the problem via a one-dimensional search for the optimal Lagrange multiplier of the single constraint. Let λ denote the Lagrange multiplier for $\sum_{i=1}^{n} g_i(x_i) \le b$, let v_i

denote the multiplier for $l_i \le x_i$, and let w_i denote the multiplier for $x_i \le u_i$. Also, let f_i' denote the derivative of $f_i(x_i)$ with respect to x_i and let g_i' denote the derivative of $g_i(x_i)$ with respect to x_i. Assume that a solution $\bar{x}_i(\lambda)$ exists to the nonlinear equation $f_i' + g_i' = 0$ as a function of λ for $i = 1, \ldots, n$. Consider the following expressions:

$$x_i(\lambda) = \begin{cases} l_i & \text{if } \bar{x}_i(\lambda) \le l_i \\ \bar{x}_i(\lambda) & \text{if } l_i < \bar{x}_i(\lambda) < u_i \\ u_i & \text{if } \bar{x}_i(\lambda) \ge u_i \end{cases}$$

$$v_i(\lambda) = \begin{cases} f_i'(l_i) + \lambda g_i'(l_i) & \text{if } \bar{x}_i(\lambda) \le l_i \\ 0 & \text{if } \bar{x}_i(\lambda) > l_i \end{cases}$$

$$w_i(\lambda) = \begin{cases} 0 & \text{if } \bar{x}_i(\lambda) < u_i \\ f_i'(l_i) + \lambda g_i'(l_i) & \text{if } \bar{x}_i(\lambda) \ge u_i \end{cases}$$

It can be shown that these equations for $x_i(\lambda), v_i(\lambda)$, and $w_i(\lambda)$ satisfy all the Karush-Kuhn-Tucker (KKT) conditions of problem (CP) except $\sum_{i=1}^{n} g_i(x_i) \le b$ and $\lambda(\sum_{i=1}^{n} g_i(x_i) - b) = 0$ for every $\lambda \ge 0$. Problem (CP) is solved by searching for the optimal λ value such that these two remaining KKT conditions are also satisfied. This search requires finding the root of one nonlinear equation if $\bar{x}_i(\lambda)$ can be written in closed form as a function of λ, or finding the root of one nonlinear equation several times if $\bar{x}_i(\lambda)$ cannot be written in closed form as a function of λ. The optimal solution to problem (CP) is obtained by substituting the optimal λ value into the equations for $x_i(\lambda), v_i(\lambda)$, and $w_i(\lambda)$.

Relaxation or variable pegging algorithms have been developed by Kodialam and Luss (1998) for the problem with nonnegativity conditions and Bretthauer and Shetty (1990) for the general problem (CP) with lower and upper bounds on the variables. See Bitran and Hax (1981), Ibaraki and Katoh (1988), and Robinson, Jiang, and Lerme (1992) for pegging algorithms for more specialized versions of the problem. These algorithms are based on the observation that if the bounds on the variables are ignored, then the resulting relaxed problem is generally easier to solve than problem (CP) and often has a simple closed form solution. Each iteration of this class of algorithms solves a relaxed subproblem of this type and then fixes (or pegs) a subset of the variables not satisfying their bounds at either their lower or upper bounds. Pegging variables reduces the size of the relaxed subproblem that must be solved in the next iteration. The algorithm terminates in a finite number of iterations when all unpegged variables in the solution to a relaxed subproblem satisfy their bounds.

Special versions of the problem where all terms in the objective function are convex and the single constraint is a simple sum of the variables of the form $\sum_{i=1}^{n} x_i = b$ have been widely studied. Ibaraki and Katoh (1988) provide an excellent discussion of methods for solving both continuous and integer variable versions of this special class of nonlinear knapsack problems. A related version of the problem with a strictly convex quadratic objective and a linear constraint is discussed in (Bretthauer et al. 1995; Brucker 1984; Helgason et al. 1980; Pardalos and Kovoor 1990; Robinson et al. 1992; Shetty and Muthukrishnan 1990). Nielsen and Zenios (1992) addressed the convex objective and linear constraint problem.

Integer Convex Problems

Of special interest is the nonlinear knapsack problem (IP) with integer variables and all functions assumed convex. The most common methods for solving the integer convex version of the problem include branch and bound (Bretthauer and Shetty 1995, 1998); (0, 1) linearization (Hochbaum 1995; Mathur et al. 1986); and dynamic programming (Denardo 1982). In addition, Lawler (1979) describes an approximation algorithm for problem (IP).

Branch and bound algorithms for solving the integer problem (IP) solve a series of continuous subproblems of the form of (CP) using the algorithms discussed in the previous section. Bretthauer and Shetty (1995) show how the performance of the branch and bound algorithm can be improved via reoptimization procedures that reduce the time spent solving each subproblem and by heuristically generating feasible integer solutions to problem (IP) from the fractional subproblem solutions.

Hochbaum (1995) and Mathur, Salkin, and Mohanty (1986) present a method for solving problem (IP) that first converts it to an equivalent linear (0, 1) knapsack problem with a larger number of variables, and then solves the resulting (0, 1) knapsack problem. Hochbaum assumes the objective is the sum of

nonincreasing convex functions and the single constraint is the sum of nondecreasing convex functions. Mathur, Salkin, and Mohanty assumes the objective is the sum of decreasing convex functions and the single constraint is the sum of increasing convex functions. Problem (IP) can be converted to an equivalent linear (0, 1) knapsack problem as follows. For $i = 1, \ldots, n$ let $p_{ij} = f_i(l_i + j) - f_i(l_i + j + 1)$ and $q_{ij} = g_i(l_i + j) - g_i(l_i - j - 1)$ for $j = 1, \ldots, u_i - l_i$. Using these coefficients, problem (IP) is equivalent to the following linear knapsack problem:

$$\text{Minimize} \quad \sum_{i=1}^{n} \sum_{j=1}^{u_i - l_i} p_{ij} x_{ij} + \sum_{i=1}^{n} f_i(l_i)$$

$$\text{s.t.} \quad \sum_{i=1}^{n} \sum_{j=1}^{u_i - l_i} q_{ij} x_{ij} \leq b - \sum_{i=1}^{n} g_i(l_i)$$

$$x_{ij} \varepsilon \{0, 1\}, \; i = 1, \ldots, n; \; j = 1, \ldots, u_i - l_i$$

The solution to problem (IP) is obtained from the solution to the above problem using $x_i = l_i + \sum_{j-1}^{u_i - l_i} x_{ij}$ for $i = 1, \ldots, n$.

Nonconvex Problems

Assume that any of the functions $f_i(x_i)$ and $g_i(x_i)$ for all i may be nonconvex. This modification makes the continuous problem much more difficult to solve because a locally optimal solution may not be globally optimal. However, nonconvex functions often arise in practice, in particular concave functions. For example, fixed charges and costs exhibiting economies of scale can be modeled with a concave function.

Branch and bound methods for solving continuous and integer nonconvex optimization problems typically use convex underestimating functions to the nonconvex functions for lower bounding purposes in each subproblem of the search tree. This approach results in continuous convex subproblems that can be solved with the methods previously discussed. The convex envelope is commonly used as an underestimating function to a nonconvex function. Simply stated, the convex envelope is the highest possible convex function that underestimates the nonconvex function over a given set (see Horst and Tuy 1990, for a formal definition). Let $f_i^c(x_i)$ denote the convex envelope of $f_i(x_i)$ over $l_i \leq x_i \leq u_i$ and let $g_i^c(x_i)$ denote the convex envelope of $g_i(x_i)$ over $l_i \leq x_i \leq u_i$. The convex envelope is particularly attractive as an underestimating function to a separable concave function because then it is easy to construct and linear.

A continuous convex underestimating subproblem of the form

$$\text{Minimize} \quad \sum_{i=1}^{n} f_i^c(x_i)$$

$$\text{s.t.} \quad \sum_{i=1}^{n} g_i^c(x_i) \leq b$$

$$l_i \leq x_i \leq u_i, \quad i = 1, \ldots, n$$

is solved at every node in the branch-and-bound tree using the results discussed in the section on continuous convex problems. The lower and upper bounds on the variables will change at the nodes as branching is performed. *See* Horst and Tuy (1990) for further details on algorithms for solving continuous nonconvex problems and convergence results. Incorporating the above subproblem within the prototype branch and bound framework for discrete global optimization developed by Benson, Erenguc, and Horst (1990) yields a globally optimal solution to the integer nonconvex problem (IP) in a finite number of iterations.

Handling Additional Specially Structured Constraints

It is also possible to develop efficient algorithms for solving problems that include additional specially structured constraints. For example, consider the continuous convex problem with a type of block diagonal or angular structure in the constraints (Bretthauer and Shetty 1997; Federgruen and Zipkin 1983; Hochbaum 1994):

$$\text{Minimize} \quad \sum_{i \varepsilon S}^{n} f_i(x_i)$$

$$\text{s.t.} \quad \sum_{i \varepsilon S} g_i(x_i) \leq b$$

$$\sum_{i \varepsilon S_k} h_i(x_i) \leq c_k, \quad k = 1, \ldots, K$$

$$l_i \leq x_i \leq u_i, \quad i \varepsilon S$$

In this problem, S is the index set of the variables; $S_1, S_2, \ldots, S_K$ are disjoint subsets of S; the objective

function is assumed convex; the single constraint $\sum_{i \varepsilon S} g_i(x_i) \leq b$ involving all the variables is convex; and the K constraints $\sum_{i \varepsilon S_k} h_i(x_i) \leq c_k$ involving disjoint subsets of the variables are convex.

To solve the above problem, first form a Lagrangian relaxation with respect to the one constraint $\sum_{i \varepsilon S} g_i(x_i) \leq b$. The resulting Lagrangian subproblem then decomposes into K singly constrained convex problems. These singly constrained problems can be solved with the previously discussed methods. A one-dimensional search must be performed to identify the optimal value of the single Lagrange multiplier.

See

- ▶ Integer and Combinatorial Optimization
- ▶ Knapsack Problem
- ▶ Linear Programming
- ▶ Nonlinear Programming

References

Benson, H. P., Erenguc, S. S., & Horst, R. (1990). A note on adapting methods for continuous global optimization to the discrete case. *Annals of Operations Research, 25*, 243–252.

Bitran, G. R., & Hax, A. C. (1981). Disaggregation and resource allocation using convex knapsack problems with bounded variables. *Management Science, 27*, 431–441.

Bitran, G. R., & Tirupati, D. (1989). Tradeoff curves, targeting and balancing in manufacturing queueing networks. *Operations Research, 37*, 547–564.

Bretthauer, K. M. (1996). Capacity planning in manufacturing and computer networks. *European Journal of Operational Research, 91*, 386–394.

Bretthauer, K. M., & Shetty, B. (1995). The nonlinear resource allocation problem. *Operations Research, 43*, 670–683.

Bretthauer, K. M., & Shetty, B. (1997). Quadratic resource allocation with generalized upper bounds. *Operations Research Letters, 20*, 51–57.

Bretthauer, K. M., & Shetty, B. (1999). *A pegging algorithm for the nonlinear resource allocation problem*. College Station, TX: Department of Information and Operations Management, Texas A&M University.

Bretthauer, K. M., Shetty, B., & Ross, A. (1999). Nonlinear integer programming for optimal allocation in stratified sampling. *European Journal of Operational Research*.

Bretthauer, K. M., Shetty, B., & Syam, S. (1995). A branch and bound algorithm for integer quadratic knapsack problems. *ORSA Journal on Computing, 7*, 109–116.

Bretthauer, K. M., Shetty, B., Syam, S., & White, S. (1994). A model for resource constrained production and inventory management. *Decision Sciences, 25*, 561–580.

Brucker, P. (1984). An O(n) algorithm for quadratic knapsack problems. *Operations Research Letters, 3*, 163–166.

Cochran, W. G. (1963). *Sampling techniques* (2nd ed.). New York: Wiley.

Denardo, E. V. (1982). *Dynamic programming algorithms and applications*. Englewood Cliffs, NJ: Prentice-Hall.

Federgruen, A., & Zipkin, P. (1983). Solution techniques for some allocation problems. *Mathematical Programming, 25*, 13–24.

Gerla, M., & Kleinrock, L. (1977). On the topological design of distributed computer networks. *IEEE Transactions on Communications, 25*, 48–60.

Helgason, R., Kennington, J., & Lall, H. (1980). A polynomially bounded algorithm for a singly constrained quadratic program. *Mathematical Programming, 18*, 338–343.

Hochbaum, D. S. (1994). Lower and upper bounds for the allocation problem and other nonlinear optimization problems. *Mathematics of Operations Research, 17*, 103–110.

Hochbaum, D. S. (1995). A nonlinear knapsack problem. *Operations Research Letters, 17*, 103–110.

Horst, R., & Tuy, H. (1990). *Global optimization: Deterministic approaches*. Berlin: Springer.

Ibaraki, T., & Katoh, N. (1988). *Resource allocation problems*. Cambridge, MA: MIT Press.

Kodialam, M. S., & Luss, H. (1998). Algorithms for separable nonlinear resource allocation problems. *Operations Research, 46*, 272–284.

Lawler, E. L. (1979). Fast approximation algorithms for knapsack problems. *Mathematics of Operations Research, 4*, 339–356.

Martello, S., & Toth, P. (1990). *Knapsack problems: Algorithms and computer implementations*. New York: Wiley.

Mathur, K., Salkin, H. M., & Mohanty, B. B. (1986). A note on a general non-linear knapsack problem. *Operations Research Letters, 5*, 79–81.

Mathur, K., Salkin, H. M., & Morito, S. (1983). A branch and search algorithm for a class of nonlinear knapsack problems. *Operations Research Letters, 2*, 155–160.

Nielsen, S. S., & Zenios, S. A. (1992). Massively parallel algorithms for singly constrained convex programs. *ORSA Journal on Computing, 4*, 166–181.

Pardalos, P. M., & Kovoor, N. (1990). An algorithm for a singly constrained class of quadratic programs subject to upper and lower bounds. *Mathematical Programming, 46*, 321–328.

Robinson, A. G., Jiang, N., & Lerme, C. S. (1992). On the continuous quadratic knapsack problem. *Mathematical Programming, 55*, 99–108.

Shetty, B., & Muthukrishnan, R. (1990). A parallel projection for the multicommodity network model. *Journal of the Operational Research Society, 41*, 837–842.

Zipkin, P. (1980). Simple ranking methods for allocation of one resource. *Management Science, 26*, 34–43.

Knowledge Acquisition

The activity of eliciting, structuring, analyzing knowledge from some source of expertise and representing it in a form that can be used by an inference engine.

See

- ▶ Artificial Intelligence
- ▶ Expert Systems
- ▶ Inference Engine

Knowledge Base

That part of an expert system containing application-specific reasoning knowledge that the inference engine uses in the course of reasoning about a problem. In expert systems whose reasoning knowledge is represented as rules, the knowledge base is a rule set or rule base. A knowledge base can also contain other kinds of knowledge.

See

- ▶ Artificial Intelligence
- ▶ Expert Systems

Knowledge Engineer

One who develops an expert system, or one who elicits reasoning knowledge from a human expert for use in an expert system.

See

- ▶ Expert Systems

Knowledge Management

Heiner Müller-Merbach
Technische Universität Kaiserslautern,
Kaiserslautern, Germany

Introduction

Knowledge management (KM) is a modern term based on old philosophical insights, such as:

- "All I know is that I know nothing." (Socrates, 490-399 BC).
- "Knowledge itself is power." (Francis Bacon, 1561-1626).
- "Cogito ergo sum." ("I think, therefore I am.") (René Descartes, 1596-1650).
- "Where is the wisdom we have lost in knowledge? Where is the knowledge we have lost in information?" (T. S. Eliot, 1888-1965; in: *The Rock*, 1934).

Two views of knowledge management: In most publications on knowledge management, only one of two aspects of KM is emphasized: Either (i) management, leadership, learning, group organization, i.e. human aspects, or (ii) information systems, i.e. technical aspects. Or as Begona Lloria (2003, pp. 77, 88) put it in her extended review: "... managing knowledge (either with greater emphasis on the human factor or on information technologies)."

Since (i) most of the information technologies used in KM were known beforehand and since (ii) typologies of basic knowledge are neglected in the KM literature, emphasis will here be put on the content of KM systems, i.e. the types of knowledge. References will be given to the relevant philosophers.

Francis Bacon, forerunner of enlightenment: The British philosopher Francis Bacon (1561-1626) taught: "Knowledge itself is power." His intention was the emancipation of the mind from the predominance by the church. His doctrine was accompanied by technological development, according to Durant (1926, p. 105) "Paper now came cheaply from Egypt, replacing the costly parchment that had made learning the monopoly of priests; printing, which had long awaited an inexpensive medium, broke out like a liberated explosive, and spread its destructive and clarifying influence everywhere."

Bacon, in his publication, *"The Praise of Knowledge"* (1592), even defined the individual by his/her knowledge: "My praise shall be dedicate to the mind itself. The mind is the man, and knowledge mind; a man is but what he knoweth" (Durant, 1926; p. 111). Today, one might not fully agree with this statement because everybody is also determined by his/her character, his/her morals and other attributes, not only by his/her knowledge.

Such ideas as those by Bacon are the forerunners of the epoch of enlightenment, i.e. the philosophy of the 17th and 18th century with Descartes in

France (1596-1650), John Locke in England (1732-1704), Kant in Germany (1724-1804) and many others who set the foundation of modern science.

Did Francis Bacon ring in the knowledge age and/or the knowledge society, perhaps even KM? Possibly, at least in a very rudimentary state. But he did not mean printed material as an end; instead, he meant individual human knowledge – only supported by printed material as a means to the end.

Bacon argues with passion: "Is not that knowledge alone that doth clear the mind of all perturbations?" (Durant, 1926; p. 111).

Russell (1979, p. 527) wrote: "Bacon's most important book, *The Advancement of Learning*, is in many ways remarkably modern. . . . The whole basis of his philosophy was practical: to give mankind mastery over the forces of nature by means of scientific discoveries and inventions."

Bacon did not explicitly consider KM; how could he, 400 years ago? He concentrated, however, on human knowledge, and not its sediment: printed (or electronically stored) information. For him, printed material (and the same is true for electronic devices) are only carriers of information and help to increase human knowledge, but the printed documents, as such, do not own knowledge themselves, because they do not have consciousness.

There are some relations between Francis Bacon and René Descartes who was 35 years younger than Bacon. One of the most important doctrines by Descartes was the sentence: *"Cogito ergo sum,"* ("I think, therefore I am.") (Angeles, 1992; p. 47 f.). To think means creation of knowledge, of understanding, of insight.

Socrates and his pretended ignorance: About 2,000 years prior to Bacon, Socrates postulated: "All I know is that I know nothing," (Angeles, 1992; p. 280). Was it only modesty? Certainly not! Or was it an insincere understatement, because he was considered one of the wisest men in his time, and he probably knew it? Again no! Instead, he was convinced that his own knowledge was almost negligible in comparison to the conglomeration of all the knowledge of mankind and to the not yet discovered secrets of the world.

The skepticism outspoken by Socrates in the quoted sentence, was his certainty. And it is (at least almost) true for everybody. The piece of knowledge that anybody owns is nearly zero compared to the knowledge of the world and the hidden secrets of nature.

In spite of the small amount of knowledge that anybody owns, it is worth to expand his/her knowledge because knowledge helps to master one's own live: live long learning.

Discussion

Knowledge Networks: Here lies a trigger for KM: Since the knowledge of anybody is – in spite of any individual's effort in learning – narrowly restricted, the cooperation between groups of individuals may have the potential to increase the available group knowledge remarkably. This requires leadership, i.e. knowledge management.

This goes together with the development of information processing. The German philosopher, Jürgen Mittelstrass (born 1936), has repeatedly warned that all of us may become information giants and, at the same time, be knowledge dwarfs (Mittelstrass. 1972; p. 8d). This means that each of us may have immediate access to almost any information available through information systems, but, at the same time, understand less and less of the information available. This deficiency can to some extent be overcome by knowledge networks.

Knowledge a priori versus knowledge a posteriori: Knowledge has alternatively two different origins (Müller-Merbach, 2007a): deduction (emphasized by Descartes) and induction (emphasized by Locke).

René Descartes founded French rationalism and suggested the deductive method of knowledge creation. Russell (1979; p. 549), interprets Descartes: "Knowledge of external things must be by the mind, not by the senses." Angeles (1992; p. 157) seconds: "All knowledge is derived by a deductive process similar to that in axiomatic geometry from this primitive and absolutely infallible truth."

Example 1. Any triangle has an angular sum of 180°, any quadrangle an angular sum of 360°, any polygon with n nodes an angular sum of (n-2)*180°. This can be derived by deduction; no empirical experience is necessary. "The truth of a priori knowledge (a) is not derived from sense experience, (b) cannot be checked against sense experience, and (c) cannot be refuted by any sense experience," (Angeles 1992; p. 159).

The other origin of knowledge is British empiricism. According to Russell (1979; p. 589), John Locke "may be regarded as the founder of empiricism, which is the doctrine that all our knowledge ... is derived from experience." Other empiricists are George Berkeley (Ireland, 1685-1753) and David Hume (Scotland, 1711-1776). According to Angeles (1992; p. 85), empiricism is defined as "the view that all ideas are abstractions formed by compounding ... what is experienced." He added: "Experience is the sole source of knowledge," and "All that we know is ultimately dependent on sense data."

It was Immanuel Kant (Germany) who came up with a synthesis between the rational, deductive, French, Cartesian doctrine and the empirical, inductive, British process of knowledge creation. He termed them a priori and a posteriori knowledge.

Karl R. (Charles) Popper (1902-1994) stated that scientific discoveries based upon empirical experience can never be proven to be right. Instead, they are subject to the threat of falsification in principle.

Knowledge a priori and knowledge a posteriori may require quite different handling approaches in KM systems.

Type and token: Another pair of knowledge terms are type and token, as suggested by Charles Sanders Peirce (1839-1914), or schema and actualization, or generic acts and individual acts, according to Georg Henrik von Wright (1916-2003). The type level represents the general structure of anything, whereas the token level applies to a single case (Müller-Merbach, 2007c).

Example 2. A general system of linear equations could be written at the type level such as: $\boldsymbol{Ax} = \boldsymbol{b}$ (in matrix notation). At the token level, a case could perhaps read:

$$\begin{aligned} 5x_1 + 7x_2 &= 41 \\ 3x_1 + 2x_2 &= 18 \\ (\text{Solution} : x_1 = 4, x_2 &= 3) \end{aligned}$$

Example 3. A balance sheet and a profit and loss account of an enterprise would at the type level just refer to the formal structure of these documents. At the token level, the concrete numbers of a particular enterprise and a particular year would be presented.

The two levels are mutually interdependent. Familiarity with the type level is required as a frame to understand and to cope with any actual case and would include the general rules. Familiarity with cases is necessary in order to collect experience.

The four causes of Aristotle: Aristotle (384/3-322/1 BC) created a four-part system of describing things of any kind: (i) material cause, (ii) formal cause, (iii) efficient cause, and (iv) final cause. It is surprising how well the four causes are suitable to describe anything. Bertrand Russell (1872-1970) gives an example: "Let us take again the man who is making a statue. The material cause of the statue is the marble, the formal cause is the essence of the statue to be produced, the efficient cause is the contact of the chisel with the marble, and the final cause is the end that the sculptor has in view" (Russell 1979; p. 181).

These four causes can well be used for KM documentation, e.g. product documents: (i) The material cause is represented by the bill-of-material documents. (ii) The formal cause is represented by the construction drawings. (iii) The efficient cause is represented by the production process documents. (iv) The final cause is represented by the user documents.

Aristotle's documentation system is universal. It is only surprising that this four causes schema is not widely used. It is quoted in many philosophical publications, but almost nowhere in connection with KM, product management, database design, or business administration in general.

Information, knowledge, and opinion: Mittelstrass (1972) distinguishes between information, knowledge, and opinion. So does Müller-Merbach (2006b). The terminology in this field is much broader and includes e.g.: data, news, intelligence, prudence, comprehension, ability, judgment, sapience, inspiration, insight, understanding, wisdom, and many more. Any discussion of such terms should (i) make clear what the author means by the single terms, and (ii) help the author to use the terms in a consistent, not in an arbitrarily changing sense. This indicates a difficulty at the present state: The corresponding technical terms of KM are used quite arbitrarily and with changing content.

It does not seem to be necessary that everyone uses a unique terminology. It could perhaps be sufficient if every contributor (author or speaker) provides the reader (or listener) with a clear statement on how the contributor's terms should be

understood; see Eysenck (1979) where he distinguishes between definitions of real objects and concepts. Real objects always have concrete things to be compared with, while concepts do not have concrete things to be compared with. Thus, everyone has a personal understanding of concepts such as live, love, intelligence (Eysenck's example), knowledge, etc. To have a productive discussion about any such concept, there is a need to clarify the individual understandings of the concepts.

The distinction between information, knowledge, and opinion seems to be quite useful:

- Information may be understood as objective properties of something. It can be written down on paper or stored in computer systems;
- Knowledge may be understood as something like information plus understanding or insight. Knowledge may be understood as a property that only human beings can have. A book or an information system may be designed with the input of the knowledge of many human beings. However, the book or the computer itself does not have knowledge in this sense. Human beings can learn and therefore develop knowledge by reading books. These human beings develop knowledge from information;
- However, neither information nor knowledge initiates much action. It is one's opinion why one votes for candidate A, B, or C in political elections. It is one's opinion why one has a preference for car F, G, or H. It is one's opinion to either listen to the music of Debussy or Duke Ellington or Johnny Cash.

This is the reason why the author emphasizes the combination of information management, knowledge management, and opinion management (Müller-Merbach, 2006b).

Only living beings can have knowledge and can have opinions.

Five a posteriori types of knowledge: Knowledge can be quite different, due to its origin. Five types of origin are distinguished here (Fig. 1, the horseshoe of *a* posteriori knowledge):

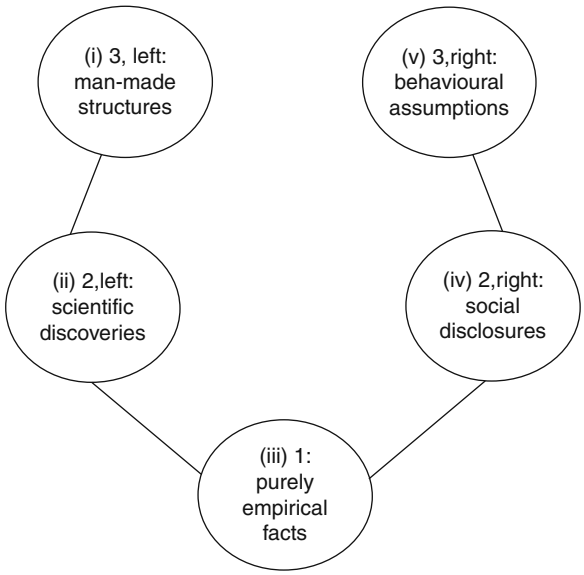
(i) *Man-made structures:* Many structures which are designed by man have to be dealt with. These can be laws, contracts, statutes, machines, (mathematical) models, constructions (and their drawings), bill-of-material-graphs etc. It is a KM task to design formal models, databases for those structures and to implement them into information systems. It is also a KM task to make these structures understandable to human beings. This may require a corresponding course of study, be it quantitative economics or engineering or product design etc.

(ii) *Scientific discoveries:* In other cases, knowledge is based on scientific understanding, e.g., on gravitation, on vibration and oscillation. This may require a scientific background.

(iii) *Purely empirical facts:* In other cases, one may have to deal with purely empirical facts, such as statistical distributions (e.g., of population).

(iv) *Social disclosures:* In other cases, there may be a need of knowledge based on the social sciences, e.g. the influence of prices on demand. Such information is often vague, i.e., much less reliable than the results of scientific discoveries. But, such disclosures may provide empirical evidence to start from.

(v) *Behavioral assumptions:* In other cases, one may depend on assumptions about people



Knowledge Management, Fig. 1 Horseshoe of five types of a posteriori (i.e. empirical) knowledge, organized in three levels (facts only – natural and social sciences – decisions by man) and two branches (*left*: precise, positive; *right*: vague, conditional), (Müller-Merbach (2007b)

and their behavior, their desires, their objectives, their needs, their actions and reactions, their motivation, etc. This kind of knowledge may be related to results of the liberal arts and social sciences.

Quite frequently, an interdisciplinary mix of knowledge is required for the solution of problems. Therefore, KM may require an interdisciplinary understanding.

Utopia of Total Knowledge Management: Any KM in practice would be far away from a collection of all the knowledge of the world. Such a collection could hardly be handled, and it could hardly be designed. However, it could be challenging to think about the structure and the size of such a collection. It would have to be much more than a collection of all available encyclopedia. It would have to be a network in which all the related entries would be connected.

Total KM is utopia today and will remain utopia.

KM in practice should start with tiny islands of KM, and then perhaps spread out.

Large-scale KM systems are extremely rare, in contrast to large-scale information systems. It is recommended to start KM with small projects.

Professional organization of knowledge management: KM is supported by several professional societies. In some countries, KM working groups or KM divisions were founded within computer science societies and OR societies, as well as in other disciplines. In addition, the Knowledge Management Professional Society (KMPro) is an independent organization that was founded in the Washington, D.C. in 2001.

There are at least five independent KM professional journals:

- *Journal of Knowledge Management*, Quarterly, English, U.K., since 1997.
- *Journal of Knowledge Management Practice*, Quarterly, English, Canada, since 1998.
- *Journal of Information & Knowledge Management*, Quarterly, English, Singapore, since 2002.
- *Electronic Journal of Knowledge Management*, English, U.K., since 2003.
- *Knowledge Management Research & Practice*, Bimonthly, English, U.K., since 2003.

An *Encyclopedia of Knowledge Management* was published in 2006 (Schwartz 2006).

Many reviews and surveys of KM have been published, such as a review of the main approaches to KM (Lloria 2008), and survey on concept maps (Martin and Wrice 2009). One of the earliest foundations of KM is the frequently quoted book by Nonaka and Takeuchi, (1995). A broad and practical introduction to KM is Nissen (2006).

References

Angeles, P. A. (1992). *The Harper Collins dictionary of philosophy* (2nd ed.). New York: Harper Perennial.

Bacon, F. (1605). *The advancement of learning*. London.

Durant, W. (1926). *The lives and opinions of the world's greatest philosophers*. New York: Pocket Books.

Eysenck, H. J. (1979). *The structure and measurement of intelligence*. Berlin: Springer.

Lloria, M. B. (2008). A review of the main approaches to knowledge management. *Knowledge Management Research and Practice, 6*(1), 77–89.

Martin, N., & Wrice, J. (2009). Concept maps: a technique for assessing knowledge manager learning needs. *Knowledge Management Research and Practice, 7*(2), 152–161.

Mittelstrass, J. (1972). "Der Verlust des Wissens" (The loss of knowledge). In J. Mittelstrass (Ed.), *Leonardo-Welt*. Frankfurt/Main: Suhrkamp.

Müller-Merbach, H. (2004–2008). Philosophers and knowledge management. *Knowledge Management Research & Practice (KMRP)*, 2–6.

Müller-Merbach, H. (2005a). Francis Bacon's praise: Knowledge, the source of power. *Knowledge Management Research and Practice, 3*(1), 45–46.

Müller-Merbach, H. (2005b). How to structure knowledge: Aristotle and the four causes. *Knowledge Management Research and Practice, 3*(3), 183–184.

Müller-Merbach, H. (2006a). Eysenck's advice: Why and when to define knowledge. *Knowledge Management Research and Practice, 4*(3), 250–251.

Müller-Merbach, H. (2006b). Mittelstrass' triad: Information, knowledge, opinion. *Knowledge Management Research and Practice, 4*(4), 331–332.

Müller-Merbach, H. (2007a). Kant's two paths of knowledge creation: *a priori* vs. *a posteriori*. *Knowledge Management Research and Practice, 5*(1), 64–65.

Müller-Merbach, H. (2007b). A system of five object types of *a posteriori* knowledge. *Knowledge Management Research and Practice, 5*(2), 151–153.

Müller-Merbach, H. (2007c). Type and token, schema and actualisation: Hierarchies of knowledge. *Knowledge Management Research and Practice, 5*(3), 222–223.

Müller-Merbach, H. (2008). Knowledge management: a program for education and leadership (a survey of the series). *Knowledge Management Research and Practice, 6*(4), 350–356.

Nissen, M. E. (2006). *Harnessing knowledge dynamics: Principled organizational knowing & learning*. Hershey, PA: IRM Press.

Nonaka, I., & Takeuchi, H. (1995). *The knowledge creating company*. Oxford: Oxford University Press.
Russell, B. (1979). *A history of western philosophy* (2nd ed.). London: Unwin Hyman.
Schwartz, D. (Ed.) (2006). *"Encyclopedia of knowledge management"*. Hershey, PA. (reviewed by Edwards, J. S. (2007) *Knowledge Management Research and Practice* 5, (4), 315–316).

König's Theorem

▶ Hungarian Method

Königsberg Bridge Problem

▶ Chinese Postman Problem
▶ Combinatorics

Kruskal's Algorithm

A procedure for finding a minimum spanning tree in a network. The method selects the lowest cost arcs in sequence, while ensuring that no cycles are allowed. Ties are broken arbitrarily. For a network with n nodes, the process stops when $n - 1$ arcs are selected.

See

▶ Greedy Algorithm
▶ Minimum Spanning Tree Problem
▶ Prim's Algorithm

Kuhn-Tucker (KT) Conditions

▶ Karush-Kuhn-Tucker (KKT) Conditions

Kullback-Leibler Divergence

Measure of difference between two probability distributions, given by

$$\int f(x) \log \frac{f(x)}{g(x)} dx \text{ for PDFs,}$$

and by

$$\sum f(x) \log \frac{f(x)}{g(x)} \text{ for PMFs,}$$

or more generally for two probability measures P and Q, by

$$\int_{\omega \in \Omega} \log \frac{dP(\omega)}{dQ(\omega)} dP(\omega)$$

for P absolutely continuous with respect to Q, where dP/dQ is the Radon-Nikodym derivative.

Also called relative entropy, cross entropy, information divergence. Because it is neither symmetric nor does it satisfy the triangle inequality, it is not a true distance metric.

See

▶ Cross-Entropy Method
▶ Radon-Nikodym Derivative

L

Lack of Memory

- ▶ Exponential Arrivals
- ▶ Markov Processes
- ▶ Markov Property
- ▶ Memoryless Property
- ▶ Poisson Process
- ▶ Queueing Theory

Lagrange Multipliers

The multiplicative, linear-combination constants that appear in the Lagrangian of a mathematical programming problem. They are generally dual variables if the dual exists, so-called shadow prices in linear programming, giving the rate of change of the optimal value with constraint changes, under appropriate conditions.

See

- ▶ Lagrangian Function
- ▶ Nonlinear Programming

Lagrangian Decomposition

- ▶ Integer and Combinatorial Optimization
- ▶ Lagrangian Relaxation

Lagrangian Function

The general mathematical-programming problem of minimizing $f(\boldsymbol{x})$ subject to a set of constraints $\{g_i(\boldsymbol{x}) \leq b_i\}$ has associated with it a Lagrangian function defined as $L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_i \lambda_i[g_i(\boldsymbol{x}) - b_i]$, where the components λ_i of the nonnegative vector $\boldsymbol{\lambda}$ are called Lagrange multipliers. For a primal linear-programming problem, the Lagrange multipliers can be interpreted as the variables of the corresponding dual problem.

See

- ▶ Lagrangian Relaxation
- ▶ Nonlinear Programming

Lagrangian Relaxation

Monique Guignard
University of Pennsylvania, Philadelphia, PA, USA

Introduction

Many practical optimization problems include decision variables that are integer or 0-1. These problems, called mixed-integer programming problems or MIP for short, are in general difficult to solve, and there have been traditionally two classes of approaches to solve them: branch-and-bound or enumeration, and heuristic methods, either ad hoc or generic. Broadly speaking, branch-and-bound methods construct a tree, usually

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

binary, that allows the systematic exploration of all integer or 0-1 combinations of the discrete variables. Logical considerations and/or bounds on the optimal value computed as one moves down the tree may allow the pruning of a branch and backtracking to its root because one discovers that it would lead to infeasibilities or inferior solutions. Typically, bounds are obtained by solving a simpler, relaxed, optimization problem, most of the time the continuous relaxation of the MIP problem in which integer or 0-1 variables are allowed to take on fractional values. Heuristics, on the other hand, search for better and better feasible integer solutions, and do not usually compute bounds on the optimum, and therefore, even though they are getting more and more sophisticated and excel at finding optimal or near optimal solutions, cannot guarantee the quality of the solutions found.

Lagrangian relaxation stands somehow at the crossroads of both approaches. More powerful in terms of bound quality than the continuous relaxation, it also produces partially infeasible, but integer, solutions. These can usually serve as excellent starting points for specialized heuristics, referred to as Lagrangian heuristics. Contrary to the general heuristics mentioned above, given that one has found a bound called the Lagrangian bound, one knows whether the best solution found is good enough, or if it requires further investigation.

There is an enormous amount of literature devoted to the theory and applications of Lagrangian relaxation, starting with the seminal papers of Held and Karp (1970, 1971) and of Geoffrion (1974), although one could trace it back to earlier sources, for instance Everett's multipliers work (1963). Some early guides include (Fisher 1981, 1985).

Some of the questions to be addressed: Why use Lagrangian relaxation for integer programming problems? How does one construct a Lagrangian relaxation? What tools are there to analyze the strength of a Lagrangian relaxation? Are there more powerful extensions than standard Lagrangian relaxation, and when should they be used? Why is it that one can sometimes solve a strong Lagrangian relaxation by solving trivial subproblems? How does one compute the Lagrangian relaxation bound? Can one take advantage of Lagrangian problem decomposition? Does the strength of the model used make a difference in terms of bounds? Can one strengthen Lagrangian relaxation bounds by cuts, either kept or dualized? How can one design a Lagrangian heuristic? Can one achieve better results by remodeling the problem prior to doing Lagrangian relaxation?

The problems considered here have some integer variables, linear objective functions and constraints, and everything described below applies to maximization as well as minimization problems via the trivial sign transformations:

$$\text{Max } \{f(x)|x \in V\} = -\text{Min } \{-f(x)|x \in V\}.$$

Notation

If (P) is an optimization problem,

FS(P) denotes	the set of feasible solutions of problem (P)
OS(P)	the set of optimal solutions of problem (P)
v(P)	the optimal value of problem (P)
u^k, s^k, etc.,	the value of u, s, etc., used at iteration k
x^T	the transpose of x
x^k	the k^{th} extreme point of some polyhedron (see context)
$x^{(k)}$	a solution found at iteration k.
$\subset$	denotes strict inclusion.
Co(V)	denotes the convex hull of set V.

Relaxations of Optimization Problems

Geoffrion (1974) formally defines a relaxation of a generic minimization problem as follows.

Definition 1. *Problem* (RP_{min}): Min $\{g(x)|x \in W\}$ *is a relaxation of problem* (P_{min}): Min $\{f(x) \mid x \in V\}$ *if and only if* (i) *the feasible set of* (RP_{min}) *contains that of* (P_{min}), *and* (ii) *over the feasible set V of* (P_{min}), *the objective function of* (RP_{min}) *dominates* (*is better than*) *that of* (P_{min}), *i.e.*, $\forall\, x \in V$, $g(x) \leq f(x)$.

It clearly follows that $v(RP_{min}) \leq v(P_{min})$, in other words ($RP_{min}$) is an optimistic version of (P_{min}): it has more feasible solutions than (P_{min}), and for feasible solutions of (P_{min}), its own objective function is at least as good as (smaller than or equal to) that of (P_{min}), thus it has a smaller minimum.

Lagrangian Relaxation (LR)

In the rest of the note, (P) is assumed to be of the form $\text{Min}_x\{fx|Ax \leq b, Cx \leq d, x \in X\}$, where X contains the integrality restrictions on x, i.e. $X = \mathbb{R}^{n-p} \times \mathbb{Z}^p$, or $X = \mathbb{R}^{n-p} \times \{0,1\}^p$. Let I($X$) be the set of the p indices of x restricted to be integer (or binary). The constraints $Ax \leq b$ are assumed complicating, in the sense that, without them, problem (P) would be much simpler to solve. The constraints $Cx \leq d$ (possibly empty) will be kept, together with X, to form the Lagrangian relaxation of (P) as follows. Let λ be a nonnegative vector of weights, called Lagrangian multipliers.

Definition 2. *The Lagrangian relaxation of* (P) *relative to the complicating constraints* $Ax \leq b$, *with nonnegative Lagrangian multipliers* λ, *is the problem* (LR_λ) $\text{Min}_x\{f\ x + \lambda(Ax - b)|Cx \leq d, x \in X\}$.

Notice that (LR_λ) is still an integer programming problem, so its solutions, unlike those of the continuous relaxation, are integer solutions. However they need not be feasible solutions of (P), as they may violate some, or all, of the complicating constraints $Ax \leq b$, which are not enforced any more. In (LR_λ), the slacks of the complicating constraints $Ax \leq b$ have been added to the objective function with weights λ. One says that the constraints $Ax \leq b$ have been dualized. (LR_λ) is a relaxation of (P), since (i) FS (LR_λ) contains FS(P), and (ii) for any x feasible for (P), and any $\lambda \geq 0$, $fx + \lambda(Ax - b)$ is less than or equal to fx (i.e., not worse, since it is a minimization problem). It follows that $v(\text{LR}_\lambda) \leq v(\text{P})$, for all $\lambda \geq 0$, i.e., the optimal value $v(\text{LR}_\lambda)$, which depends on λ, is a lower bound on the optimal value of (P).

Definition 3. *The problem of finding the tightest Lagrangian lower bound on* v(P) *, i.e.*, (LR) $\text{Max}_{\lambda \geq 0}$ $v(\text{LR}_\lambda)$, *is called the Lagrangian dual of* (P) *relative to the complicating constraints* $Ax \leq b$. v(LR) *is called the Lagrangian relaxation bound, or simply the Lagrangian bound.*

Let (LP) denote the linear programming relaxation of problem (P). By LP duality, any Lagrangian relaxation bound is always at least as good as the LP bound, i.e., v(P), never worse. Notice also that (LR) is a problem in the dual space of the Lagrangian multipliers, whereas (LR_λ) is a problem in x, i.e., in the primal space.

Feasible Lagrangian solution

Let $x(\lambda)$ denote an optimal solution of (LR_λ) for some $\lambda \geq 0$, then $x(\lambda)$ is called a Lagrangian solution. One may be tempted to think that a Lagrangian solution $x(\lambda)$ that is feasible for the integer problem (i.e., that satisfies the dualized constraints) is also optimal for that problem. In fact this is generally not the case. What is true is that the optimal value of (P), v(P), lies in the interval between $fx(\lambda)+\lambda[Ax(\lambda)-b]$ and $fx(\lambda)$, where $fx(\lambda)$ is the value of a feasible solution of (P), thus an upper bound on v(P), and $fx(\lambda)+\lambda[Ax(\lambda)-b]$ is the optimal value of the Lagrangian problem (LR_λ), thus a lower bound on v(P). If, however, complementary slackness holds, i.e., if $\lambda[Ax(\lambda)-b]$ is 0, then $fx(\lambda)+\lambda[Ax(\lambda)-b] = v(\text{P}) = fx(\lambda)$, and $x(\lambda)$ is an optimal solution for (P).

Theorem 1. (1) *If* $x(\lambda)$ *is an optimal solution of* (LR_λ) *for some* $\lambda \geq 0$, *then* $fx(\lambda)+\lambda[Ax(\lambda)-b] \leq v(\text{P})$. *If in addition* $x(\lambda)$ *is feasible for* (P), *then* $fx(\lambda) +\lambda[Ax(\lambda)-b] \leq v(\text{P}) \leq fx(\lambda)$.

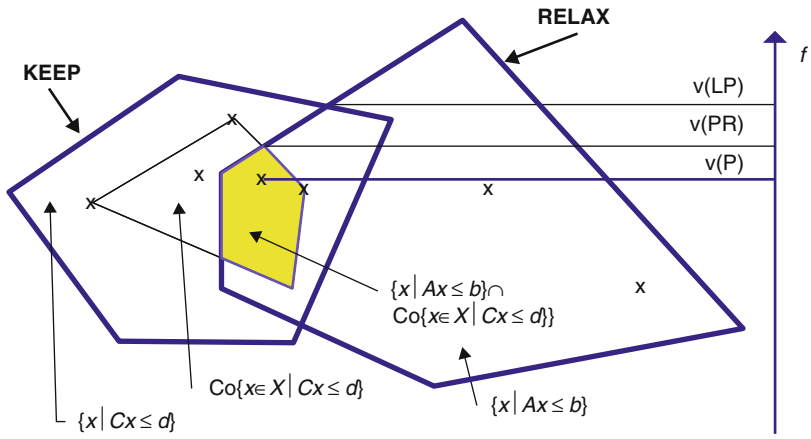
(2) *If in addition* $\lambda[Ax(\lambda)-b] = 0$, *then* $x(\lambda)$ *is an optimal solution of* (P), *and* $v(\text{P}) = fx(\lambda)$.

Remarks. Notice first that (2) is a sufficient condition of optimality, but it is not necessary. I.e., it is possible for a feasible $x(\lambda)$ to be optimal for (P), even though it does not satisfy complementary slackness. If the constraints that are dualized are equality constraints, and if $x(\lambda)$ is feasible for (P), complementary slackness holds automatically, thus $x(\lambda)$ is an optimal solution of (P), with $v(\text{P}) = fx(\lambda)$.

Geometric Interpretation

The following theorem, from (Geoffrion 1974), is probably what sheds most light on Lagrangian relaxation. It gives a geometric interpretation of the Lagrangian dual problem in the x- space, i.e., in the primal space, and this permits an in-depth study of the strength of specific Lagrangian relaxation schemes.

Theorem 2. *The Lagrangian dual* (LR) *is equivalent to the primal relaxation* (PR) $\text{Min}_x\{fx|Ax \leq b, x \in Co\{x \in X|Cx \leq d\}\}$, *in the sense that* v(LR) = v(PR) (Fig. 1).



Lagrangian Relaxation, Fig. 1 Geometric interpretation of Lagrangian relaxation

This result is based on LP duality and properties of optimal solutions of linear programs. Remember though that this result may not be true if the constraint matrices are not rational.

The following important definition and results follow from this geometric interpretation.

Definition 4. *One says that* (LR) *has the Integrality Property* (IP *for short*) *if* $\mathrm{Co}\{x \in X | Cx \leq d\} = \{x \in \mathbb{R}^n | Cx \leq d\}$.

If (LR) has the Integrality Property, then the extreme points of $\{x \in \mathbb{R}^n | Cx \leq d\}$ are in X. The unfortunate consequence of this property, as stated in the following corollaries, is that such an LR scheme cannot produce a bound stronger than the LP bound. Sometimes, however, this is useful anyway because the LP relaxation cannot be computed easily. This may be the case for instance for some problems with an exponential number of constraints that can be relaxed anyway into easy to solve subproblems. The traveling salesman problem is an instance of a problem which contains an exponential number of (subtour elimination) constraints. A judicious choice of dualized constraints leads to Lagrangian subproblems that are 1-tree problems, thus eliminating the need to explicitly write all the subtour elimination constraints (Held and Karp 1970, 1971).

Here are the two corollaries of Theorem 2 that explain the important role played by the Integrality Property.

Corollary 1. *If* $\mathrm{Co}\{x \in X | Cx \leq d\} = \{x \in \mathbb{R}^n | Cx \leq d\}$, *then* $v(\mathrm{LP}) = v(\mathrm{PR}) = v(\mathrm{LR}) \leq v(\mathrm{P})$.

In that case, the Lagrangian relaxation bound is *equal* to (cannot be better than) the LP bound.

Corollary 2. *If* $\mathrm{Co}\{x \in X | Cx \leq d\} \subset \{x \in \mathbb{R}^n | Cx \leq d\}$, *then* $v(\mathrm{LP}) \leq v(\mathrm{PR}) = v(\mathrm{LR}) \leq v(\mathrm{P})$, *and it may happen that the Lagrangian relaxation bound is strictly better than the LP bound.*

Unless (LR) does *not* have the Integrality Property, it will *not* yield a stronger bound than the LP relaxation. It is thus important to know if all vertices of the rational polyhedron $\{x \in \mathbb{R}^n | Cx \leq d\}$ are in X.

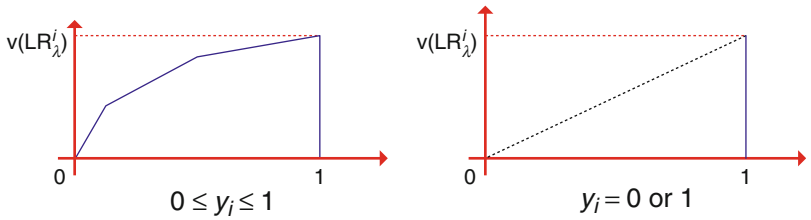
Easy-to-Solve Lagrangian Subproblems

It may happen that Lagrangian subproblems, even though in principle hard to solve because they do not have the Integrality Property, are in fact much easier to solve through some partial decomposition; they can sometimes even be solved in polynomial time, by exploiting their special structure. It is of course important to be able to recognize such favorable situations, especially if one can avoid using Branch-and-Bound to solve them. It should be noted that these favorable cases do not in general occur naturally, but only after some constraint(s) have been dualized, due to a weakening of the original links between continuous and integer variables.

One case is due to what is sometimes called the Integer Linearization Property (or ILP for short) for mixed 0-1 problems.

Integer Linearization Property

Geoffrion (1974) and Geoffrion and McBride (1978) described and used this important property of *some* Lagrangian subproblems. W.l.o.g., assume that all



Lagrangian Relaxation, Fig. 2 Integer linearization property

variables are indexed by $i \in I$, and maybe by some additional indices, and that some of the 0-1 variables are called y_i. If, except for constraints containing only these 0-1 variables y_i, the Lagrangian problem, say, (LR_λ), has the property that the value taken by a given y_i decides alone the fate of all other variables containing the same value of the index i – that usually means that if variable y_i is 0, all variables in its family are 0, and if it is 1, they are solutions of a subproblem – one may be able to reformulate the problem in terms of the variables y_i only. Often, but not always, when this property holds, it is because the Lagrangian problem, after removal of all constraints containing only the y_i's – call it (LRP_λ), for partial problem – decomposes into one problem (LRP^i_λ) for each i, i.e., for each 0-1 variable y_i. The use of this property is based on the following fact. In problem (LRP^i_λ), the integer variable y_i can be viewed as a parameter, however one does know that for the mixed-integer problem (LRP^i_λ), the feasible values of that parameter are only 0 and 1, and one can make use of the fact that there are only two possible values for $v(LRP^i_\lambda)$, the value computed for y_i=1, say v_i $(= v_i.y_i$ for y_i=1), and the value for y_i=0, that is, 0 $(= v_i.y_i$ for y_i=0), which implies that for all possible values of y_i, $v(LRP^i_\lambda) = v_i.y_i$. Hence the name integer linearization, as one replaces a piecewise linear function corresponding to $0 \le y_i \le 1$ by a line through the points (0, 0) and (1, v_i) (Fig. 2).

One may in such cases obtain LR bounds much tighter than the LP bounds, even though the subproblems are trivial to solve.

Constructing a Lagrangian Relaxation

There are often many ways in which a given problem can be relaxed in a Lagrangian fashion. A few standard ones are listed here, mostly to point out that often some reformulation prior to relaxation can help, and that for many complex models, intuition and some understanding of the constraint interactions may suggest ingenious and efficient relaxation schemes.

(1) **One can isolate an interesting subproblem and dualize the other constraints**.

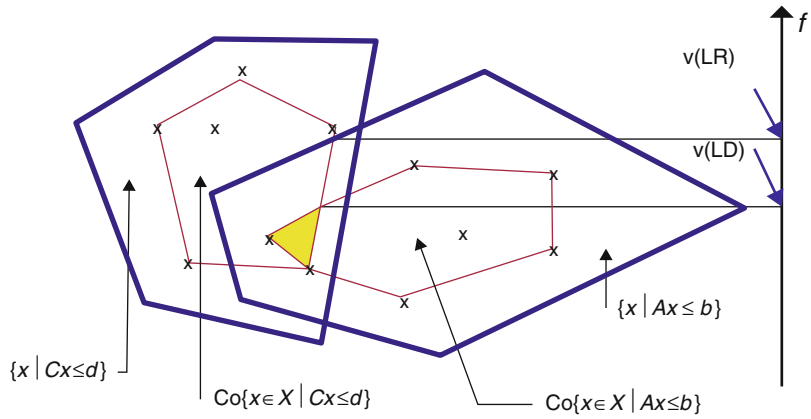
This is the most commonly used approach. It has the advantage that the Lagrangian subproblems are interesting (in the sense usually that they have a special structure that can be exploited) and there may even exist specialized algorithms for solving them efficiently.

(2) **If there are two (or more) interesting subproblems with common variables, one can split these variables first, then dualize the copy constraint.**

This is called Lagrangian decomposition (LD) (Soenen 1977), variable splitting (Näsberg et al. 1985), or variable layering (Glover and Klingman 1988). One must first reformulate the problem using variable splitting, in other words, one must rename the variables in part of the constraints as if they were independent variables. Problem (P): $\mathrm{Min}_x\{f\ x | Ax \le b, Cx \le d, x \in X\}$ is clearly equivalent to problem (P′): $\mathrm{Min}_{x,y}\{f\ x | Ax \le b, x \in X, Cy \le d, y \in X, x = y\}$, in the sense that they have equal optimal values (but notice that they have different variable spaces). In addition if x^* is an optimal solution of (P), then the solution $(x, y) \equiv (x^*, x^*)$ is optimal for (P′), and if (x^*, y^*) is an optimal solution of (P′) with $x^* = y^*$, then x^* is optimal for (P). One dualizes the copy constraint $x = y$ in (P′) with multipliers λ, this separates the problem into an x-problem and a y-problem: (LD_λ) $\mathrm{Min}_{x,y}\{fx + \lambda(y - x) | Ax \le b, x \in X,\ Cy \le d,\ y \in X\} = \mathrm{Min}_x\{(f - \lambda)x \mid Ax \le b,\ x \in X\} + \mathrm{Min}_y\{\lambda y \mid Cy \le d, y \in X\}$.

This process creates a staircase structure, and thus decomposability, in the model. Notice that here λ is not required to be nonnegative.

Remember also that when one dualizes equality constraints, a feasible Lagrangian solution is



Lagrangian Relaxation, Fig. 3 Geometric interpretation of Lagrangean decomposition

automatically optimal for the original integer programming problem. The copy constraint being an equality constraint, if both Lagrangian subproblems have the same optimal solution, that solution is optimal for the IP problem.

Guignard and Kim (1987) showed that the LD bound can strictly dominate the LR bounds obtained by dualizing either set of constraints:

Theorem 3.

$$\begin{aligned} &\textit{If } v(\mathrm{LD}) = \mathrm{Max}_{\lambda}[\mathrm{Min}_{x}\{(f-\lambda)x|\ Ax \le b, x \in X\} \\ &\qquad + \mathrm{Min}\{\lambda y|Cy\ \le\ d, y \in X\}] \textit{ then} \\ &\qquad v(\mathrm{LD}) = \mathrm{Min}\ \{fx \overset{y}{|}\, x \in \mathrm{Co}\{x \in X|Ax \le b\} \\ &\qquad \cap\, \mathrm{Co}\{x \in X|Cx \le d\}. \end{aligned}$$

This new geometric interpretation is demonstrated in Fig. 3.

Corollary 3.

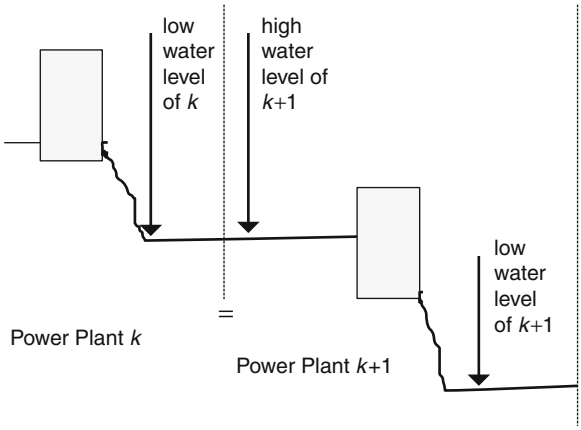
- *If one of the subproblems has the Integrality Property, then v(LD) is equal to the better of the two LR bounds corresponding to dualizing either* $Ax \le b$ *or* $Cx \le d$.
- *If both subproblems have the Integrality Property, then* $v(LD) = v(LP)$.

A very important application of the splitting variable scheme can be found in stochastic optimization, when the uncertainty is represented by 2-stage or multistage scenario trees. The non-anticipativity constraints (or NAC) must be satisfied by the variables attached to the scenario groups or nodes in the tree. Splitting variables in the NAC and dualizing the copy constraints produces a Lagrangean decomposition of the Deterministic Equivalent Model. See Escudero (2009) and Birge and Louveaux (2011), among others.

Occasionally the variable splitting will correspond to a physical split of one of the problem's decision variables. This is illustrated by the following example.

Example 1. Guignard and Yan (1993) described the following problem and scheme for a hydroelectric power management problem.

Electric utility production planning is the selection of power generation and energy efficiency resources to meet customer demands for electricity over a multi-period time horizon. The project described in the paper is a real-world hydropower plant operations management problem of a dispatch type. The system consists of a chain of 10 consecutive hydropower plants separated by reservoirs and falls with 23 identical machines installed to generate electric power. Specifically there are two machines installed in eight power plants (plants 1, 2, 3, 4, 5, 6, 7, and 10), three machines in one power plant (plant 8) and four machines in the last power plant (plant 9). Each machine has two or four work parts for producing electric power, according to different water throughput. Since demand for electric power varies with different time periods, power plant managers must make optimal decisions concerning the number of machines that should be operated in each power plant during each time period. Managing the power generation requires decisions concerning water



Lagrangian Relaxation, Fig. 4 Lagrangian decomposition splits the water level

releases at each plant k in each time period. A period is two hours. The model (which is confidential) was constructed by an independent consulting firm. This results in a large mixed-integer program. The problem is complex, with 2,691 variables, 384 of which are binary, and 12,073 constraints. The firm had tried to solve the problem for the utility company with several of the best MIP software packages available, with help from the software companies themselves. Yet they did not succeed. Guignard and Yan repeated the tests with several solvers running under GAMS, on several RISC systems, also to no avail. The best result after 5 days and six hours on an HP workstation was a bracket [3174.97, 3534.17], i.e., a residual gap of more than 11%.

In order to reduce the complexity of the model, they tried several Lagrangian relaxations and decompositions. One of the decompositions tested consists in "cutting" each reservoir in half (see Fig. 4), i.e. splitting the water level variable in each reservoir, and dualizing the following copy constraint:

$$\text{high water level in } k+1 = \text{low water level in } k.$$

This Lagrangian decomposition produces one power management problem per power plant k. These subproblems do not have a special structure, but are much simpler and smaller than the original problem, are readily solvable by commercial software, and do not have the Integrality Property. They were solved by Branch-and-Bound.

This LD shrinks problem size, and yields Lagrangian bounds much stronger than the LP bounds. In addition the Lagrangian solutions can be modified to provide feasible schedules.

(3) **One can dualize linking constraints**:

After possibly some reformulation, problems may contain independent structures linked by some constraints: $\text{Min}_{x,y}\{f\ x+gy|Ax \le b,\ x \in X,\ Cy \le d,\ y \in Y,\ Ex+Fy \le h\}$. Dualizing the linking constraints $Ex + Fy \le h$ splits the problem into an x-problem and a y-problem. The original problem may only contain x and some reformulation introduces a new variable y, while the relationship between x and y is captured by the new constraints $Ex + Fy \le h$.

Example 2. A production problem over multiple facilities contains constraints related to individual facilities, while the demand constraints link all plant productions. If one dualizes the demand constraints, the Lagrangian problem decomposes into a production problem for each facility, which is typically much easier to solve than the overall problem. If at least one of these subproblems does not have the Integrality Property, this LR may yield a tighter bound than the LP bound. In (Andalaft et al. 2003), a forest company must harvest geographically distinct areas, and dualizing the demand constraints splits the problem into one subproblem per area, which is typically much easier to solve than the overall problem.

(4) **One can sometimes dualize aggregate rather than individual copies of variables.**

Instead of creating a copy y of variable x and introducing y into model (P) by rewriting the constraint $Cx \le d$ as $Cy \le d$, to yield the equivalent model (P′): $\text{Min}_{x,y}\{f\ x|Ax \le b,\ x \in X,\ Cy \le d,\ y \in X,\ x = y\}$, one can also create a problem (P″) equivalent to problem (P) by introducing a new variable y and forcing the constraint $Dy = Cx$. This constraint is in general weaker than the constraint $x = y$. Model (P″) is $\text{Min}_{x,y}\{f\ x|Ax \le b,\ x \in X,\ Dy \le d,\ y \in X,\ Dx = Cy\}$. The LR introduced here dualizes the aggregate copy constraint $Dx = Cy$.

Notice that the copy constraint is an equality constraint, therefore if the Lagrangian subproblems have optimal solutions x and y that satisfy the aggregate copy constraint, i.e., if $Dy = Cx$, then the x- solution is optimal for the IP problem.

Example 3. Consider the bi-knapsack problem

(BKP)$\text{Max}_x\{\sum_i c_i x_i | \sum_i b_i x_i \le m,\ \sum_i d_i x_i \le n, x_i \in \{0,1\}, \forall i\}$.

One can introduce a new variable y, and write $\sum_i b_i x_i = \sum_i b_i y_i$. The equivalent problem is

(BKP')$\text{Max}_{x,y}\{\sum_i c_i x_i | \sum_i b_i y_i \le m,\ \sum_i d_i x_i \le n, \sum_i b_i x_i = \sum_i b_i y_i,\ x_i, y_i \in \{0,1\}, \forall i\}$

and the LR problem is

$$(\text{LR}_\lambda)\text{Max}_{x,y}\left\{\sum_i c_i x_i - \lambda\left(\sum_i b_i x_i - \sum_i b_i y_i\right)\right| \sum_i b_i y_i \le m, \sum_i d_i x_i \le n,\ x_i, y_i \in \{0,1\}, \forall i\}$$
$$= \text{Max}_x\left\{\sum_i (c_i - \lambda b_i) x_i \middle| \sum_i d_i x_i \le n, x_i \in \{0,1\}, \forall i\right\} + \text{Max}_y\left\{\lambda \sum_i b_i y_i) \middle| \sum_i b_i y_i \le m,\ y_i \in \{0,1\}, \forall i\right\}.$$

Here λ is a single real multiplier of arbitrary sign. The Lagrangian bound produced by this scheme is in between that of the LP bound and that of the Lagrangian decomposition bound obtained by dualizing $x_i = y_i\ \forall i$. This is similar in spirit to the copy constraints introduced in Reinoso and Maculan (1992).

It would seem natural that a reduction in the number of multipliers should imply a reduction in the quality of the LR bound obtained. This is not always the case, however, as shown in example 4.

Example 4. Chen and Guignard (1998) considered an aggregate Lagrangian relaxation of the capacitated facility location problem. The model uses continuous variables x_{ij} that represent the percentage of the demand d_j of customer j supplied by facility i, and binary variables y_i, equal to 1 if facility i with capacity a_i is operational. The constraint $\sum_j d_j x_{ij} \le a_i y_i$ imposes a conditional capacity restriction on the total amount that can be shipped from potential facility i.

(CPLP)

$\text{Min}_{x,y}\ \Sigma_i \Sigma_j c_{ij} x_{ij} + \Sigma_i f_i y_i$		
s.t. $\Sigma_i x_{ij} = 1$, all j	(D)	*meet 100% of customer demand*
$x_{ij} \le y_i$, all i, j	(B)	*ship nothing if plant is closed*
$\Sigma_i a_i y_i \ge \Sigma_j d_j$,	(T)	*enough plants to meet total demand*
$\Sigma_j d_j x_{ij} \le a_i y_i$, all i	(C)	*ship no more than plant capacity*
$x_{ij} \ge 0$, $y_i = 0$ or 1, all i, j.		

Constraint (T) is redundant, but may help getting tighter Lagrangian relaxation bounds.

The three best Lagrangian schemes are:

(LR) (Geoffrion and McBride 1978)

One dualizes (D) then uses the integer linearization property. The subproblems to solve are one continuous knapsack problem per plant ((C) with $y_i = 1$) and one 0-1 knapsack problem over all plants (constraint (T)). The Lagrangian relaxation bound is tight, and it is obtained at a small computational cost.

(LD) (Guignard and Kim 1987).

Duplicate (T). Make copies $x_{ij} = x'_{ij}$ and $y_y = y'_i$ and use x'_{ij} and y'_i in (C) and in one of the (T)'s. One obtains the split

{(D), (B), (T)} → APLP

{(B), (T), (C)} → this is like in (LR)

This LD bound is tighter than the (LR) bound, but expensive to compute, in particular because of a large number of multipliers.

(LS) (Chen and Guignard 1998).

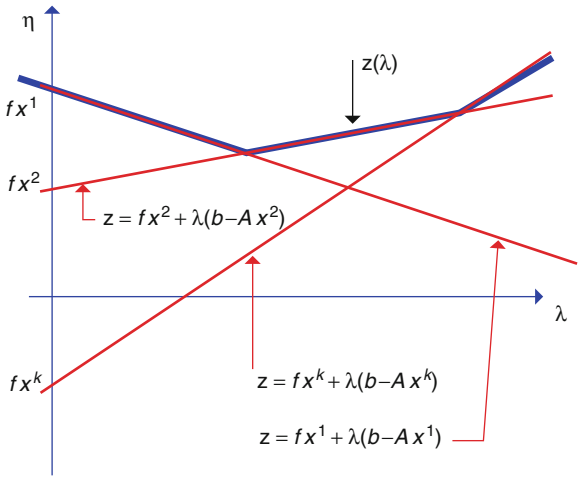
Copy $\sum_j d_j x_{ij} = \sum_j d_j x'_{ij}$ and $y_i = y'_i$ in (C). This yields the same split as (LD), and the same bound. This is very surprising, as it is less expensive to solve (LS) than (LD), in particular because (LS) has far fewer multipliers.

In example 4, creating new copy variables x'_{ij} and y'_i, one can create an LS by dualizing the aggregate (linking) copy constraints $\sum_j d_j x_{ij} = \sum_j d_j x'_{ij}$ and $a_i y_i = a_i y'_i$. Surprisingly, one can prove that the LS bound for this problem is as strong as the LD bound obtained by dualizing individual copies $x_{ij} = x'_{ij}$ and $y_i = y'_i$. This suggests that "aggregating" variables before copying them may be an attractive alternative to Lagrangian decomposition, at least for some problem structures. A more general structure than CPLP is actually described in Chen and Guignard (1998).

Characteristics of the Lagrangian Function

The Lagrangian function $z(\lambda) = v(\text{LR}_\lambda)$ is an implicit function of λ. Suppose that the set $\text{Co}\{x \in X | Cx \le d\}$ is a polytope, i.e., a bounded polyhedron, then there exists a finite family $\{x^1, x^2, \ldots, x^K\}$ of extreme points of $\text{Co}\{x \in X | Cx \le d\}$, i.e., of points of $\{x \in X | Cx \le d\}$, such that $\text{Co}\{x \in X | Cx \le d\} = \text{Co}\ \{x^1, x^2, \ldots, x^K\}$. It then follows that

$$\text{Min}_x\{fx + \lambda(b - Ax) | Cx \le d, x \in X\} = \text{Min}_{k=1,\ldots,K}\{f\ x^k + \lambda(b - Ax^k)\}$$



Lagrangian Relaxation, Fig. 5 The Lagrangean function of a maximization problem

and $z(\lambda)$ is the lower envelope of a family of linear functions of $\lambda, f\ x^k + \lambda\left(b - Ax^k\right)$, k=1,. . .,K, and thus is a concave function of λ, with breakpoints where it is not differentiable, i.e., where the optimal solution of (LR_λ) is not unique. Figure 5 shows a Lagrangian function for the case where (P) is a maximization problem, this (LR) is a minimization problem, and $z(\lambda)$ a convex function of (λ).

A concave function $f(x)$ is continuous over the relative interior of its domain, and it is differentiable almost everywhere, i.e., except over a set of measure 0. At points where it is not differentiable, the function does not have a gradient, but is always has subgradients.

Definition 5. *A vector* $y \in (\mathrm{R}^n)^*$ is *a* subgradient *of a concave function* $f(x)$ *at a point* $x^0 \in \mathrm{R}^n$ *if for all* $x \in \mathrm{R}^n$

$$f(x) - f\left(x^0\right) \leq y \cdot (x - x^0).$$

Definition 6. *The set of all subgradients of a concave function* $f(x)$ *at a point* x^0 *is called the subdifferential of* f *at* x^0 *and it is denoted* $\partial f(x^0)$.

Theorem 4. *The subdifferential* $\partial f(x^0)$ *of a concave function* $f(x)$ *at a point* x^0 *is always nonempty, closed, convex and bounded.*

If the subdifferential of f at x^0 consists of a single element, that element is the gradient of f at x^0, denoted by $\nabla f(x^0)$.

The dual problem (LR) is

$$\begin{aligned}&\mathrm{Max}_{\lambda\ \geq 0} v(\mathrm{LR}_\lambda) = \mathrm{Max}_{\lambda \geq 0} z(\lambda) = \\ (\mathrm{LR})\ &\mathrm{Max}_{\lambda\ \geq 0} \mathrm{Min}_{k=1,\ldots,K}\left\{fx^k + \lambda\left(b - Ax^k\right)\right\} = \\ &\mathrm{Max}_{\lambda\ \geq 0,\eta}\left\{\eta | \eta \leq fx^k + \lambda(b - Ax^k), k = 1, \ldots, K\right\}.\end{aligned}$$

Let λ^* be a minimizer of $z(\lambda)$, $\eta^* = z(\lambda^*)$, λ^k be a current "guess" at λ^*, let $\eta_k = z(\lambda^k)$, and $\mathrm{H}_k = \left\{\lambda | fx^k + \lambda(b - Ax^k) = \eta^k\right\}$ be a level hyperplane passing through λ^k.

- If $z(\lambda)$ is differentiable at λ^k, i.e., if (LR_λ) has a unique optimal solution x^k, it has a **gradient** $\nabla z(\lambda^k)$ at λ^k:

$$\nabla^{\mathrm{T}} z(\lambda^k) = (b - Ax^k) \perp \mathrm{H}_k.$$

- If $z(\lambda)$ is nondifferentiable at λ^k, i.e., if (LR_λ^k) has multiple optimal solutions, the vector $s^k = (b - Ax^k)^{\mathrm{T}}$ is a subgradient of $z(\lambda)$ at λ^k. That vector s^k is orthogonal to H_k.

If one considers the contours $\mathrm{C}\{k\} = \left\{\lambda \in \mathbb{R}_+^m | z(\lambda) \geq a\right\}$, α a scalar, these contours are convex polyhedral sets. See Fig. 6.

Note: A subgradient is not necessarily a direction of increase for the function, even locally, as seen on Fig. 6.

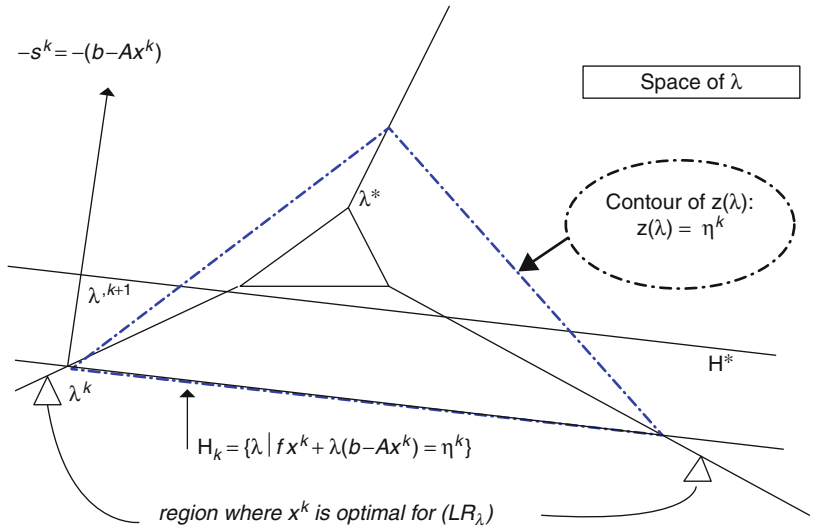
Theorem 5. *The vector* $(b - Ax^k)^{\mathrm{T}}$ is a *subgradient* of $z(\lambda)$ at λ^k.

Primal and Dual Methods to Solve Relaxation Duals

A number of methods have been proposed to solve Lagrangian duals. They are either ad-hoc, like for instance dual ascent methods, or general purpose, usually aiming at solving a generic nonsmooth convex optimization problem. This section reviews the most important approaches.

Subgradient Method

This method was proposed in (Held and Karp 1971). It is an iterative method in which at iteration k, given the current multiplier vector λ^k, a step is taken along a subgradient of $z(\lambda^k)$, then, if necessary, the resulting point is projected onto the nonnegative orthant.



Lagrangian Relaxation, Fig. 6 Contours and subgradient

Let $x^{(k)}$ be an optimal solution of (LR^k_λ). Then $s^k = (b - Ax^{(k)})^T$ is a subgradient of z(λ) at λ^k. If λ^* is an (unknown) optimal solution of (LR), with $\eta^* = z(\lambda^*)$, let λ'^{k+1} be the projection of λ^k on the hyperplane H^* parallel to H_k, defined by

$$H^* = \left\{\lambda | fx^k + \lambda(b - Ax^{(k)}) = \eta^*\right\}.$$

The vector s^k is perpendicular to both H_k and H^*, therefore $\lambda'^{k+1} - \lambda^k$ is a nonnegative multiple of s^k:

$$\lambda'^{k+1} - \lambda^k = \mu s^k, \mu \geq 0.$$

Also, λ'^{k+1} belongs to H^*:

$$f\ x^{(k)} + \lambda'^{k+1}(b - Ax^{(k)}) = \eta^*,$$

therefore $f\ x^k + \mu s^k(b - Ax^{(k)}) = \eta^k + \mu s^k.s^k = \eta^*$
and $\mu = (\eta^* - \eta^k)/||s^k||^2$,
so that $\lambda'^{k+1} = \lambda^k + s^k.\ (\eta^* - \eta^k)\ /||s^k||^2$.

Finally define $\lambda^{k+1} = [\lambda'^{k+1}]^+$, i.e., define the next iterate λ^{k+1} as the projection of λ'^{k+1} onto the nonnegative orthant, as λ must be nonnegative. Given the geometric projections described above, it is clear that λ^{k+1} is closer to λ^* than λ^k, thus the sequence $||\lambda^k - \lambda^*||^2$ is monotone nonincreasing.

Remark. This formula unfortunately uses the unknown optimal value η* of (LR). One can try to use an estimate for that value, but then one may be using either too small or too large a multiple of s^k. If one sees that the objective function values do not improve for too many iterations, one should suspect that η^* has been overestimated (for a maximization problem) and that one is overshooting, thus one should try to reduce the difference η^*-η^k. This can be achieved by introducing from the start a positive factor $\varepsilon_k \in (0,2)$, in the subgradient formula:

$$\lambda^{k+1} = \lambda^k + s^k.\ \varepsilon_k(\eta^* - \eta^k)\ /||s^k||^2,$$

and reducing the scalar ε_k when there is no improvement for too long.

Practical convergence of the subgradient method is unpredictable, sometimes quick and fairly reliable, sometimes erratic. Many authors have studied this problem and have proposed a variety of remedies.

Dual Ascent Methods

In this kind of approach, one takes advantage of the structure of the Lagrangian dual to create a sequence of multipliers that guarantee a monotone increase in Lagrangian function value. This approach had been pioneered by Bilde and Krarup (1967, 1977) for solving approximately the LP relaxation of the uncapacitated facility location problem (UFLP). General principles for developing a successful Lagrangian dual ascent method can be found in (Guignard and Rosenwein 1989).

Constraint Generation Method (Also Called Cutting Plane Method, or CP)

In this method , one uses the fact that $z(\lambda)$ is the lower envelope of a family of linear functions:

$$\begin{aligned}&\mathrm{Max}_{\lambda\geq 0} v(\mathrm{LR}_\lambda) = \mathrm{Max}_{\lambda\geq 0} z(\lambda) =\\ (\mathrm{LR})\ &\mathrm{Max}_{\lambda\geq 0}\mathrm{Min}_{k=1,\ldots,K}\left\{f\,x^k + \lambda\left(b - Ax^k\right)\right\} =\\ &\mathrm{Max}_{\lambda\geq 0,\eta}\left\{\eta | \eta \leq f\,x^k + \lambda\left(b - Ax^k\right), k = 1,\ldots,K\right\}.\end{aligned}$$

At each iteration k, one generates one or more cuts of the form

$$\eta \leq fx^k + \lambda(b - Ax^{(k)}),$$

by solving the Lagrangian subproblem (LR_λ^k) with solution $x^{(k)}$. These cuts are added to those generated in previous iterations to form the current LP master problem:

$$\left(\mathrm{MP}^k\right)\quad \mathrm{Max}_{\lambda\geq 0,\eta}\left\{\eta | \eta \leq fx^{(h)} + \lambda(b - Ax^{(h)}), h = 1,\ldots,k\right\},$$

whose solution is the next iterate λ^{k+1}. The process terminates when $v(\mathrm{MP}^k) = z(\lambda^{k+1})$. This value is the optimal value of (LR).

Column Generation (CG)

(CG) has been used extensively, in particular for solving very large scheduling problems (airline, buses, etc.). It consists in reformulating a problem as an LP (or an IP) whose activities (or columns) correspond to feasible solutions of a subset of the problem constraints, subject to the remaining constraints. The variables are weights attached to these solutions.

There are two aspects to column generation: first, the process is dual to Lagrangian relaxation and to CP. Secondly, it can be viewed as an application of Dantzig and Wolfe's decomposition algorithm (Dantzig and Wolfe 1960, 1961).

Let the $x^k \in \left\{x \in X | Cx^k \leq d\right\}$, $k \in K$, be chosen such that $\mathrm{Co}\left\{x^k\right\} = \mathrm{Co}\{x \in X | Cx \leq d\}$. A possible choice for the x^k's is all the points of $\mathrm{Co}\{x \in X | Cx \leq d\}$ but a cheaper option is all extreme points of $\mathrm{Co}\{x \in X | Cx \leq d\}$.

Problem (P): $\mathrm{Min}_x\{fx | Ax \leq b, Cx \leq d, x \in X\}$ yields the Lagrangian dual (i.e., in the λ-space) problem

$$(\mathrm{LR})\quad \mathrm{Max}_{\lambda\geq 0}\mathrm{Min}_x\{fx + \lambda(Ax - b) | Cx \leq d, x \in X\}$$

which is equivalent to the primal (i.e., in the x-space) problem

$$(\mathrm{PR})\quad \mathrm{Min}_x\{fx | Ax \leq b, x \in \mathbf{Co}\{x \in X | Cx \leq d\}\},$$

which itself can be rewritten as (PR) $\mathrm{Min}_x\left\{f\left(\sum_{k\in K}\mu_k x^k\right)\middle| A\left(\sum_{k\in K}\mu_k x^k\right)x \leq b\right\}$ given $= \mathrm{Min}_x\left\{\sum_{k\in K}\mu_k.(fx^k)\middle|\sum_{k\in K}\mu_k.(Ax^k) \leq b\right\}$, that one can write $x \in \mathrm{Co}\{x \in X | Cx \leq d\}$ as $x = \sum_{k\in K}\mu_k x^k$, with $\sum_{k\in K}\mu_k = 1$ and $\mu_k \geq 0$.

The separation of a problem into a master- and a sub-problem is equivalent to the separation of the constraints into kept and dualized constraints. The columns generated are solutions of integer subproblems that have the same constraints as the Lagrangian subproblems.

The value of the LP relaxation of the master problem is equal to the Lagrangian relaxation bound. The strength of a CG or LR scheme would then seem to be based on the fact that the subproblems do not have the integrality property. It may happen however that such a scheme can be successful at solving problems with the integrality property because it permits the indirect computation of v(LP) when this value could not be computed directly, e.g., because of an exponential number of constraints (Held and Karp 1970, 1971).

One substantial advantage of (CP) or (CG) over subgradient algorithms is the existence of a true termination criterion $v(\mathrm{MP}^k) = z(\lambda^{k+1})$.

Bundle Methods

Lemaréchal (1974) introduced an extension of subgradient methods, called bundle methods, in which past information is collected to provide a better approximation of the Lagrangian function. The standard CP algorithm uses the bundle of the subgradients that were already generated, and constructs a piecewise linear approximation of the Lagrangian function. This method is usually slow and unstable. Three different stabilization approaches have been proposed. At any moment, one has a model representing the Lagrangian function, and a so-called stability center, which should be a reasonable approximation of the true optimal solution.

One generates a next iterate which is a compromise between improving the objective function and keeping close to the stability center. The next iterate becomes the new stability center (a serious step) only if the objective function improvement is "good enough". Otherwise, one has a null step, in which however one improves the function approximation. In addition, this next iterate shouldn't be too far away from the stability center. The three stabilization approaches propose different ways of controlling the amount of move that is allowed. Either the next iterate must remain within a so-called trust region, or one adds a penalty term to the approximation of the function that increases with the distance from the stability center, or one remains within a region where the approximation of the function stays above a certain level (for a maximizaton problem). This proximity measure is the one parameter that may be delicate to adjust in practical implementations. There is a trade-off between the safety net provided by this small move concept, and the possibly small size of the bound improvement.

The Volume Algorithm (VA)

The volume algorithm (Barahona and Anbil 2000), an extension of the subgradient algorithm, can be seen as a fast way to approximate Dantzig-Wolfe decomposition, with a better stopping criterion, and it produces primal as well as dual vectors by estimating the volume below the faces that are active at an optimal dual solution. It has been used successfully to solve large-scale LP's arising in combinatorial optimization, such as set partitioning or location problems.

Subproblem Decomposition

In many cases, the Lagrangian subproblem decomposes into smaller problems, and this means that the feasible region is actually the Cartesian product of several smaller regions. One clear advantage is the reduction in computational complexity for the Lagrangian subproblems: indeed, it is generally much easier to solve 50 problems with 100 binary variables each, say, than a single problem with 5,000 (i.e., 50x100) binary variables.

It also means that in column generation, the columns (i.e., the vectors that are feasible solutions of the kept constraints) decompose into smaller subcolumns, and each subcolumn is a convex combination of extreme points of a small region. By assigning different sets of weights to these convex combinations, one allows mix-and-match solutions, in other words, one may combine a subcolumn for the first subproblem that was generated at iteration 10, say, with a subcolumn for the second subproblem generated at iteration 7, etc. , to form a full size column. If one had not decomposed the problem ahead of time, one may have had to wait a long time for such a complete column to be generated.

By duality, this means that in a cutting plane environment, one can also generate sub-cuts for each subproblem, which amounts to first replacing η by $z - \lambda b$ in

$$\left(\mathrm{MP}^k\right) \mathrm{Max}_{\lambda \geq 0, \eta}\left\{\eta | \eta \leq f\, x^{(h)} + \lambda(b - Ax^{(h)}), h = 1, \ldots, k\right\}$$
$$= \mathrm{Max}_{\lambda \geq 0, z}\left\{z - \lambda b | z \leq (f - \lambda A)x^{(h)}, h = 1, \ldots, k\right\},$$

and then z by a sum of scalars z_l, with $z_l \leq (f^l - \lambda A_l)\, x_l^{(h)}$, where l is the index of the Lagrangian subproblem, f^l, A_l, and $x_l^{(h)}$ are the l^{th} portions of the corresponding submatrices and vectors, and x_l^h is a Lagrangian solution of the l^{th} subproblem found at iteration h, yielding the disaggregated master problem

$$\left(\mathrm{MPD}^k\right) \mathrm{Max}_{\lambda \geq 0, z_l}\left\{\sum\nolimits_l z_l - \lambda b | z_l \leq (f - \lambda A)^l x_l^h, h = 1, \ldots, k\right\}.$$

Example 5. Consider the Generalized Assignment Problem, or GAP (for the minimization case, although it would work in exactly the same way with maximization).

$$\begin{aligned} (\mathrm{GAP})\ \mathrm{Min}\ & \sum\nolimits_i \sum\nolimits_j c_{ij} x_{ij} \\ \text{s.t.}\ & \sum\nolimits_j a_{ij} x_{ij} \leq b_i, \quad \forall i \in I \quad (\mathrm{KP}) \\ & \sum\nolimits_i x_{ij} = 1, \quad \forall j \in J \quad (\mathrm{MC}) \\ & x_{ij} \in \{0, 1\}, \quad \forall i \in I, j \in J. \end{aligned}$$

Its strong Lagrangian relaxation is

$$(\mathrm{LR}_\lambda) \mathrm{Min} \sum\nolimits_{i,j} c_{ij} x_{ij} + \sum\nolimits_j \lambda_j (1 - \sum\nolimits_i x_{ij})$$
$$\text{s.t.} \quad \sum\nolimits_j a_{ij} x_{ij} \leq b_i, \ \forall i \quad (\mathrm{KP})$$
$$= \sum\nolimits_j \lambda_j + \sum\nolimits_i \{\mathrm{Min} \sum\nolimits_j (c_{ij} - \lambda_j) x_{ij} \Big| \sum\nolimits_j a_{ij} x_{ij} \leq b_i, \forall i$$
$$x_{ij} \in \{0, 1\}, \ \forall j\},$$

and (LR) is the maximum with respect to λ of $\mathrm{v}(\mathrm{LR}_\lambda)$.

Let $\mathrm{EP(KP)} = \{x^k | k \in K\}$ be the set of all integer feasible solutions of the constraints (KP), and let $\mathrm{EP(KP_i)} = \{x_i^k | k \in K_i\}$ be the set of all integer feasible solution of the i^{th} knapsack, with $K = \prod_i K_i$. Then a feasible solution of (LR_λ) can be described by $x_{ij} = \sum_{k \in K_i} \mu_k^i x_{ij}^k, \forall i, j$.

The Lagrangian dual is equivalent to the aggregate master problem AMP:

$$\begin{aligned}&(\mathrm{AMP})\ \mathrm{Max}_{\lambda,\zeta}\left\{\zeta | \zeta \le \sum_{i,j} c_{ij} x_{ij}^k + \sum_j \lambda_j (1 - \sum_i x_{ij}^k), \forall k \in K\right\}\\ &= \mathrm{Max}_{\lambda,z}\left\{z + \sum_j \lambda_j | z \le \sum_{ij} (c_{ij} - \lambda_j) x_{ij}^k,\ \forall k \in K\right\}\end{aligned}$$

with the substitution $\zeta = z + \sum_j \lambda_j$.

If one had first written the column generation formulation for the Lagrangian dual, one would naturally have de-coupled the solutions of the independent knapsack subproblems, using the independent sets K_i instead of K, the column generation master problem would have been disaggegated:

$$\begin{aligned}&(\mathrm{DMP})\mathrm{Max}_{\lambda,z} \sum_i z_i + \sum_j \lambda_j\\ &\qquad \text{s.t.} \quad z_i \le \sum_j (c_{ij} - \lambda_j) x_{ij}^k,\ \forall i, \forall k \in K_i\end{aligned}$$

and its dual

$$\begin{aligned}\mathrm{Min}_\mu \sum_{k \in K_i} \sum_{i,j} c_{ij} x_{ij}^k \mu_k^{(i)} \Big| &\sum_{k \in K_i} \sum_i x_{ij}^k \mu_k^{(i)} = 1, \forall j,\\ &\mu_k^i \ge 0, \sum_{k \in K_i} \mu_k^{(i)} = 1, \forall i\},\end{aligned}$$

is clearly the Dantzig-Wolfe decomposition of the primal equivalent

$$(\mathrm{PR}) \quad \mathrm{Min}_x \left\{\sum_{i,j} c_{ij} x_{ij} \Big| \sum_i x_{i,j} = 1,\ x_{ij} \ge 0\right\}$$

of (LR).

Relax-and-Cut

One question that often arises in the context of Lagrangian relaxation is how to strengthen the Lagrangian relaxation bound. One possible answer is the addition of cuts that are currently violated by the Lagrangian solution. It is clear however that adding these to the Lagrangian problem will change its structure and may make it much harder to solve. One possible way out is to dualize these cuts (for a more detailed analysis, see (Guignard 1998)). Remember that dualizing does not mean discarding! The cuts will be added to the set of complicating constraints, and intuitively they will be useful only if the intersection NI (for "new intersection") of the new relaxed polyhedron and of the convex hull of the integer solutions of the kept constraints is "smaller" than the intersection OI (for "old intersection") of the old relaxed polyhedron and of the convex hull of the integer solutions of the kept constraints. This in turn is only possible if the new relaxed polyhedron is smaller than the old one, since the kept constraints are the same in both cases. This has the following implications. Consider a cut that is violated by the current Lagrangian solution:

(1) if the cut is just a convex combination of the current constraints, dualized and/or kept, it cannot possibly reduce the intersection, since every point of the "old" intersection will also satisfy it; so in particular surrogate constraints of the dualized constraints cannot help.
(2) if the cut is a valid inequality for the Lagrangian problem, then every point in the convex hull of the integer points of the kept constraints satisfies it, because every integer feasible solution of the Lagrangian subproblem does;
(3) it is thus necessary for the cut to use "integer" information from both the dualized and the kept constraints, and to remove part of the intersection. (Remember that the Lagrangian solution is an integer point required to satisfy only the kept constraints).

A Relax-and-Cut scheme could proceed as follows:

1. Initialize the Lagrangian multiplier λ.
2. Solve the current Lagrangian problem, let $x(\lambda)$ be the Lagrangian solution. If the Lagrangian dual is not solved yet, update λ. Else end.
3. Identify a cut that is violated by $x(\lambda)$, and dualize it. Go back to 2.

The term Relax-and-Cut was first used by (Escudero et al. 1994). In that paper, a partial description of the constraint set was used, and violated constraints (not cuts) were identified, added to the model and immediately dualized. The idea, if not the name, had actually been used earlier. For instance

in solving TSP problems, subtour elimination constraints were generated on the fly and immediately dualized in Balas and Christofides (1981). The usefulness of constraints is obvious, contrary to that of cuts. A missing constraint can obviously change the problem solution.

Here are examples of cuts that if dualized cannot possibly tighten Lagrangian relaxation bounds.

Non-improving Dualized Cuts: Example for the GAP

Consider again the GAP model.

If one dualizes (MC), the Lagrangian relaxation problem decomposes into one subproblem per *j:*

$$
\begin{aligned}
(\mathrm{LR}_\lambda)\ \mathrm{Min} & \sum_{i,j} c_{ij}x_{ij} + \sum_j \lambda_j\Big(1 - \sum_i x_{ij}\Big) \\
\text{s.t.} \quad & \sum_j a_{ij}x_{ij} \le b_i, \quad \forall i \quad (\mathrm{KP}) \\
= \mathrm{Min}\ & \Big\{\sum_{i,j}(c_{ij} - \lambda_j)x_{ij} + \sum_j \lambda_j \Big| \\
& \sum_j a_{ij}x_{ij} \le b_i, \quad \forall i, \quad x_{ij} \in \{0,\ 1\}, \forall i, j\Big\} \\
= \sum_j \lambda_j + \sum_i & \Big\{\mathrm{Min} \sum_j (c_{ij} - \lambda_j)\ x_{ij} \Big| \\
& \sum_j a_{ij}x_{ij} \le b_i, \quad \forall i\, x_{ij} \in \{0, 1\}, \quad \forall j\Big\}.
\end{aligned}
$$

Thus the i^{th} Lagrangian subproblem is a knapsack problem for the i^{th} machine. After solving all knapsack problems, the solution $x(\lambda)$ may violate some multiple choice constraint, i.e., there may exist some j for which $\sum_i x_{ij} \neq 1$, and as a consequence the condition $\sum_i \sum_j x_{ij} = |J|$ may be violated. Adding this "cut" (it indeed cuts out the current Lagrangian solution!), and immediately dualizing it, does not reduce the intersection, as every point of the old intersection OI already satisfies all multiple choice constraints (MC), i.e., the dualized constraints.

Can kept Cuts Strengthen the Lagrangian Bound?

What happens if one keeps the cuts instead of dualizing them? It is clear that adding these to the Lagrangian problem will change its structure, but it may still be solvable rather easily. The cuts will be added to the set of easy constraints, and intuitively they will be useful only if the intersection NI (for "new intersection") of the relaxed polyhedron and of the new convex hull of the integer solutions of the kept constraints is smaller than the intersection OI (for "old intersection") of the relaxed polyhedron and of the old convex hull of the integer solutions of the kept constraints. This in turn is only possible if the new convex hull polyhedron is smaller than the old one, since the dualized constraints are the same in both cases.

Example 6. Consider again the GAP, and its weak Lagrangian relaxation in which the knapsack constraints (KP) are dualized. One could add to the remaining multiple choice constraints a surrogate constraint of the dualized constraints, for instance the sum of all knapsack constraints, which is obviously weaker than the original knapsack constraints. The Lagrangian problem does not decompose anymore, but its new structure is that of a multiple choice knapsack problem, which is usually easy to solve with specialized software, and much easier than the aggregate knapsack without multiple choice constraints. The above strengthening of the Lagrangian bound is simple, yet potentially powerful.

Lagrangian Heuristics and Branch-and-Price

Lagrangian relaxation provides bounds, but it also generates Lagrangian solutions. If a Lagrangian solution is feasible and satisfies complementary slackness (CS), one knows that it is an optimal solution of the IP problem. If it is feasible but CS does not hold, it is at least a feasible solution of the IP problem and one still has to determine, by BB or otherwise, whether it is optimal. Otherwise, Lagrangian relaxation generates infeasible integer **solutions**. Yet quite often these solutions are nearly feasible, as one got penalized for large constraints violations. There exists a very large body of literature dealing with possible ways of modifying existing infeasible Lagrangian solutions to make them feasible. Lagrangian heuristics are essentially problem dependent. Here are a few hints on how one may want to proceed. One may for instance try to get **feasible** solutions in the following way:

(1) by modifying the solution to correct its infeasibilities while keeping the objective function deterioration small.

Example: in production scheduling, if one relaxes the demand constraints, one may try to change production levels (down or up) so as to meet the demand (de Matta and Guignard 1994).

(2) by fixing (at 1 or 0) some of the meaningful decision variables according to their value in the current Lagrangian solution, and solving optimally

the remaining problem. Chajakis et al. (1996) called this generic approach the lazy Lagrangian heuristic. One guiding principle may be to fix variables that satisfy relaxed constraints.

Part of the success of Lagrangian relaxation comes from clever implementations of methods for solving the Lagrangian dual, with powerful heuristic imbedded at every iteration. In many cases, the remaining duality gap, i.e., the relative percentage gap between the best Lagrangian bound found and the best feasible solution found by heuristics is sufficiently small to forego enumeration. In some instances however an optimal or almost optimal solution is desired, and a Branch-and-Bound scheme adapted to replace LP bounds by LR bounds can be used. If the Lagrangian dual is solved by column generation, the scheme is called Branch-and-Price, as new columns may need to be priced-out as one keeps branching see Desrosiers et al. 1984), (Barnhart et al., 1998). In that case, branching rules need to be carefully designed. The hope is that such schemes will converge faster than LP-based Branch-and-Bound, as bounds will normally be tighter and nodes may be pruned faster. The amount of work done at a node, though, may be substantially more than solving an LP.

Concluding Remarks

- Lagrangian relaxation is a powerful family of tools for solving approximately integer programming problems. It provides
  - stronger bounds than LP relaxation when the problem(s) don't have the Integrality Property.
  - good starting points for heuristic search.
- The availability of powerful interfaces (GAMS, AMPL, etc.) and of flexible IP packages makes it possible for the user to try various schemes and to implement and test them.
- As illustrated by the varied examples described in this paper, Lagrangian relaxation is very flexible. Often some reformulation is necessary for a really good scheme to appear.
- It is not necessary to have special structures embedded in a problem to try to use Lagrangian schemes. If it is possible to decompose the problem structurally into meaningful components and to split them through constraint dualization, possibly after having introduced new variable expressions, it is probably worth trying.
- Finally, solutions to one or more of the Lagrangian subproblems might lend themselves to Lagrangian heuristics, possibly followed by interchange heuristics, to obtain good feasible solutions.

Lagrangian relaxation bounds coupled with Lagrangian heuristics provide the analyst with brackets around the optimal integer value. These are usually much tighter than the brackets coming from LP-based bounds and heuristics

See

- ▶ Branch and Bound
- ▶ Convex Hull
- ▶ Convex Optimization
- ▶ Heuristics
- ▶ Integer and Combinatorial Optimization
- ▶ Traveling Salesman Problem

References

Andalaft, N., Andalaft, P., Guignard, M., Magendzo, A., Wainer, A., & Weintraub, A. (2003). A problem of forest harvesting and road building solved through model strengthening and Lagrangean relaxation. *Operations Research, 51*(4), 613–628.

Balas, E., & Christofides, N. (1981). A restricted Lagrangean approach to the traveling salesman problem. *Mathematical Programming, 21*, 19–46.

Barahona, F., & Anbil, R. (2000). The volume algorithm: Producing primal solutions with a subgradient method. *Mathematical Programming, 87*(3), 385–399.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P., & Vance, P. (1998). Branch-and-price: Column generation for solving huge integer programs. *Operations Research, 46*(3), 316–329.

Bilde, O., & Krarup, J. (1977). Sharp lower bounds and efficient algorithms for the simple plant location problem. *Annals of Discrete Mathematics, 1*, 79–97. Also a 1967 report in Danish Bestemmelse af optimal beliggenhed af produktionssteder, Research Report, IMSOR.

Birge, J., & Louveaux, F. (2011). *Introduction to stochastic programming* (Springer series in operations research and financial engineering, 2nd ed.). New York: Springer.

Chajakis, E., Guignard, M., Yan, M., & Zhu, S. (1996). The Lazy Lagrangean heuristic. Optimization Days, Montreal, May 1996.

Chen, B., & Guignard, M. (1998). Polyhedral analysis and decompositions for capacitated plant location-type problems. *Discrete Applied Mathematics, 82*, 79–91.

Dantzig, G. B., & Wolfe, P. (1960). The decomposition principle for linear programs. *Operations Research, 8*, 101–111.

Dantzig, G. B., & Wolfe, P. (1961). The decomposition algorithm for linear programs. *Econometrica, 29*(4), 767–778.

de Matta, R., & Guignard, M. (1994). Dynamic production scheduling for a process industry. *Operations Research, 42*, 492–503.
Desrosiers, J., Soumis, F., & Desrochers, M. (1984). Routing with time windows by column generation. *Networks, 14*(4), 545–565.
Escudero, L. F. (2009). On a mixture of the fix-and-relax coordination and Lagrangean substitution schemes for multistage stochastic mixed integer programming. *TOP, 17*, 5–29.
Escudero, L., Guignard, M., & Malik, K. (1994). A Lagrangean relax-and-cut approach for the sequential ordering problem with precedence relationships. In C. Ribeiro (Ed.), *Annals of operations research*, 50, Applications of combinatorial optimization, pp. 219–237.
Everett, H., III. (1963). Generalized Lagrange multiplier method for solving problems of optimum allocation of resources. *Operations Research, 11*, 399–417.
Fisher, M. L. (1981). The Lagrangian relaxation method for solving integer programming problems. *Management Science, 27*, 1–18.
Fisher, M. L. (1985). An applications oriented guide to Lagrangian relaxation. *Interfaces, 15*, 10–21.
Geoffrion, A. M. (1974). Lagrangean relaxation for integer programming. *Mathematical Programming Study, 2*, 82–114.
Geoffrion, A. M., & McBride, R. (1978). Lagrangean relaxation applied to capacitated facility location problems. *AIIE Transactions, 10*, 40–47.
Glover, F., & Klingman, D. (1988). Layering strategies for creating exploitable structure in linear and integer programs. *Mathematical Programming, 40*(2), 165–182.
Guignard, M. (1998). Efficient cuts in Lagrangean relax-and-cut schemes. *European Journal of Operational Research, 105*(1), 216–223.
Guignard, M., & Kim, S. (1987). Lagrangean decomposition: a model yielding stronger Lagrangean bounds. *Mathematical Programming, 39*, 215–228.
Guignard, M., & Rosenwein, M. B. (1989). An application-oriented guide for designing Lagrangian dual ascent algorithms. *European Journal of Operational Research, 43*, 197–205.
Guignard, M., & Yan, H. (1993). Structural decomposition methods for dynamic multi-hydropower plant optimization. *Research report 93-12-01*, Operations and Information Management Department, University of Pennsylvania.
Held, M., & Karp, R. M. (1970). The traveling salesman problem and minimum spanning trees. *Operations Research, 18*, 1138–1162.
Held, M., & Karp, R. M. (1971). The traveling salesman problem and minimum spanning trees: Part II. *Mathematical Programming, 1*, 6–25.
Lemaréchal, C. (1974). An algorithm for minimizing convex functions. In *Proceedings IFIP'74 congress*, North Holland, Amsterdam, pp. 552–556.
Näsberg, M., & Jörnsten, K. O., & Smeds, P. A. (1985). Variable splitting – A new Lagrangean relaxation approach to some mathematical programming problems. *Report LITH-MAT-R-85-04*, Linkoping University.
Reinoso, H., & Maculan, N. (1992). Lagrangean decomposition in integer linear programming: A new scheme. *INFOR 30*(1).
Soenen, R. (1977). *Contribution à l'étude des systèmes de conduite en temps réel en vue de la commande d'unités de fabrication*. Thèse de Doctorat d'Etat, Université de Lille, France.

Lanchester Attrition

The concept of an explicit mathematical relationship between opposing military forces and casualty rates. The two classical laws are the linear law, that gives the casualty rate (derivative of force size with respect to time) of one side as a negative constant multiplied by the product of the two sides' force sizes, and the square law, which gives the casualty rate of one side as a negative constant multiplied by the opposing side's force size.

See

▶ Battle Modeling
▶ Homogeneous Lanchester Equations
▶ Lanchester's Equations

Lanchester's Equations

Joseph H. Engel
Bethesda, MD, USA

Introduction

Lanchester's equations are named for the Englishman, F.W. Lanchester, who formulated and presented them in 1914 in a series of articles contributed to the British journal, *Engineering*, which then were printed in toto in Lanchester (1916). More recent presentation of these results appeared in the 1946 Operations Evaluation Group Report No. 54, *Methods of Operations Research* by Philip M. Morse and George E. Kimball, which was published commercially by John Wiley and Sons (Morse and Kimball 1951). In addition, a reprint of the original 1916 Lanchester work, "Mathematics in Warfare," appeared in *The World of Mathematics*, Vol. 4, prepared by James R. Newman and published by Simon and Schuster in 1956.

The significance of these equations is that they represented possibly the first mathematical analysis of forces in combat, and served as the guiding light (for the U.S. and its allies) behind the development,

during and after World War II, of all two sided combat models, simulations, and other methods of calculating combat losses during a battle.

It appears that M. Osipov developed and published comparable equations in a Tsarist Russian military journal in 1915, perhaps independent of Lanchester's results. A translation of his work into English, prepared by Robert L. Helmbold and Allen S. Rehm, was printed in September 1991 by the U.S. Army Concepts Analysis Agency.

Lanchester's equations present a mathematical discussion of concepts such as the relative strengths of opposing forces in battle, the nature of the weapons, the importance of concentration, and their effects on casualties, and the outcome of the battle. His arguments are paraphrased here, preserving much of his original symbolism. The equations deal with ancient warfare and modern warfare.

Ancient Warfare

Lanchester explained that, because of the limited range of weapons in ancient warfare (like swords), the number of troops on one side of a battle (the Blue force) that are actively engaged in hand-to-hand combat on the combat front at any time during the battle must equal approximately the number of troops responding to them on the other side (the Red force). For this reason, one may assume that the rate at which casualties are produced is constant, because the number of troops actively engaged on each side is constant (until very near the end of the battle), and the rate c (>0), at which Blue combatants become casualties is a product of the fixed number of Red troops engaged and their average individual casualty producing effectiveness (dependent on the average strength of Red's weapons and the effectiveness of the Blue defenses). Similar results apply to k (>0), the Red casualty rate. The two casualty rates need not be the same, as the weapons and defenses of the two sides may differ.

If $b(t)$ is the number of effective Blue troops at time t after the battle has started and $r(t)$ is the number of effective Red troops, the following equations can be assumed to obtain:

$$db/dt = -c, dr/dt = -k. \tag{1}$$

The relationship between the sizes of the two forces may easily be ascertained by observing from (1) that

$$db/dr = c/k, \tag{2}$$

from which it can be deduced that

$$k[b(0) - b(t)] = c[r(0) - r(t)]. \tag{3}$$

In the above equations, $b(0)$ and $r(0)$ are assumed to be the initial (positive) sizes of the forces at time 0, the beginning of the battle, and the equations are valid only as long as $b(t)$ and $r(t)$ remain greater than zero. Assuming the combatants battle until all the troops on one side or the other are useless for combat, having become casualties, the battle ends at the earliest time when $b(t)$ or $r(t)$ becomes equal to zero. Thus, solving for r in (3) when b becomes 0 (or vice versa) yields: when

$$b(t) = 0, r(t) = [c^* r(0) - k^* b(0)]/c$$

and when

$$r(t) = 0, b(t) = [k^* b(0) - c^* r(0)]/k. \tag{4}$$

Thus, if $c^* r(0) > k^* b(0)$, the Red force wins the battle, while if $k^*\ b(0) > c^*\ b(0)$, the Blue force wins the battle. Summarizing these observations by designating the initial effectiveness of the Blue force to be $k^* b(0)$, and that of the Red force $c^* r(0)$, shows that the force with the larger initial effectiveness wins, while equal initial effectiveness ensures a draw.

It is also simple to return to the original differential equations of (1) and to solve them to determine the number of effective troops of either force as a linear function of time. This essentially completes Lanchester's modeling of ancient warfare.

Modern Warfare

Lanchester postulated that the major difference between modern and ancient warfare is the ability of modern weapons (such as rifles and, to a lesser degree bows and arrows, cross bows, etc.) to produce casualties at long range. As a result, the troops on one side of an engagement can, in principle, be fired upon by the entire opposing force. Consequently, assuming

that all of each of the troops on a side have the same (average) ability to produce casualties at a fixed rate, the combined casualty rate against a given side is proportional to the number of effective troops on the other side.

This leads directly to the following differential equations constituting Lanchester's model of modern warfare:

$$db/dt = -c^*r, dr/dt = -k^*b. \quad (5)$$

As in the ancient warfare case, the individual casualty producing rates, c and k, are assumed to be known constants for the duration of the battle.

Now combine these two equations (as was done in the ancient warfare case) and obtain

$$db/dr = (c^*r)/k^*b. \quad (6)$$

Equation (6) is solved to obtain the relationship between the numbers of effective forces on the two sides as the battle progresses. This leads to

$$k\left[b^2(0) - b^2\right] = c\left[r^2(0) - r^2\right]. \quad (7)$$

Since these equations are valid only when $b \geq 0$ and $r \geq 0$, observe, as in the ancient warfare case, that, with the battle ending when the losing side has been reduced through casualties to no effective troops, and the victor has a positive number of effective troops, the force with the larger initial effectiveness, [$k^*b^2(0)$ for Blue and $c^*r^2(0)$ for Red], will win the battle, while equal initial effectiveness produces a draw. Equation (7) and this paragraph constitute Lanchester's Square Law for his model of modern warfare.

Again, as in the ancient warfare case, it is possible to solve the initial differential equations in (5) to obtain the specific functions that describe the behavior of the side of either force as a function of time. These results also appear in Morse and Kimball, (1951), and this essentially completes Lanchester's modeling of modern warfare.

Extensions

In presenting his results, Lanchester used many techniques that are taken for granted in contemporary OR practice. He formulated clear assumptions about the operation of the system he was studying, derived the mathematical consequences of his assumptions, and discussed how variation of assumptions affected results. Consequently he was able to provide specific numerical insights into characteristics of the system that could be translated into useful ways of improving a system that operated in accordance with the specified assumptions.

It was possible for Lanchester to accomplish his mathematical modeling by using what is often referred to as the First Theorem of Operations Research:

> A function of the average equals the average of the function.

The above result applies only in very special circumstances; nevertheless, there are many cases in which use of this theorem allows deterministic results to be derived easily. Such results will usually provide a good approximation of average results occurring in reality. It is through this technique that various chemical formulas or formulas in the physical sciences pertaining to concepts such as temperature, thermodynamics, etc., were derived.

In those formulations, it is assumed that a group of many small objects moving at various speeds with a known average speed will function in the same manner as if all the objects moved at the same (average) speed. Similarly, in his warfare modeling, Lanchester assumed that the casualty producing rate of every one of the troops on one side of a battle was constant and equal to the average (per troop) casualty producing rate of the entire force, and the same is true of the troops on the other side.

The usefulness of Lanchester's work is primarily in its demonstration of the fact that it is possible to draw mathematical and numerical conclusions concerning the occurrence of casualties in certain battles that can be described, a priori, as conforming to certain specified assumptions concerning how the battle is conducted. From such an observation, it is possible to generalize and derive other models that conform to other sets of assumptions, so that a wider range of combat situations can be dealt with. This has led to all sorts of models that can be handled through generalizations of Lanchester's techniques.

The analyst can take into account other factors not specifically covered by Lanchester, such as addition or

withdrawal of troops in the course of an engagement. Movement of forces can be considered. Different weapons and defensive techniques can be studied.

Dispersing and hiding the troops on one side of a battle (as in guerrilla warfare) affects the rate at which they can be hit by the other side, which led Lanchester to present another differential equation for such a force. This leads to analyses in which one or the other or both forces engage in ancient, modern, or guerrilla warfare. There are nine kinds of battles that an analyst can deal with just by adding the consideration of the possibility of guerrilla warfare to his bag of tricks (Deitchman 1962).

Clearly there is a great deal of flexibility in deriving models involving the use of deterministic differential equations that predict specific average results. The probabilistic events that take place during the course of a battle can also be dealt with in comparatively simple cases as demonstrated by B.O. Koopman and described in Morse and Kimball (1951). Regrettably, the mathematics of probabilistic systems is frequently much more difficult than that of deterministic systems, and the need to recognize the existence of all sorts of complications in a battle, frequently leads to rather complicated and abstruse mathematics which can best be handled through the use of computers for the required numerical calculations.

The field of combat simulation is recognized as a direct descendant of the Lanchester approach. Of historic interest in this connection is the fact that Lt. Fiske of the U.S. Navy presented, in 1911, a model of warfare consisting of a salvo by salvo table that computed casualties on two sides of a battle. This material was brought to the attention of contemporary analysts by H.K. Weiss (1962).

Engel (1963) showed that the equations of the Fiske model were difference equations that became, in the limit as the time increment between successive salvos approached zero, identical to the Lanchester differential equations of modern warfare. In a sense, this validated the use of discrete time models that approximated combat models for computer calculations, allowing greater confidence on the part of the analyst that no great surprises would result from a use of such discrete time approximations of combat models.

A cautionary note must be sounded at this point. Before using whatever mathematical model the analyst may have derived in discussing any past or future battles, the analyst must be certain that the assumptions of the model on how the battle will be conducted and terminated pertain to the battle being analyzed. The analyst should be able to derive the appropriate values of any parameters (such as $b(0)$, $r(0)$, c and k) to be used in the Lanchester or other models believed to apply in the case under study. Thought experiments do not suffice. The analyst must examine data to determine whether the assumptions provide a valid description of the way the battle proceeds, and to ascertain from relevant combat and experimental data that the model's numerical values for the parameters are appropriate.

Validation of Equations

Lanchester did not provide any demonstration of the relevance of his models to any specific historic battles, although he did discuss examples from history in which he suggested that the results of certain tactical actions were consistent with results that could be derived from his models. A validation of Lanchester's modern warfare equations was first given by Engel (1954), based on an analysis of the Battle of Iwo Jima during World War II. The analysis showed that the daily casualties inflicted on the U.S. forces over the approximately forty days of the battle were consistent with Lanchester's model for modern warfare. Since that time, additional analyses of combat results and experiments have demonstrated that the values of various parameters can be estimated for use in specified combat situations, and that appropriate combat models can be used in conjunction with those parameter values to obtain results of interest to military planners and decision makers.

The modeling methodology pioneered by Lanchester in the field of combat casualty analysis has served as a most important guide for analysts of military problems. He showed how application of these techniques can be used in developing mathematical models of combat that can be applied in forecasting the results of hypothetical battles. This enables operations research analysts to predict outcomes of these battles, plan tactics and strategy, develop weapons requirements, determine force requirements, and otherwise assist planners and decision makers concerned with the effective use of military forces.

See

- ▶ Battle Modeling
- ▶ Military Operations Research
- ▶ Verification, Validation, and Testing of Models

References

Deitchman, S. J. (1962). A Lanchester model of guerrilla warfare. *Operations Research, 10*, 818–827.
Engel, J. H. (1954). A verification of Lanchester's Law. *Journal of the Operations Research Society of America, 2*, 163–171.
Engel, J. H. (1963). Comments on a paper by H.K. Weiss. *Operations Research, 11*, 147–150.
Lanchester, F. W. (1916). *Aircraft in warfare: The dawn of the fourth arm*. London: Constable and Company.
Morse, P. M., & Kimball, G. E. (1951). *Methods of operations research*. New York: Wiley. Also Dover Publications, 2003.
Weiss, H. K. (1962). The Fiske model of warfare. *Operations Research, 10*, 569–571.

Laplace Transform

For any function $g(t)$ defined on $t \geq 0$ (e.g., a probability density), its Laplace transform is defined as $\int_0^\infty e^{-st} g(t)dt$, $\mathrm{Re}(s) > 0$.

Laplace-Stieltjes Transform

For any function $G(t)$ defined on $t \geq 0$ (e.g., a cumulative probability distribution function), its Laplace-Stieltjes transform (LST) is defined as $\int_0^\infty e^{-st} dG(t), \mathrm{Re}(s) > 0$. When the function $G(t)$ is differentiable, it follows that the LST is equivalent to the regular Laplace transform of the derivative, say $g(t) = dG(t)/dt$.

Large Deviations

In probability theory, the study of asymptotic tail behavior of sequences of probability distributions. For example, the probability that a sample mean exceeds a certain threshold decays exponentially to zero according to some rate function. Large deviations theory is used in stochastic simulation for more effectively estimating rarely occurring events.

See

- ▶ Rare Event Simulation

References

Dembo, A., & Zeutoni, O. (2009). *Large deviations techniques and applications* (2nd ed.). New York: Springer.
Varadhan, S. R. S. (2008). Special invited paper: Large deviations. *Annals of Probability, 36*(2), 397–419.

Large-Scale Systems

James K. Ho
University of Illinois at Chicago, Chicago, IL, USA

Introduction

In OR/MS, large-scale systems refer to the methodology for the modeling and optimization of problems that, due to their size and information content, challenge the capability of existing solution technology (Lasdon 1970). There is no absolute measure to classify such problems. In any given computing environment, the cost-effectiveness of problem solving generally depends on the dimensions and the volume of data involved. As problems get larger, the cost tends to go up, lowering effectiveness. Even before the physical limits of the hardware or the numerical resolution of the software are exceeded, the effectiveness of the solution environment may have become unacceptable. Efforts to improve on any of the relative performance measures such as solution time, numerical accuracy, memory and other resource requirements, are subjects in the topic of large-scale systems. Since solving larger problems more effectively is also an obvious goal in all specializations of operations research, there are natural linkages and necessary overlaps with most other areas in the field (Nemhauser 1994).

All known methodology for large-scale systems can be viewed as the design of computational techniques to take advantage of various structural properties exhibited by both the problems and known solution algorithms (Koussoulas and Groumpos 1999). Broadly speaking, such special properties can be regarded as either micro-structures or macro-structures. Micro-structures are properties that are independent of permutations in the ordering of the variables and constraints in the problem. An example is sparsity in the constraint coefficients. Macro-structures are those that depend on such orderings. An example is the block structure of loosely coupled or dynamic systems.

Using Micro-Structures of Problems

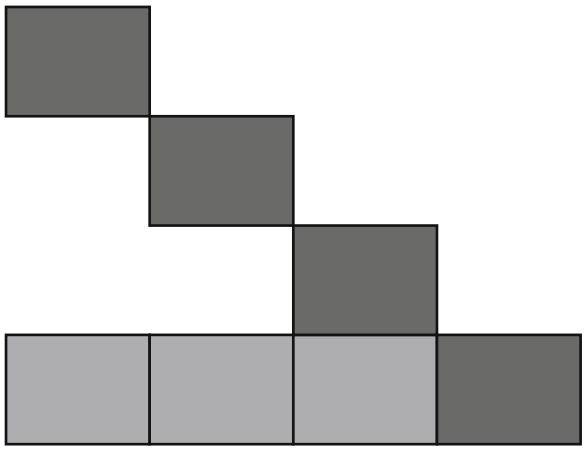
In the modeling of real systems, the larger the problem, the less likely it is for a variable to interact with all the others. If each variable is coupled only to a small subset of the total, the resulting constraints will be sparse. Techniques that eliminate the representation of the nonexistent interactions can reduce storage requirement significantly. For example, a linear program with 10,000 variables and 10,000 constraints has potentially 10^8 coefficients. If on the average, each variable appears in 10 constraints, there will be only 10^5 nonzero coefficients, implying a density of 0.1%. Sparse matrix methods from numerical analysis have been used with great success here. Furthermore, the nonzero coefficients may come from an even smaller pool of unique values. This feature is known as supersparsity and allows additional economy in data storage. Large, complex models are usually generated systematically by applying the logic of the problem iteratively over myriad parameter sets. This may lead to formulations with redundant variables and constraints. Examples include flow balance equations that produce a redundant constraint when total input equals total output; lower and upper bounds that are equal imply the variable can be fixed. Methods to simplify the problem by identifying and removing such redundancies are incorporated into the procedure of preprocessing. It is not unusual to observe reductions of problem dimensions by 10 to 50% with this approach.

Using Micro-Structures of Algorithms

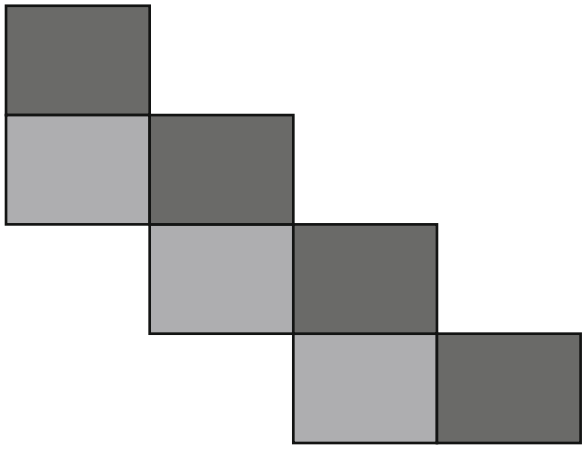
Algorithms may have steps that are adaptable to advanced computing architecture at the micro-processing level. An example is the vectorization of inner-product calculations in the simplex method. A completely different exploit is the relatively low number of iterations required by interior-point methods. As the number of iterations seems to grow rather slowly with problem size, it is a micro-structure of such algorithms that automatically sheds light on the optimization of large-scale systems. Yet another promising approach that falls under this heading is the use of sampling techniques in stochastic optimization.

Using Macro-Structures of Problems

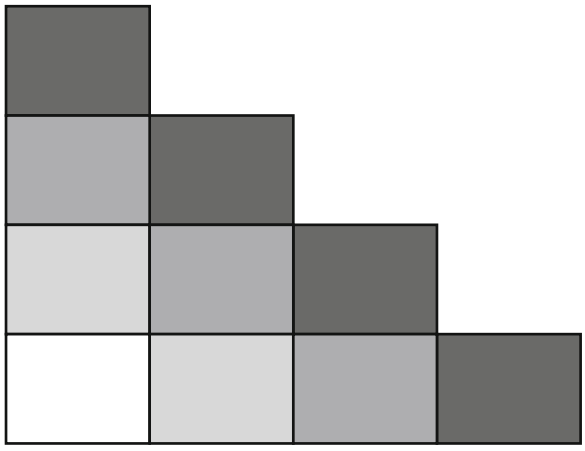
Most large-scale systems are comprised of interacting subsystems. Examples are multidivisional firms with a headquarters coordinating the activities of the semi-autonomous divisions; time-phased models of dynamic systems with linkages only among adjacent time periods; capital investment or financial planning models with each period linked to all subsequent periods. Linear programming modeling of the above examples gives rise to problems with the block-angular, staircase and block-triangular structures, respectively (Figs. 1, 2, and 3). Other variations and combinations are also possible. Two major approaches to take advantage of such structures are decomposition and factorization. Decomposition relies on algorithms that transform the problem into a sequence of smaller subproblems that can be solved independently. Various schemes are devised to coordinate the subproblems and steer them towards the overall solution. Many algorithms are derived from the Dantzig-Wolfe decomposition principle which provides a rigorous framework for this approach. Factorization is the adaptation of existing algorithms to take advantage of the problem structure. In the case of the simplex method, the representation of the basis matrix required at each step can be partitioned into blocks and updated separately. It has been shown that all of the simplex-based techniques proposed over the years under somewhat confusing guises



Large-Scale Systems, Fig. 1 Block-angular structure



Large-Scale Systems, Fig. 2 Staircase structure



Large-Scale Systems, Fig. 3 Block-triangular structure

of partitioning and decomposition are indeed special cases of the factorization approach (Dantzig et al. 1981).

Using Macro-Structures of Algorithms

Both decomposition and factorization algorithms are natural candidates for parallel and distributed computation since they involve the solution of independent subproblems. The latter can be solved concurrently on multiprocessor computers of various architectures. Particularly suitable is the class of Multiple-Instruction-Multiple-Data (MIND) machines that are essentially networks of processors that can execute independent instructions. They represent a cost-effective way to harness tremendous computing power from relatively modest and economical components. One processor can be programmed as the coordinator of the algorithmic procedures. Each of the other processors can be assigned a subproblem and programmed to communicate with the coordinating process. As the gain in overall efficiency is bounded by the number of processors used, the intent of this approach is to realize the full potential of certain algorithms rather than fundamentally enhancing their performance. It is, however, becoming an essential aspect of large-scale systems, as multi-processor computers are expected to be prevalent (Eckstein 1993). Early results have been obtained for decomposition (Ho and Sundarraj 1997), factorization (Ho and Sundarraj 1994), and barrier methods (Lustig and Rothberg 1996).

Concluding Remarks

Linear and mixed integer programming remain the primary focus in the optimization of large-scale systems. New computer architectures with ever-increasing processing power and memory capacities have facilitated the empirical approach to algorithmic development. Experimentation with large-scale problems becomes a viable strategy to identify, test, and fine tune ideas for improvement. This has been especially successful in commercial implementations of both the simplex and interior-point methods exploiting mainly the micro-structures of problems and algorithms. Problems with hundreds of

thousands of constraints and millions of variables are solvable on workstation-grade computers (Fourer 2009). Earlier experiences with macro-techniques in decomposition and factorization did not have the benefits of the more modern technological advances. The results are either inconclusive or less than promising (Ho 1987). Future work, especially in hybrid schemes using advanced hardware, may lead to significant contributions to large-scale non-linear, integer and stochastic optimization.

See

- ▶ Dantzig-Wolfe Decomposition Algorithm
- ▶ Density
- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Nonlinear Programming
- ▶ Parallel Computing
- ▶ Sparsity
- ▶ Super-Sparsity

References

Dantzig, G. B., Dempster, M. A. H., & Kallio, M. J., (Eds.) (1981). Large-scale linear programming. IIASA CP-81-S1, Laxenburg, Austria.

Eckstein, J. (1993). Large-scale parallel computing, optimization and operations research: A survey. ORSA Computer Science Technical Section Newsletter, 14, Fall.

Fourer, R. (2009). 2009 Linear programming software survey. OR/MS Today, 5 June 2009.

Ho, J. K. (1987). Recent advances in the decomposition approach to linear programming. *Mathematical Programming Study, 31*, 119–128.

Ho, J. K., & Sundarraj, R. P. (1994). On the efficacy of distributed simplex algorithms for linear programming. *Computational Optimization and Applications, 3*, 349–363.

Ho, J. K., & Sundarraj, R. P. (1997). Distributed nested decomposition of staircase linear programs. *ACM Transactions on Mathematical Software, 23*, 148–173.

Koussoulas, N. T., & Groumpos, P. P., (Eds.) (1999). Large scale systems: Theory and applications. In *Proceedings of the 8th IFAC/IFORS/IMACS/IFIP symposium*, July 1998, Elsevier Science, Amsterdam.

Lasdon, L. S. (1970). *Optimization theory for large systems*. New York: MacMillan.

Lustig, I. J., & Rothberg, E. (1996). Gigaflops in linear programming. *Operations Research Letters, 18*, 157–165.

Nemhauser, G. L. (1994). The age of optimization: Solving large-scale real-world problems. *Operations Research, 42*, 5–13.

Las Vegas Algorithm

Randomized algorithm that is guaranteed to give the correct result 100% of the time, in contrast to Monte Carlo methods, which provide statistical bounds.

See

- ▶ Monte Carlo Methods
- ▶ Randomized Algorithm

References

Hromkovic, J. (2005). *Design and analysis of randomized algorithms*. New York: Springer.

Latest Finish Time

The latest time an activity must be completed without delaying the end of a project. It is simply the sum of the latest start time of the activity and its duration.

See

- ▶ Network Planning

Latest Start Time

The latest time an activity can start without delaying the end of a project. A delay of an activity beyond the latest start time will delay the entire project completion by a corresponding amount. These times are calculated on the basis of a reverse pass through the network.

See

- ▶ Network Planning

Latin Square

▶ Combinatorics

LCFS

A queueing discipline wherein customers are selected for service in reverse order of their order of their arrival, i.e., on a last-come, first-served basis.

See

▶ LIFO
▶ Queueing Theory

LCP

Linear complementarity problem.

See

▶ Complementarity Problems
▶ Quadratic Programming

LDU Matrix Decomposition

For a nonsingular square matrix $\boldsymbol{A}$, the transformation by Gaussian elimination of $\boldsymbol{A}$ into the form $\boldsymbol{LDU}$, where $\boldsymbol{L}$ is a lower triangular matrix, $\boldsymbol{D}$ is a diagonal matrix, and $\boldsymbol{U}$ is an upper triangular matrix. It can be written so that the diagonal elements of $\boldsymbol{L}$ and $\boldsymbol{U}$ are equal to one and $\boldsymbol{D}$ is the diagonal matrix of pivots.

See

▶ LU Matrix Decomposition
▶ Matrices and Matrix Algebra

Lean Manufacturing

▶ Quality Control

Lean Six Sigma

▶ Quality Control

Learning

James R. Buck
The University of Iowa, Iowa City, IA, USA

Introduction

Learning is a human phenomenon where performance improves with experience. There are a number of reasons for task improvement. As tasks are repeated, elements of the task are: better remembered, cues are more clearly detected, skills are sharpened, eye-hand coordinations are more tightly coupled, transitions between successive tasks are smoothed, and relationships between task elements are discovered. Barnes and Amrine (1942), Knowles and Bell (1950), Hancock and Foulke (1966), Snoddy (1926), *and* Wickens (1992) have described these and other sources of human performance change. All these causes of individual person improvement manifest themselves in faster performance times, fewer errors, less effort, and there is often a better disposition of the person as a result.

Learning is implied by performance changes due primarily to experience. Changes in the methods of performing a task, replacing human activities with machines, imparting information about the job, training, acquiring performance changes with incentive systems, and many other things can cause performance changes other than learning. Thus, detection involves the identification of an improvement trend as a function of more experience. It also involves the elimination of other explanations for this improvement. Analogous to a theory, learning can never be proved; it can only be disproved.

After detecting learning, measurement and prediction follows. These activities involve fitting mathematical models, called learning curves, to performance data. First, there is the selection of an appropriate model. Following the selection of a model, there is the matter of fitting the selected model to performance data. In some cases alternative models are fit to available data and the quality of fit is a basis in the choice of a model.

Some of those sources which contribute to an individual person's improvement in performance with experience are similar to the causes of improvement by crews, teams, departments, companies, or even industries with experience. As a result, similar terms and descriptions of performance change are often fit to organizational performance changes. However, the term progress curves (Konz 1990) is more often applied to cases involving: assembly lines, crews, teams, departments, and other smaller groups of people, whereas the term experience curves is sometimes applied to larger organizational groups such as companies and industries (Hax and Majluf 1982). A principal distinction between these different types of improvement curves is that between-person activities (e.g., coordination) occur as well as within-person learning. In the case of progress curves, there are improvement effects due to numerous engineering changes. Experience curves also embody scientific and technological improvements, as well progressive engineering changes and individual-person learning. Regardless of the person, persons, or thing which improves or the causes of improvement, the same learning curve models are frequently applied. Progress and experience curves are really forms of personification.

Learning occurs in a number of important applications. One of these applications is the prediction of direct labor changes in production. Not only is this application important to cost estimation, it is also important in production planning and manning decisions. Another application is the selection of an operational method. If there are alternative methods of performing particular operations which are needed, then one significant criterion in the selection of an appropriate method is learning because the average cost can favor one method over another that has lower initial performance costs. In other cases, one operation can cause bottlenecks in others unless the improvements with experience are sufficient over time. Also, production errors can be shown to decrease with experience as another form of learning and so learning is important in quality engineering and control.

Performance Criteria and Experience Units

Performance time is the most common criterion used for learning curves in industry. Production cycles are also the most commonly used variable for denoting experience. If t_i is the performance time on the ith cycle, then a learning curve should predict t_i as a function of n cycles. Since learning implies improvement with experience, then one would expect $t_i \leq t_{i-1}$ for the typical case, $i = 1, 2, \ldots, n$ cycles.

An associated time criterion on the ith cycle is the cumulative average performance time on the ith cycle or A_i. Cumulative average times consists of the sum of all performance times up to and including the nth cycle divided by n. In the first cycle, $A_1 = t_1$. With learning, t_i tends to decrease with i and so does A_i. However, A_i decreases at a slower rate than t_i. This effect can be shown by the first-forward difference of A_i, which is

$$\Delta A_n = A_{n+1} - A_n = \frac{\sum_{i=1}^{n+1} t_i}{n+1} - \frac{\sum_{i=1}^{n} t_i}{n} = \frac{t_{n+1} - A_n}{n+1}. \quad (1)$$

So long as t_{n+1} is less than A_n, then ΔA_n is negative and the cumulative average time continues to decrease. It is also noted in (1) that with sequential values of A_i for $i = 1, 2, \ldots, n$, the values of t_i can be found. On the other hand, A_i can be predicted directly rather than t_i.

Another criterion of interest is accuracy. However, it is usually easier to measure errors in production as the complement of accuracy. Thus, the sequence of production errors are $e_1, e_2, \ldots, e_i, \ldots, e_n$ over n serial cycles where e_i is the number of errors found in a product unit as in typing errors per page (Hutchings and Towill 1975). If the person is doing a single operation on a product unit, then either an error is observed with a unit of production or it is not and observations over a production sequence is a series of zeros and ones. A more understandable practice is to define e_i as the fraction of the possible errors, where the observed number of errors is divided by the m possible

errors at an operation (Fitts 1966; Pew 1969). In this way, e_i is 0, some proper fraction, or 1. It also follows that a learning curve could be fit to the series of e_i values over the n observations sequential units of production or to the cumulative average errors. If learning is present, then one would expect to see a general decrease in e_i with increases in $i = 1, 2, \ldots, n$ and also the cumulative average errors would similarly decrease, but with a rate lag compared to the serial errors.

Pew (1969) invented the speed-accuracy-operating-characteristic graph which provides simultaneous analyses of correlated criteria. This operating characteristic consists of a bivariate graph where one axis denotes performance time per unit (complement is the speed) and the other axis denotes the number of errors per unit (complement is the accuracy). Simultaneous plots of speeds and accuracies with experience would be expected to show increases in both criteria with more experience. The slope of these plots with increases of experience describes bias between these criteria. It should be noted that when the power-form model is used for a prediction of learning performance, then logarithmic axes' measurements will linearize the plots.

Other Learning Metrics

Most applications of learning description, usually known as learning curves, use the production units as experience units, either as single units or lots. The time required to produce that product unit is the corresponding performance units. An alternative approach to predicting learning effects is to describe cumulative time as the experience unit (i.e., hours or days) and the number of production units produced during that experience unit. Thus, for cumulative production time $t = 1, 2, 3, \ldots, k, \ldots, m$ and corresponding production of $n_1, n_2, n_3, \ldots, n_k, \ldots, n_m$. Most learning curve models merely relate n_k to k. An alternative model of learning, which is not often shown, is the discrete exponential model which relates pairs of n_k values as

$$n_k = an_1 + b \tag{2}$$

where a and b are parameters. This model was originally proposed by Pegels (1969) for startup cost prediction. Later, Buck, Tanchoco, and Sweet (1976) showed that this model was really a first-order for-ward-difference equation (Goldberg 1961). It follows in this model that

$$n_k = a^k[n_1 - n^*] + n^* \tag{3}$$

where $n^* = b/(1 - a) > n_1$ and $0 < a < 1$. Since the parameter a is a fraction, the first term of (3) approaches zero with increasing k and so n^* is the asymptote. Accordingly, n_k approaches n^* exponentially with each discrete unit of time. Bevis et al. (1970) provided a similar model as

$$n_k = n^* + [n_1 - n^*]e^{-ck} \tag{4}$$

where k is a continuous measure to time and c is a parameter. Buck and Cheng (1993) used the discrete form in traditional format, but they showed that this model can be more difficult to fit to data than the more common power-form model. It can, however, give a more accurate description of human learning.

See

- Cost Analysis
- Cost-Effectiveness Analysis
- Learning Curves

References

Barnes, R., & Amrine, H. (1942). The effect of practice on various elements used in screw-driver work. *Journal of Applied Psychology, 26*, 197–209.

Bevis, F. W., Finnicat, C., & Towill, D. R. (1970). Prediction of operator performance during learning of repetitive tasks. *International Journal of Production Research, 8*, 293–305.

Buck, J. R., & Cheng, S. W. J. (1993). Instructions and feedback effects on speed and accuracy with different learning curve functions. *IIE Transactions, 25*(6), 34–47.

Buck, J. R., Tanchoco, J. M. A., & Sweet, A. L. (1976). Parameter estimation methods for discrete exponential learning curves. *AIIE Transactions, 8*, 184–194.

Fitts, P. M. (1966). Cognitive aspects of information processing III: Set for speed versus accuracy. *Journal of Experimental Psychology, 71*, 849–857.

Goldberg, S. (1961). *Introduction to difference equations*. New York: Wiley.

Goldberg, M. S., & Touw, A. E. (2003). *Statistical methods for learning curves and cost analysis*. Hannover, MD: INFORMS.
Hancock, W. M., & Foulke, J. A. (1966). Computation of learning curves. *MTM Journal, XL*(3), 5–7.
Hax, A. C., & Majluf, N. S. (1982). Competitive cost dynamics: The experience curve. *Interfaces, 12*(5), 50–61.
Hutchings, B., & Towill, D. R. (1975). An error analysis of the time constraint learning curve model. *International Journal of Production Research, 13*, 105–135.
Knowles, A., & Bell, L. (1950). Learning curves will tell you who's worth training and who isn't. *Factory Management, 108*, 114–115.
Konz, S. (1990). *Work design and industrial ergonomics* (3rd ed.). New York: Wiley.
Pegels, C. C. (1969). On startup of learning curves: An expanded view. *AIIE Transactions, 1*, 216–222.
Pew, R. W. (1969). The speed-accuracy operating characteristic. *Acta Psychologia, 30*, 16–26.
Snoddy, G. S. (1926). Learning and stability. *Journal of Applied Psychology, 10*, 1–36.
Wickens, C. D. (1992). *Engineering psychology and human performance* (2nd ed.). New York: Harper Collins.

Learning Curves

Andrew G. Loerch
Center for Army Analysis, Fort Belvoir, VA, USA

Introduction

With experience and training, individuals and organizations learn to perform tasks more efficiently, reducing the time required to produce a unit of output. This simple and intuitive concept is expressed mathematically through the use of the learning curve.

The learning curve was introduced in the literature by Wright (1936) who observed the learning phenomenon through his study of the construction of aircraft prior to World War II. Since then, these models have been used in the areas of work measurement, job design, capacity planning, and cost estimation in many industries. Yelle (1979) summarized 90 articles dealing with learning curves. Dutton et al. (1984) traced the history of progress functions by examining 300 articles. They note that the terms learning curve, progress function, and experience curve are often used interchangeably. However, many authors differentiate between them in the following way. Learning curves are used to describe only direct-labor learning, while progress functions also incorporate learning by managerial and technical personnel, as well as improvements due to technological change. The term experience curve is used to describe learning or progress at the industry level. Experience curves often use price as a surrogate measure for progress or learning. In the discussion below, no distinctions are made between these terms.

Dutton et al. (1984) also noted that learning curves are frequently confused with economies of scale. Although they are observed together in many cases, the two are separate effects with different causes. Progress and learning can occur in the absence of changes in size or scale of operations.

Basic learning-curve theory is described below, with emphasis given to the so-called power model. Other models are then introduced. Finally, issues regarding the estimation of learning-curve parameters are presented.

The Power Model

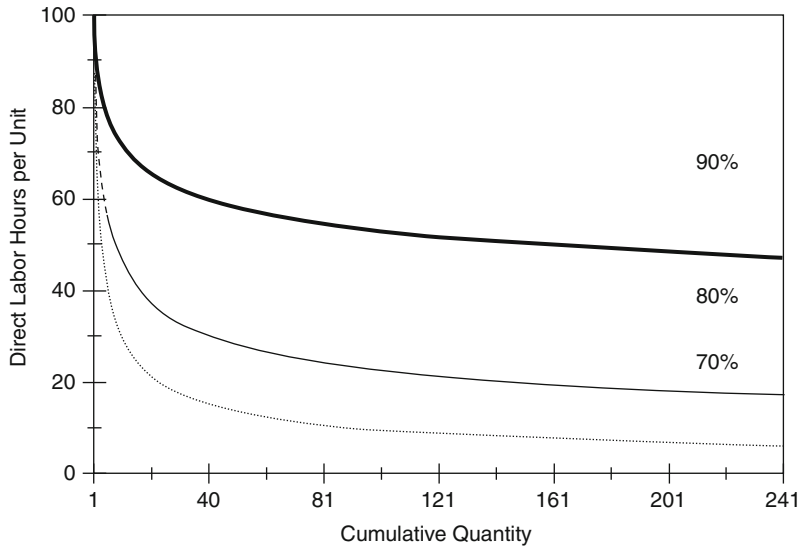
Also known as the log-linear model, the power model is the most frequently encountered implementation of the various learning-curve models. Wright observed that as the quantity of units manufactured doubles, the number of direct labor hours it takes to produce an individual unit decreases at a uniform rate. So, after one doubling of the cumulative production, direct-labor hours may have declined to, say 80% of its previous value. After an additional doubling there is another decline to 80% of that value, or 64% of the original. The learning rate, which is the actual decline per doubling, 80% in the above example, is assumed to be a characteristic of each particular type of manufacturing process.

In this model, learning curves have the following mathematical form:

$$L(y) = Ay^b,$$

where $L(y)$ = the number of hours needed to produce the yth unit, A = the number of hours needed to produce the first unit, y = the cumulative unit number, and b = the learning index, the learning-curve parameter, or the learning-curve slope parameter. To account for the effect of doubling, the learning-curve index is computed as follows:

$$b = (\log r)/(\log 2),$$



Learning Curves, Fig. 1 Learning curves with different rates

where r is the learning rate. Figure 1 shows graphs of three such curves with different learning rates.

Note that this model is also applicable to cost in addition to direct-labor hours. In a cost application, the parameter A would represent the cost of the first unit produced. The use of learning-curve costing is complicated by the problem of accounting for inflation and the change in hourly wages over time. In any event, labor hours can be easily converted into cost.

In the above model, the number of direct-labor hours required to produce the yth unit, or the cost of producing the yth unit is computed. Thus, the model is referred to as the Unit Formulation, and it is attributed to James Crawford who introduced its use to the Lockheed Corporation in 1944 (Smith 1989). A related model based on the original work of Wright is the so-called Cumulative Formulation, where, in the above notation, $L(y)$ would represent the average labor hours or cost of all the units produced through the yth unit. Note that the cumulative formulation tends to smooth the effects of unusually high or low labor hours or costs for individual or groups of units, and it has been found to be more useful for application to batch-type production processes. Although much of the work on learning curves has been directed at specifying the functional relation between unit costs or direct-labor hours and cumulative output, the range of output measures has been expanded to include, for example, industrial accidents per unit output, defects and complaints to quality control per unit output, and service requirements during warranty periods.

Variations of the Power Model

While the log-linear model has been, and is the most widely used model, several other geometries have been found to provide a better fits in particular sets of circumstances. Some of the more well-known models are:

1. Plateau model,
2. Stanford-B model, and
3. S-model.

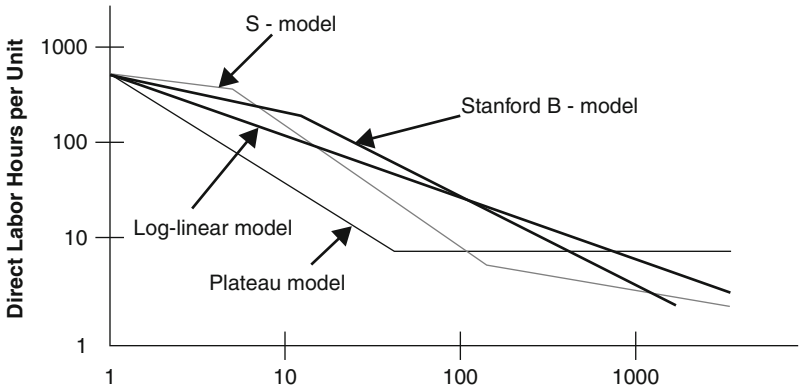
Figure 2 depicts these models on a logarithmic scale.

The plateau model was first described by Conway and Schultz (1959). It is used to represent the phenomenon that the learning phase of a process is finite and is followed by a steady state phase. This model is often associated with machine-intensive manufacturing.

The Stanford-B model, expressed symbolically as

$$L(y) = A(B + y)^b,$$

represents a process that experiences accelerated learning after B units are produced (other notation as



Learning Curves, Fig. 2 Log-log plot of learning curve geometrics

previously defined). This model was developed at the Stanford Research Institute and is useful for processes with design changes (Garg and Milliman 1961).

The *S*-model, described by Cochran (1960), combines reduced learning at the outset of production, with another slackening of learning later in the production process. This model is usually approximated as a three-segment straight line on a log-log graph and is sometimes used for heavy labor-intensive industries.

The choice of the appropriate model is usually based on empirical studies of the process in question and historical experience with similar processes. The utilization of these more complex representations involves increased difficulty in parameter estimation, coupled with limited improvement in accuracy. As such, the basic log-linear model continues to find favor among practitioners.

Other Factors Affecting Learning

Frequently, other factors affect production that, if ignored, could bias the estimation of the rate of learning. As mentioned, the presence of economies of scale would result in the situation where a more than proportional increase in output would be obtained due to an increase in inputs. If the effects of this variable are not controlled for in the estimation of learning rates, and the scale of the operation is gradually increased over time, the amount of learning would be overestimated. Other such factors that are independent of direct labor learning include increased capital investment, multiple shifts, time lapses between performance of operations, and production rate. Argote and Epple (1990) provided a review of the literature regarding the incorporation of factors that affect learning.

Estimation of Learning-Curve Parameters

Most estimation schemes rely on the logarithmic representation of the learning curve, written as follows:

$$\log L = \log A + \log y.$$

The learning-curve parameters, A and b, are estimated either by plotting historical values on a log-log graph and visually fitting a line, or by computing the least squares regression line through the log-log data. Several computer programs are commercially available to estimate the learning-curve parameters.

Frequently, organizations collect historical data for batches or lots, as opposed to discrete units. To estimate the parameters in this case, the batch's average labor or cost and the unit whose labor or cost corresponds to that average, the lot midpoint, must be known. The logarithm of this value is then used as the independent variable in the regression with the log of the average unit cost of the lot as the dependent variable. Note that the unit expressed by the batch size divided by two is not the lot midpoint since the learning curve is nonlinear. The actual lot midpoint, Q, is represented as the following:

$$Q = \left[\frac{(y_l - y_f + 1)(1+b)}{(y_l + .5)^{1+b} - (y_f - .5)^{1+b}}\right]^{-1/b},$$

where $y_f =$ the first unit of the batch, and $y_l =$ the last unit of the batch. Observe that this value cannot be computed without first knowing the learning-curve index, b. As such, the approximate algebraic lot midpoint is used. This value is computed by:

$$Q = \frac{y_f + y_l + 2\sqrt{y_f y_l}}{4}$$

The learning-curve parameters are estimated first using the approximate value of Q for each lot. The value of b is then used to calculate the actual lot midpoint, and the parameters are estimated again, and then iterated until the desired accuracy is obtained.

Concluding Remarks

Research in the area of learning curves has been extensive and many models have been hypothesized to describe the learning process. Learning-curve models have proven to be useful tools in many business and government applications. These include cost estimation, bid preparation and evaluation, labor requirement estimation, establishment of work standards, and financial planning.

See

- ▶ Cost Analysis
- ▶ Cost-Effectiveness Analysis
- ▶ Learning

References

Argote, L., & Epple, D. (1990). Learning curves in manufacturing. *Science, 247*, 920–924.

Buck, J. R., & Cheng, S. W. J. (1993). Instructions and feedback effects on speed and accuracy with different learning curve functions. *IIE Transactions, 25*(6), 34–47.

Cochran, E. B. (1960, July–August). New concepts of the learning curve. *Journal of Industrial Engineering*.

Conway, R. W. & Schultz, A. (1959, January–February). The manufacturing progress function. *Journal of Industrial Engineering*.

Dutton, J. M., Thomas, A., & Butler, J. E. (1984). The history of progress functions as a management technology. *Business History Review, 58*, 1984.

Garg, A., & Milliman, P. (1961). The aircraft progress curve modified for design changes. *Journal of Industrial Engineering, 12*, 23.

Goldberg, M. S., & Touw, A. E. (2003). *Statistical methods for learning curves and cost analysis*. Hanover, MD: INFORMS.

Smith, J. (1989). *Learning curves for cost control, industrial engineering and management press*. Norcross, GA: Institute of Industrial Engineers.

Wright, T. P. (1936). Factors affecting the cost of airplanes. *Journal of Aeronautical Sciences, 3*(4), 122–128.

Yelle, L. E. (1979). The learning curve: Historical review and comprehensive survey. *Decision Sciences, 10*, 302–328.

Least-Squares Analysis

- ▶ Quadratic Programming
- ▶ Regression Analysis

Leontief Matrix

- ▶ Input–Output Analysis

Level Crossing Methods

Percy H. Brill
University of Windsor, Windsor, Ontario, Canada

Introduction

Level crossing methods for obtaining probability distributions in stochastic models such as queues and inventories were originated by Brill (1975, 1976, 1979) and elucidated further in Brill and Posner (1974, 1975, 1977, 1981), and Cohen (1976, 1977). These methods began as an essential part of system point theory and are also known as system point analysis, sample path analysis, or level crossing technique, approach, theory, or analysis in the literature (Brill 1975, 2008). Level crossing methods are very useful rate conservation techniques for stochastic models (Miyazawa 1994).

Model and Stationary Distribution

Consider a stochastic process $\{W(t), t \geq 0\}$ where both the parameter set and state space are continuous. The random variable $W(t)$ at time point t may denote the content of a dam with general efflux, the stock on hand in an (s,S) or (r,nQ) inventory system with stock decay, or the virtual wait or workload in a queue. Assume that upward jumps of $\{W(t)\}$ occur at Poisson rate λ_u and downward jumps at Poisson rate λ_d. Let upward and downward jump magnitudes have cumulative distribution function (CDF) B_u and B_d, respectively. Assume that the model parameters are such that the stationary distribution of $W(t)$ exists as $t \to \infty$. Let G and g denote the stationary CDF and probability density function (PDF), respectively. The aim here is to obtain expressions for g and G in terms of the model parameters by using a level crossing approach.

Sample Paths

A sample path of the $\{W(t)\}$ process is a right-continuous, real-valued function on the nonnegative reals whose value at time-point t is the realized value of random variable $W(t)$. Denote an arbitrary sample path by the function $X(t)$, $t \geq 0$. The function X has either jump or removable discontinuities on a sequence of strictly increasing time points $\{\tau_n, n = 0, 1, \ldots\}$, where $\tau_0 = 0$ without loss of generality. Typically, the time points $\{\tau_n\}$ represent input or output epochs in dams, arrival epochs in queues, or demand or replenishment epochs in inventories. Assume that when a sample path is positive valued, it decreases continuously on time segments between jump points, described by $dX(t)/dt = -rX(t), X(t) > 0, \tau_n \leq t < \tau_{n+1}$, $n = 0, 1, 2, \ldots$ wherever the derivative exists, and where $r(x) > 0$ for $x > 0$. Note that for the virtual wait process in queues, $r(x) = 1(x > 0)$ and $r(0) = 0$. In an (s,S) continuous review inventory system, where the stock on hand decays at constant rate k, then $r(x) = k$ for all x between the reorder level s and order-up-to-level S.

Level Crossing by Sample Paths

Let x denote a fixed state space level and t_0 an arbitrary positive time point. Let t_0 be one of the jump time points $\{\tau_n\}$, $n = 1, 2, \ldots$ and let d_0 and u_0 denote the corresponding downward and upward jump magnitudes, respectively, where at least one of u_0, d_0, is strictly positive. The sample path may down cross level x at $t_0 > 0$ if t_0 is any positive epoch, but it can up cross level x at t_0 only if t_0 is one of the $\{\tau_n\}$.

If a sample path down crosses level x at t_0 which is not one of the $\{\tau_n\}$, then the down crossing is a continuous down crossing, since the sample path is continuous at t_0. If a sample path down crosses level x at t_0 which is one of the $\{\tau_n\}$, then the downward jump of magnitude d_0 brings it from above x to a level below x. If a sample path up crosses level x at t_0, then, necessarily, t_0 is one of the epochs $\{\tau_n\}$, and the upward jump of magnitude u_0 brings it from below x to a level above x.

If both u_0 and d_0 are strictly positive at t_0 which is one of the $\{\tau_n\}$, the model mechanism would determine whether the downward or upward jump is considered to precede the other. In inventories without lead time, for example, stock depletions due to demands (downward jumps) precede stock replenishments (upward jumps). The jumps are not part of the sample path per se, but serve only to construct the path. One may also define level crossings at some time point t_0 by considering the net jump which has magnitude $|u_0 - d_0|$ and upward (downward) direction if $u_0 > d_0 (u_0 < d_0)$.

Level Crossings and the Stationary Distribution

Down crossings — Let $D^u_{ct}(x)$ denote the number of continuous down crossings of level x and $D^j_t(x)$, the number of jump down crossings of level x during $(0, t)$, $t > 0$. Then, for $r(x) = 1, x > 0$ and $r(0) = 0$, it follows with probability 1 that

$$\lim_{t\to\infty} \frac{D^c_t(x)}{t} = r(x)g(x) \quad (\text{for all } x), \tag{1}$$

(Brill 1975). The following also holds with probability 1:

$$\lim_{t\to\infty} \frac{D^j_t(x)}{t} = \lambda_d \int_{y=x}^{\infty} \bar{B}_d(y-x) g_d(y) dy \quad (\text{for all } x), \tag{2}$$

where g_{od} is the limiting PDF at embedded downward jump points as $t \to \infty$ and $\bar{B} \equiv 1 - B$.

Both Eqs. 1 and 2 also hold upon replacing $D^u_{ct}(x)$ and $D^u_{ty}(x)$ by their expectations, denoted by $E[D^c_t(x)]$ and $E[D^u_{ty}(x)]$, respectively, and deleting with probability 1. For exponentially distributed interarrivals between downward jumps (Poisson downward jumps), then $g_d \equiv g$, which is the *PASTA* principle.

Up crossings —Let $U^j_t(x)$ denote the number of jump up crossings of level x during $(0, t)$. Then, with probability 1,

$$\lim_{t\to\infty} \frac{U^j_t(x)}{t} = \lambda_u \int_{-\infty}^{x} \bar{B}_u(x-y) g_u(y) dy \qquad (\text{for all x}), \tag{3}$$

where g_u is the limiting PDF at embedded upward-jump time points as $t \to \infty$ (Brill 1975).

Formula (3) gives an expression for the long-run up crossing rate of level x by any typical sample path at upward jump points, in terms of an integral of the density g_u. For Poisson upward jumps, $g_u \equiv g$ by the PASTA principle.

A Conservation Law for Level Crossings

For each state space level, the following conservation law holds:

long run total down crossing rate = long run total up crossing rate.

This conservation law, together with Eqs. 1, 2 and 3, enables one to write an integral equation for the PDF g in which every term has a precise interpretation as a sample-path down or up crossing rate, namely,

$$r(x)g(x) + \lambda_d \int_{y=x}^{\infty} \bar{B}_d(y-x) g(y) dy = \lambda_u \int_{y=-\infty}^{x} \bar{B}_u(x-y) g(y) dy \qquad (\text{for all } x). \tag{4}$$

In (4), the left-hand side depicts the total sample path long-run down crossing rate of level x, while the right-hand side depicts the long-run up crossing rate of the level x. Equation (4) is then solved for g by using standard applied mathematics techniques.

Applicability

The level crossing technique is applicable to dams with limited capacity, blocked-input rules, various control level policies, etc.; to complex variants of M/G/1, M/M/c, G/M/1 queues with reneging, bounded virtual wait, server vacations, various state dependencies, cyclic-service queues; and to a wide class of inventory, production/inventory, counter, risk reserve, and related models.

The same level crossing ideas as in Eqs. 1, 2 and 3 have been applied to cycles in regenerative processes by Cohen (1976, 1977). Upon combining the regenerative-processes level crossing approach and the embedded level crossing technique of Brill (1976, 1979) with the previously widely known bubble diagram method (rate into a state = rate out of that state) for discrete state continuous time Markov chains, level crossing methods can be applied to obtain probability distributions and other characteristics in a broad class of stochastic models.

Level Crossing Estimation

The principle established in formula (1) motivates the idea of using $D^c_t(x)/[tr(x)]$ as an estimate for $g(x)$ when t is large. Level crossing estimation (also known as system point estimation) consists of three main steps: (I) simulating a single sample path over a large simulated time t; (ii) enumerating the continuous down crossings of all state space levels over $(0, t)$; and (iii) computing both point and interval estimates of g, G and the moments (Brill 1991).

See

- Inventory Modeling
- Markov Processes
- PASTA
- Queueing Theory

References

Azoury, K., & Brill, P. H. (1986). An application of the system-point method to inventory models under continuous review. *Journal of Applied Probability, 23*, 778–789.

Brill, P. H. (1975). System point theory in exponential queues. (*Ph.D. Dissertation*, University of Toronto).

Brill, P. H. (1976). Embedded level crossing processes in dams and queues. WP #76-022, Department of Industrial Engineering, University of Toronto.
Brill, P. H. (1979). An embedded level crossing technique for dams and queues. *Journal of Applied Probability, 16*, 174–186.
Brill, P. H. (1991). Estimation of stationary distributions in storage processes using level crossing theory. Proceedings of the Statistical Computing Section, American Statistical Association, 172–177.
Brill, P. H. (2008). *Level crossing methods in stochastic models*. New York: Springer.
Brill, P. H., & Posner, M. J. M. (1974). On the equilibrium waiting time distribution for a class of exponential queues. WP #74-012, Department of Industrial Engineering, University of Toronto.
Brill, P. H., & Posner, M. J. M. (1975). Level crossings in point processes applied to queues. WP #75-009, Department of Industrial Engineering, University of Toronto.
Brill, P. H., & Posner, M. J. M. (1977). Level crossings in point processes applied to queues: Single server case. *Operations Research, 25*, 662–673.
Brill, P. H., & Posner, M. J. M. (1981). The system point method in exponential queues: A level crossing approach. *Mathematics of Operations Research, 6*, 31–49.
Cohen, J. W. (1976). *On regenerative processes in queueing theory* (Lecture notes in economics and mathematical systems, p. 121). New York: Springer-Verlag.
Cohen, J. W. (1977). On up and down crossings. *Journal of Applied Probability, 14*, 405–410.
Miyazawa, M. (1994). Rate conservation laws: A survey. *Queueing Systems: Theory & Applications, 18*, 1–58.
Ross, S. (1985). *Introduction to probability models* (4th ed.). New York: Academic Press.

Level Curve

Also called isovalue contour: a curve along which the values of a given associated function remain constant.

See

► Isoquant

Lexicographic Ordering

An ordering of a set of vectors based on the lexicopositive (negative) properties of the vectors. For example, the sequence of vectors $\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_q\}$ is ordered in a lexicographic sense if $\boldsymbol{x}_i - \boldsymbol{x}_j$ is lexico-positive for $i > j$. Such orderings are similar to dictionary ordering of words and are used to prove finiteness of the simplex algorithm.

See

► Cycling
► Lexico-Positive (Negative) Vector

Lexico-Positive (Negative) Vector

A vector $\boldsymbol{x} = (x_1,\ldots, x_n)$ is called lexico-positive (negative) if $\boldsymbol{x} \neq 0$ and the first nonzero term is positive (negative). The vector $\boldsymbol{x}$ is lexico-negative if $-\boldsymbol{x}$ is lexico-positive. A vector $\boldsymbol{x}$ is greater than a vector $\boldsymbol{y}$ in a lexico-positive sense if $\boldsymbol{x} - \boldsymbol{y}$ is lexico-positive.

See

► Lexicographic Ordering

LGP

Linear goal programming.

See

► Goal Programming

Libraries

Arnold Reisman[1] and Xiaomei Xu[2]
[1]Reisman and Associates, Shaker Heights, OH, USA
[2]Cleveland, OH, USA

The American Heritage Dictionary of the English Language (1976, p. 753) defines a library is

"a repository for literary and artistic materials such as books, periodicals, newspapers, pamphlets, and prints kept for reading or reference." This rather classical notion of a library does not recognize the fact that libraries are now a subset of the broader field known as Information Systems (IS). Nevertheless, the scope of this article will be delimited to institutions which can be defined as above, albeit with some leeway.

The history of the application of operations research/management science to libraries is not very distinguished. Contributions in the library field were constrained up to and through the decade of the 1970s by the fact that few operations researchers chose libraries as a field of interest. Moreover, librarians have not sought out operations researchers to help in their problem solving, nor did they offer a particularly fertile environment for doing OR studies (Chen 1974). On the other hand, since the 1970s, computer science has made significant inroads into the library field by merging with library science to create local and extended area computer networks linking users with comprehensive databases.

The first known application of OR to libraries in the United States can be credited to Bacon and Machol (1958). The 1960s recorded a more widespread interest (Cox 1964; Morse 1968; Cook 1968). A comprehensive review on library operations research was done by Kantor (1979). In that review, Kantor summarized all of the previous review articles. Most noteworthy of these from the OR point of view are the bibliographies by Slamecka (1972) and Kraft and McDonald (1977), and surveys and/or assessments by Bommer (1975), Kraft and McDonald (1976), Leimkuhler (1970, 1972, 1977a, 1977b), Churchman (1972) and Morse (1972).

Literature on utilization of OR in libraries has classified the field in several different ways. Kantor (1979) classified papers and projects into the following groups according to the purpose of the research: system description; modeling the system; parameter identification; optimization or multi-valuation; and application. Rowley and Rowley (1981) classified the work by the nature of the research (recurrent problems, on/off decisions, etc.). For the purposes of this article, a three-dimensional classification is used with one of the dimensions adopting Rowley's (1981) classification, with slight modifications. Based on the type of problems being analyzed, the application areas are operational or recurrent problems, such as book storage problems; strategies or on/off decisions, such as library location problems; and control/design problems, such as loan policy problems (Rowley and Rowley 1981).

The second dimension on the application of OR in libraries is a classification according to the type of OR techniques used:

1. *Queueing models* – Given the average book circulation time ($1/\mu$) and the mean number of persons who borrow the book (λ), the expected circulation rate of that particular book is derived using queueing theory (Morse 1968).
2. *Simulation* – With the number of staff, the volumes of various jobs (users' requests, new issues, overdue fees, etc.) and the job processing times specified, simulation is used to estimate the delays, processing times and utilization of each member of staff and the whole facility (Thomas and Robertson 1975).
3. *Facility location algorithms* – The library facilities and relocation problems are discussed by Min (1988).
4. *Mathematical programming* – If there are two types of information services, both of which share the same set of resources (staff time in scanning, indexing, abstracting, etc.), and each of them has a different unit profit, a linear programming problem is used to find out how many services of each type to produce to maximize the total profit (Rowley and Rowley 1981, *58–64*).
5. *Network flow models* – Given the heights and thicknesses of a given collection of books and the cost of different shelf heights, a network model is developed to determine the optimal number of shelf heights for minimizing shelving costs through finding the shortest path in a directed network (Gupta and Ravindram 1974).
6. *Decision theory* – A decision regarding whether or not to install a library security system is addressed given the installation cost and the probabilities of success and failure (Rowley and Rowley 1981, 91–92).
7. *Search theory* – Patterns of browsing in libraries are addressed in Morse (1970).
8. *Transportation models* – A routing problem is explored for a vehicle delivering materials to branches (Heinritz and Hsiao 1969; McClure 1977).

9. *Inventory control theory* – An EOQ model is used to determine the optimal order quantity for the stock of a certain library supply (Rowley and Rowley 1981, 111–116).
10. *Probability and statistics* – Library book circulation and individual book popularities are considered as probabilistic processes by Gelman and Sichel (1987) who demonstrated the superiority of beta over the negative binomial distribution.
11. *Benefit cost analysis* – Library planning is addressed by Leimkuhler and Cooper (1971).

Each of these categories could be, in turn, further characterized by whether or not the research work was grounded, e.g., based on real world library systems involving real data and/or bona fide librarians in the study as opposed to models which were basically what might be called logico/deductive. A more thorough discussion is given in Reisman and Xu (1994), where Table I, page 37, provides a taxonomic review of the vast bulk of the literature in the field.

As can be seen from the above delineation and the referenced table, the utilization of OR in libraries is far from achieving its full potential. Except for simulation and probability and statistics based applications, the bulk of the literature is not well grounded in real life settings. The literature reflects the gap between the complex mathematical models in OR and the usually not very quantitatively educated library workers (Stueart and Moran 1987). To enhance the application of OR in libraries, Bommer (1975) suggested a closer working relationship between operations researchers and library managers.

See

► Information Systems and Database Design in OR/MS

References

Bacon, F. R. Jr., & Machol, R. E. (1958). *Feasibility analysis and use of remote access to library card catalogs*. Paper, presented at the Fall meeting of ORSA (Unpublished).

Bacon, F. R., Jr., Churchill, N. C., Lucas, C. J., Maxfield, D. K., Orwant, C. J., & Wilson, R. C. (1958). *Applications of a teller reference system to divisional library card catalogues: A feasibility analysis*. Ann Arbor, MI: Engineering Research Institute, University of Michigan.

Bommer, M. (1975). Operations research in libraries: A critical assessment. *Journal of the American Society for Information Science, 26*, 137–139.

Chen, Ching-chih. (1974). *Applications of operations research models to libraries: A case study of the use of monographs in the Francis A. Countway Library of Medicine, Harvard University*. Unpublished Ph.D. dissertation, Case Western Reserve University, School of Library Science, Cleveland, OH.

Churchman, C. W. (1972). Operations research prospects for libraries: The realities and ideals. *Library Quarterly, 42*, 6–14.

Cook, J. J. (1968). Increased seating in the undergraduate library: A study in effective space utilization. In B. R. Burkhalter (Ed.), *Case studies in systems analysis in a university library* (pp. 142–170). Metuchen, NJ: Scarecrow Press.

Cox, J. G. (1964). *Optimal storage of library material*. Unpublished Ph.D. dissertation, Purdue University Libraries, Lafayette, Indiana.

Gelman, E., & Sichel, H. S. (1987). Library book circulation and the beta-binomial distribution. *Journal of the American Society for Information Science, 38*, 4–12.

Gupta, S. M., & Ravindram, A. (1974). Optimal storage of books by size: An operations research approach. *Journal of the American Society for Information Science, 25*, 354–357.

Heinritz, F. J., & Hsiao, J. C. (1969). Optimum distribution of centrally processed material. *Library Resources and Technical Services, 13*, 206–208.

Kantor, P. (1979). Review of library operations research. *Library Research, 1*, 295–345.

Kraft, D. H., & McDonald, D. D. (1976). Library operations research: Its past and our future. In D. P. Hammer (Ed.), *The information age* (pp. 122–144). Metuchen, NJ: Scarecrow Press.

Kraft, D. H., & McDonald, D. D. (1977). Library operations research: A bibliography and commentary of the literature. *Information, Reports and Bibliographies, 6*, 2–10.

Leimkuhler, F. F. (1970). Library operations research: An engineering approach to information problems. *Engineering Education, 60*, 363–365.

Leimkuhler, F. F. (1972). Library operations research: A process of discovery and justification. *Library Quarterly, 42*, 84–96.

Leimkuhler, F. F. (1977a). Operational analysis of library systems. *Information Processing and Management, 13*, 79–93.

Leimkuhler, F. F. (1977b). Operations research and systems analysis. In F. W. Lancaster & C. W. Cleverdon (Eds.), *Evaluation and scientific management of libraries and information centres* (pp. 131–163). Leyden, The Netherlands: Nordhoff.

Leimkuhler, F. F., & Cooper, M. D. (1971). Analytical models for library planning. *Journal of the American Society for Information Science, 22*, 390–398.

McClure, C. R. (1977). Linear programming and library delivery systems. *Library Resources and Technical Services, 21*, 333–344.

Min, H. (1988). The dynamic expansion and relocation of capacitated public facilities: A multi-objective approach. *Computers and Operations Research (UK), 15*, 243–252.
Morse, P. M. (1968). *Library effectiveness: A systems approach*. Cambridge, MA: MIT Press.
Morse, P. M. (1970). Search theory and Browsing. *Library Quarterly, 40*, 391–408.
Morse, P. M. (1972). Measures of library effectiveness. *Library Quarterly, 42*, 15–30.
Reisman, A., & Xu, X. (1994). Operations research in libraries: A review of 25 years of activity. *Operations Research, 42*, 34–40.
Rowley, J. E., & Rowley, P. J. (1981). *Operations research: A tool for library management* (pp. 3–4). Chicago: American Library Association.
Slamecka, V. (1972). A selective bibliography on library operations research. *Library Quarterly, 42*, 152–158.
Stueart, R. D., & Moran, B. B. (1987). *Library management* (3rd ed., pp. 200–202). Littleton, CO: Libraries Unlimited.
Thomas, P. A., & Robertson, S. E. (1975). A computer simulation model of library operations. *Journal of Documentation, 31*, 1–16.

LIFO

The Last-In, First-Out queue discipline in which customers are selected for service in reverse order of their arrival (meant to be equivalent to the last-come, first-served scheme).

See

- LCFS
- Queueing Theory

Light-Tailed Distribution

A probability distribution that has an exponentially decaying complementary CDF, e.g., the normal (Gaussian) and exponential distributions.

See

- Heavy-Tailed Distribution

Likelihood Ratio Method

A method for gradient estimation in simulation used for sensitivity analysis and optimization; also known as the score function method.

See

- Perturbation Analysis
- Score Functions
- Simulation Optimization

Limiting Distribution

Let $p_{ij}(t)$ be the probability that a stochastic process takes on value j at time t (discrete or continuous), given that it began at time 0 from state i. If for each j, $p_{ij}(t)$ approaches a limit p_j as $t \to \infty$ independent of i, the set $\{p_j\}$ is called the limiting or steady-state distribution of the process. For Markov chains in discrete time, the existence of a limiting distribution implies that there is a stationary (or invariant) distribution found from $\boldsymbol{\pi} = \boldsymbol{\pi P}$, where $\boldsymbol{P}$ is the single-step transition matrix, such that $\boldsymbol{\pi} = \boldsymbol{p}$. Similarly, for continuous-time chains, the steady-state distribution is the probability vector satisfying the global balance equations $\boldsymbol{\pi Q} = \boldsymbol{0}$, where $\boldsymbol{Q}$ is the transition rate matrix.

See

- Markov Chains
- Markov Processes
- Stationary Distribution
- Statistical Equilibrium

Lindley's Equation

An integral equation for the steady-state waiting-time distribution in the first-come, first-served, single-server G/G/1 queue. If $W_q(x)$, $x \geq 0$, is the

steady-state distribution function of the delay or waiting time in the queue, then, for $x \geq 0$,

$$W_q(x) = \int_{-\infty}^{x} W_q(x-y)dU(y)$$

with $W_q(x) = 0$ for $x < 0$, where the function $U(y)$ is the distribution function of the random variable defined as the service time minus the interarrival time.

Lindley's equation can also be used to refer to the finite-time transient recursive equation relating delays in the first-come, first-served, single-server G/G/1 queue as follows:

$$D_{n+1} = \max(0, D_n + S_n - A_n),$$

where D_n is the delay of the nth arriving customer, S_n is the service time of the nth arriving customer, and A_n is the interarrival time between the nth and $(n + 1)$st arriving customer.

See

- ▶ Kendall's Notation
- ▶ Queueing Theory

Line

A line is the set of points $\{x|x = (1-\lambda)x_1 + \lambda x_2\}$, where x_1 and x_2 are points in n-dimensional space and λ is a real number. The line passes through the points x_1 and x_2, $x_1 \neq x_2$.

Line Segment

The straight line joining any two points in n-dimensional real space is a line segment. More specifically, if x_1 and x_2 are the two points, then the set of points $\{x|x = (1-\lambda)x_1 + \lambda x_2, 0 \leq \lambda \leq 1\}$ is the line segment joining x_1 and x_2.

See

- ▶ Line

Linear Combination

For a set of vectors $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, a linear combination is another vector $\boldsymbol{y} = \sum_j \alpha_j \boldsymbol{x}_j$, where the scalar coefficients α_j can take on any values.

Linear Equation

The mathematical form $a_1x_1 + a_2x_2 + \ldots + a_nx_n = b$ is a linear equation, where the a_j and b can take on any values.

See

- ▶ Hyperplane

Linear Functional

A linear functional $f(\boldsymbol{x})$ is a real-valued function defined on an n-dimensional vector space such that, for every vector $\boldsymbol{x} = \alpha\boldsymbol{u} + \beta\boldsymbol{v}$, $f(\boldsymbol{x}) = f(\alpha\boldsymbol{u} + \beta\boldsymbol{v}) = \alpha f(\boldsymbol{u}) + \beta f(\boldsymbol{v})$ for all n-dimensional vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ and all scalars α and β.

Linear Inequality

The mathematical form $a_1x_1 + a_2x_2 + \ldots + a_nx_n \leq b$ or $a_1x_1 + a_2x_2 + \ldots + a_nx_n \geq b$ is a linear inequality, where the numbers a_j and b can take on any values. The set of vectors $\boldsymbol{x} = (x_1, \ldots, x_n)$ that satisfy the inequality form a solution half space.

See

- ▶ Hyperplane

Linear Programming

Frederick S. Hillier
Stanford University, Stanford, CA, USA

Introduction

Linear programming is one of the most widely used techniques of operations research and management science. Its name means that planning (programming) is being done with a mathematical model (called a linear-programming model) where all the functions in the model are linear functions.

Linear Programming Models

Linear programming models come in a variety of forms. To illustrate one common form, consider the problem of determining the most profitable mix of products for a manufacturer. Let n be the number of possible products. For each product j ($j = 1, 2, \ldots, n$), a decision variable x_j is introduced to represent the decision on its production rate. Let c_j be the profit per unit of product j produced, and let Z be the total rate of profit resulting from the choice of product mix. This choice is constrained by the limited capacities of the production facilities available for these products. Let m be the number of different types of facilities needed. For each type i ($i = 1, 2, \ldots, m$), let b_i be the amount of capacity available per unit time and let a_{ij} be the amount of capacity used by each unit produced of product j ($j = 1, 2, \ldots, n$). The resulting linear programming model then is to choose $x_1, x_2, \ldots, x_n$ so as to

$$\begin{aligned} \text{Maximize } Z &= c_1x_1 + c_2x_2 + \cdots + c_nx_n \\ \text{subject to : } & a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \le b_1 \\ & a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \le b_2 \\ & \vdots \qquad\qquad \vdots \\ & a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \le b_m \end{aligned}$$

and

$$x_1 \ge 0, x_2 \ge 0, \ldots, x_n \ge 0.$$

The linear function being maximized in this model is called the objective function. The m inequalities with a linear function on the left-hand side are referred to as functional constraints (or structural constraints), and the inequalities in the bottom row are nonnegativity constraints. The constants (c_j, b_i, and a_{ij}) are the parameters of the model. Any choice of values of ($x_1, x_2, \ldots, x_n$) is called a solution, whereas a solution satisfying all the constraints is a feasible solution, and a feasible solution that maximizes the objective function is an optimal solution.

Many other applications of linear programming having nothing to do with product mix also fit this same form for the model. In these cases, activities of some other kind replace the production of products, and resources of some other kind replace production facilities. For each activity j ($j = 1,2, \ldots, m$), the decision variable x_j represents the decision on the level of that activity. The problem then is to allocate these limited resources to these interrelated activities so as to obtain the best mix of activities (i.e., an optimal solution) according to the overall measure of performance adopted for the objective function.

Another common form for a linear programming model is to minimize the objective function, subject to functional constraints with $\ge$ signs and nonnegativity constraints. A typical interpretation then is that the objective function represents the total cost for the chosen mix of activities and the functional constraints involve different kinds of benefits. In particular, the function on the left-hand side of each functional constraint gives the level of a particular kind of benefit that is obtained from the mix of activities, and the constant on the right-hand represents the minimum acceptable level for that benefit. The problem then is to determine the mix of activities that gives the best tradeoff between cost and benefits according to the model.

Still other linear-programming models have an equality instead of inequality sign in some or all of the functional constraints. Such constraints represent fixed requirements for the value of the function on the left-hand side.

It is fairly common for large linear-programming models to include a mixture of functional constraints — some with $\le$ signs, some with $\ge$ signs, and some with $=$ signs. Nonnegativity constraints always have a $\ge$ sign, but it occasionally is appropriate to delete this kind of constraint for some or all of the decision variables.

Successful applications of linear programming sometimes use very large models. As described in a later section, exceptionally efficient algorithms are available for solving these models. When using state-of-the-art implementations of these algorithms and a powerful desktop computer or workstation, a model with several thousand functional constraints and decision variables is considered to be of moderate size. Having a few tens of thousands of functional constraints and even more decision variables is not considered particularly large. Far bigger problems with millions of functional constraints and decision variables sometimes are solved, depending largely on whether they have a special structure that can be exploited.

With large models, it is inevitable that mistakes and faulty decisions will be made initially in formulating the model and inputting it into the computer. Therefore, a thorough process of testing and refining the model, i.e., model validation, is needed. The usual end-product is not a single static model, but rather a long series of variations on a basic model to examine different scenarios as part of post-optimality analysis (discussed later). A sophisticated modeling language usually is needed to efficiently formulate the model and then to expedite a number of model management tasks, including accessing data, transforming data into model parameters, modifying the model whenever desired, and analyzing solutions from the model.

Some Applications of Linear Programming

The applications of linear programming have been remarkably diverse. They all involve determining the best mix of activities, where the decision variables represent the levels of the respective activities, but these activities arise in a wide variety of contexts. In the context of financial planning, the activities might be investing in individual stocks and bonds (portfolio selection), or undertaking capital projects (capital budgeting), or drawing on sources for generating working capital (financial-mix strategy). In the context of marketing analysis, the activities might be using individual types of advertising media, or performing marketing research in segments of the market. In the context of production planning, applications range widely from the product-mix problem (discussed earlier) to the blending problem (determining the best mix of ingredients for various individual final products), and from production scheduling to personnel scheduling.

In addition to manufacturing, these kinds of production planning applications also arise in agricultural planning, health-care management, the planning of military operations, policy development for the use of natural resources, etc.

Linear programming has had a great impact on improving the efficiency and profitability of numerous organizations around the world. A considerable number of these applications have won a prestigious prize in the annual international competition for the Franz Edelman Award for Achievement in Operations Research and the Management Sciences. To mention a few typical award-winning applications: Bixby et al. (2006) describe how Swift & Company saved $12 million in 1 year by optimizing its product mix while dynamically scheduling its beef-fabrication operations at five plants in real time as it receives orders; Lee and Zaider (2008) discuss how a breakthrough in optimizing the application of brachytherapy to prostrate cancer is having a profound impact on both health care costs (potentially saving $500 million annually) and quality of life for treated patients; Holloran and Bryne (1986) were early pioneers in applying linear programming at United Airlines to design the work schedules for all the employees at the various reservation offices and airports, thereby saving the company more than $6 million annually; Leachman, Kang, and Lin (2002) describe how Samsung Electronics Corp. captured an additional $200 million in annual sales revenue by using a linear- programming model with tens of thousands of decision variables and functional constraints to increase the efficiency of its processes for manufacturing random access memory devices. Hillier and Lieberman (2010, Chap. 3) also reference other award-winning applications of linear programming.

Another important kind of application of linear programming arises from its close relationship to several other important areas of operations research and management science, including integer programming, nonlinear programming, and game theory. Linear programming often is useful to help solve problems in these other areas as well.

Some Special Types of Linear Programming Models

One particularly important special type of linear programming problem is the transportation problem. A typical application of the transportation problem is to determine how a corporation should distribute a product from its various factories to various distributors. In particular, given the amount of the product produced at each factory and the amount needed by each distributor, one can determine how much to ship from each factory to each distributor in order to minimize total shipping cost. Other applications extend to areas such as production scheduling.

Camm et al. (1997) describe an award-winning application of the transportation problem at Procter & Gamble that saved over $200 million annually by redesigning the company's production and distribution system for its North American operations. A major part of the study revolved around formulating and solving transportation problems for individual product categories.

The assignment problem is a special type of linear-programming problem where assignees are being assigned to perform tasks. For example, the assignees might be employees who need to be given work assignments. Assigning people to jobs is a common application of the assignment problem. However, the assignees need not be people. They also could be machines, or vehicles, or plants, or even time slots to be assigned tasks. It can be shown that the mathematical structure of the model for the assignment problem is a special case of that for the transportation problem.

Both the transportation problem and the assignment problem are a special case of another key type of linear-programming problem, called the minimum-cost network-flow problem, that involves determining how to distribute goods through a distribution network at a minimum total cost. In particular, the nodes of this network include at least one supply node and at least one demand node, and then the rest of the nodes are transshipment nodes. Given the capacity of each arc for transmitting flow, the objective is to minimize the total cost of sending the supply from the supply nodes through the network to satisfy the given demand at the demand nodes.

Klingman et al. (1987) describe a classic award-winning application of this type at the Citgo Petroleum Corporation. This minimum-cost network-flow problem involved the distribution of petroleum products through a distribution network consisting of pipelines, tankers, barges, and hundreds of terminals. This application is credited with saving the company well over $15 million annually. (Another application of linear programming involving Citgo's refinery operations was implemented at about the same time and achieved additional savings of about $50 million per year).

Another special case of the minimum-cost network-flow problem is the maximum-flow problem. Given a connected network with capacity constraints on the maximum flow through each arc, the objective now is to maximize the flow through the network from the source node to the sink node. Some typical applications include maximizing the flow through a distribution network, or through a supply network, or through a system of pipelines, or through a system of aqueducts, or through a transportation network.

The shortest-path problem (also called the shortest-route problem) is still another important special type of linear-programming problem that is also a special case of the minimum-cost network-flow problem. The objective now is to find the path through a network from an origin to a destination that minimizes the total distance traveled. Arc distances also can represent costs or times so the objective becomes to minimize the total cost or total time of a sequence of activities.

Ireland et al. (2004) describe how the Canadian Pacific Railway saves roughly $100 million annually by using network optimization techniques to route its freight each day over a massive rail network that encompasses much of North America. Numerous shortest-path problems are solved each day as part of the overall approach for this award-winning application.

There have been many other award-winning applications of the special types of linear-programming problems that are described above. Hillier and Lieberman (2010, Chap. 9) reference some of these applications.

Solving Linear Programming Models

Two crucial events have been primarily responsible for the great impact of linear programming since its

emergence in the middle of the twentieth century. One was the invention in 1947 by George Dantzig of a remarkably efficient algorithm, called the simplex method, for finding an optimal solution for a linear-programming model. The second crucial event was the computer revolution that makes it possible for the simplex method to solve huge problems.

The simplex method exploits some basic properties of optimal solutions for linear programming models. Because all the functions in the model are linear functions, the set of feasible solutions (called the feasible region) is a convex polyhedral set. The vertices (extreme points) of the feasible region play a special role in finding an optimal solution. A model will have an optimal solution if it has any feasible solutions (all the constraints can be satisfied simultaneously) and the constraints prevent improving the value of the objective function indefinitely. Any such model must have either exactly one optimal solution or an infinite number of them. In the former case, the one optimal solution must be a vertex of the feasible region. In the latter case, at least two vertices must be optimal solutions, and then all convex-linear combinations of these vertices also are optimal. It is sufficient, therefore, to find the vertices with the most favorable value of the objective function in order to identify all optimal solutions.

Based on these facts, the simplex method is an iterative algorithm that only examines vertices of the feasible region. At each iteration, it uses algebraic procedures to move along an outside edge of the feasible region from the current vertex to an adjacent vertex that is better. The algorithm terminates (except perhaps for checking ties) when a vertex is reached that has no better adjacent vertices, because the convexity of the feasible region then implies that this vertex is optimal.

The simplex method is an exponential-time algorithm (in the worst case). However, it consistently has proven to be very efficient in practice. Running time tends to grow approximately with the cube of the number of functional constraints, and less than linearly with the number of variables. Problems with many thousands of functional constraints and a larger number of decision variables are routinely solved. One key to its efficiency on such large problems is that the path followed generally passes through only a tiny fraction of all vertices before reaching an optimal solution. The number of iterations (vertices traversed) generally is of the same order of magnitude as the number of functional constraints.

The running time of the simplex method also is greatly affected by the degree of sparsity of the matrix of constraint coefficients, where the measure of sparsity is the proportion of the coefficients that are not zero. Having a very sparse coefficient matrix (say, less than 1%) can greatly accelerate the simplex method.

There also exist useful variants of the simplex method, including especially the dual simplex method, that sometimes are used to solve linear-programming problems. (Using the terminology introduced at the beginning of the next section, the dual simplex method operates on the primal problem as if the simplex method is being applied simultaneously to the dual problem).

In addition, specialized versions of the simplex method also are available for exploiting the special structure in some of the special types of linear-programming problems described in the preceding section. In particular, the network-simplex method does this for the minimum-cost network-flow problem and the transportation-simplex method does it for the transportation problem. A variety of special algorithms also are available for the assignment problem, the maximum-flow problem, and the shortest-path problem. Therefore, even though the general simplex method can solve huge instances of these problems, these special purpose algorithms can solve even vastly larger instances.

Any of the various textbooks on linear programming cited in the references will provide additional details about the simplex method and these related algorithms.

Some 37 years after the invention of the simplex method, N. Karmarkar (1984) created great excitement in the operations research/management science community by announcing a new polynomial-time algorithm for linear programming, along with claims of being many times faster than the simplex method. Actually, the first polynomial-time algorithm for linear programming had been announced earlier by L. G. Khachiyan (1979), but his ellipsoid method proved to be not nearly competitive with the simplex method in practice. Karmarkar's algorithm moves through the interior of the feasible region until it converges to an optimal solution, and so is referred to as an

interior-point method. The announcement did not include details needed for computer implementation.

Following Karmarkar's announcement, there was a long flurry of research activity to fully develop and refine similar interior-point methods, along with sophisticated computer implementations. The application of these methods to linear programming now has reached a high level of sophistication. These methods commonly are called barrier methods or barrier algorithms because they are based on introducing a logarithmic barrier function. A specific barrier algorithm then may be given a specific name to identify its main features. For example, the primal-dual predictor-corrector algorithm developed by Mehrotra (1992) established a structure that has commonly been adopted by subsequent algorithms. Ye (1997), Vanderbei (2008), and Luenberger and Ye (2008) provide further details about the interior-point approach.

A key feature of the interior-point approach is that both the number of iterations (trial solutions) and total running time tend to grow very slowly (even more slowly than for the simplex method) as the problem size is increased. Therefore, the best implementations of this approach tend to become faster than the simplex method (or the dual simplex method) for relatively large problems. This is not always true, because the efficiency of each approach depends greatly in different ways on the special structure in each individual problem. Indeed, one of the by-products of the emergence of the interior-point approach has been a major renewal of efforts to improve the efficiency of computer implementations of the simplex method and its variants. Impressive progress has been made. Consequently, when tests have been conducted to determine when a leading barrier algorithm, the simplex method, or the dual simplex method will solve various huge problems more quickly, the dual simplex method or simplex method occasionally wins. As time goes on, improving computer technology (such as massive parallel processing) will substantially increase the size of problems that any of the algorithms can solve.

A considerable number of excellent software packages for linear programming and its extensions now are available to fill a variety of needs. Leading packages include CPLEX, Express-MP, Gurobi, and LINDO. Frontline Systems also has excellent solvers, including its Risk Solver Platform, for use with Excel spreadsheets.

As mentioned earlier, when dealing with large linear-programming problems, modeling languages also are needed to efficiently input, formulate, and manage the model. The available modeling languages include AMPL, MPL, OPL, GAMS, and LINGO. These languages are designed to be integrated with the kinds of solvers mentioned in the preceding paragraph.

Duality Theory and Postoptimality Analysis

Associated with any linear-programming problem is another linear-programming problem called the dual. Furthermore, the relationship between the original problem (called the primal) and its dual is a symmetric one, so that the dual of the dual is the primal. For example, consider the two related linear-programming models shown below in matrix notation (where $\boldsymbol{A}$ is a matrix, $\boldsymbol{c}$ and $\boldsymbol{y}$ are row vectors, $\boldsymbol{b}$, $\boldsymbol{x}$, and the null vector $\boldsymbol{0}$ are column vectors, all with compatible dimensions, and $\boldsymbol{x}$ and $\boldsymbol{y}$ are the decision vectors):

Maximize $\boldsymbol{cx}$	Minimize $\boldsymbol{yb}$
subject to: $\boldsymbol{Ax} \leq \boldsymbol{b}$	subject to: $\boldsymbol{yA} \geq \boldsymbol{c}$
and $\boldsymbol{x} \geq \boldsymbol{0}$.	and $\boldsymbol{y} \geq \boldsymbol{0}$.

For each of these problems, its dual is the other problem.

There are many useful relationships between the primal and dual problems, so the dual provides considerable information for analyzing the primal. This is especially helpful when conducting postoptimality analysis, i.e., analysis done after finding an optimal solution for the initial validated version of the model. A key part of most linear-programming studies, this analysis addresses a variety of what-if questions of interest to the decision makers. The purpose is to explore various scenarios about future conditions that may deviate from the initial model. The dual simplex method frequently is helpful for quickly re-optimizing these revised models.

Although the parameters of the given linear-programming model are treated as constants, they frequently represent just best estimates of a quantity whose true value may turn out to be quite

different. A key part of postoptimality analysis is sensitivity analysis, an investigation of the parameters to determine which ones are sensitive parameters, i.e., those that change the optimal solution if a small change is made in the given parameter value, and exploring the implications. For certain parameters, the decision makers may have some control over its value (e.g., the amount of a resource to be made available), in which case sensitivity analysis guides the decision on which value to choose. An extension of sensitivity analysis called parametric programming enables systematic investigation of simultaneous changes in various parameters over ranges of values.

Fletcher et al. (1999) present an interesting case study of how an OR team at the Pacific Lumber Company made extensive use of detailed sensitivity analysis to develop a sustained yield plan for the company's entire landholding. This plan is credited with increasing the company's present net worth by over $398 million while also generating a better mix of wildlife habitat acres.

Extensions of the simplex method are well suited for performing these kinds of postoptimality analysis. However, this is less true for interior-point methods. Therefore, even when an interior-point method is used to find an optimal solution, a switch may be made to the simplex method for subsequent analysis.

When there is substantial uncertainty about what the true values of the parameters will turn out to be, it may be necessary to use a different analysis approach, called linear programming under uncertainty, in which some or all the parameters are treated as random variables. This is especially pertinent when planning must be done for multiple time periods into an uncertain future. For example, Infanger (1993) discusses solving large-scale multi-stage stochastic linear programs.

Further Reading

Dantzig (1982) describes some of the early history of linear programming. Gass (1990) gives an entertaining introduction to the field. Hillier and Lieberman (2010) expand on all the topics mentioned here at an elementary level, and F.S. Hillier and M.S Hillier (2011) emphasize the application of linear programming from a managerial viewpoint. Dantzig (1963) provides the classic textbook on the theory of linear programming. Other excellent textbooks on linear programming and its extensions include Bertsimas and Tsitsiklis (1997), Dantzig and Thapa (1997, 2003), Vanderbei (2008), Luenberger and Ye (2008), Murty (2010), and Bazaraa, Jarvis and Sherali (2010), Marsten, Subramanian, Saltzman, Lustig, and Shanno (1990) discuss the basic concepts underlying interior-point methods.

See

- Algebraic Modeling Languages for Optimization
- Assignment Problem
- Basis
- Computational Complexity
- Density
- Duality Theorem
- Game Theory
- Hierarchical Production Planning
- Integer and Combinatorial Optimization
- Interior-Point Methods for Conic-Linear Optimization
- Mathematical Model
- Model Management
- Multiplier Vector
- Nonlinear Programming
- Parametric Programming
- Postoptimal Analysis
- Primal Problem
- Sensitivity Analysis
- Simplex Method (Algorithm)
- Simplex Tableau
- Stochastic Programming
- Transportation Problem
- Verification, Validation, and Testing of Models

References

Bazarra, M. S., Jarvis, J. J., & Sharali, H. D. (2010). *Linear programming and network flows* (4th ed.). New York: Wiley.

Bertsimas, D. M., & Tsitsiklis, J. N. (1997). *Linear optimization*. Belmont, MA: Athena Scientific.

Bixby, A., Downs, B., & Self, M. (2006). A scheduling and capable-to-promise application for swift & company. *Interfaces, 36*(1), 39–50.

Camm, J. D., Chorman, T. E., Dill, F. A., Evans, J. R., Sweeney, D. J., & Wegryn, G. W. (1997). Blending OR/MS, judgment, and GIS: Restructuring P&G's supply chain. *Interfaces, 27*(1), 128–142.
Dantzig, G. B. (1963). *Linear programming and extensions*. Princeton, NJ: Princeton University Press.
Dantzig, G. B. (1982). Reminiscences about the origins of linear programming. *Operation Research Letters, 1*, 43–48.
Dantzig, G. B., & Thapa, M. N. (1997). *Linear programming 1: Introduction*. New York: Springer.
Dantzig, G. B., & Thapa, M. N. (2003). *Linear programming 2: Theory and extensions*. New York: Springer.
Fletcher, L. R., Alden, H., Holmen, S. P., Angelis, D. P., & Etzenhouser, M. J. (1999). Long-Term forest ecosystem planning at pacific lumber. *Interfaces, 29*(1), 90–112.
Gass, S. I. (1990). *An illustrated guide to linear programming*. New York: Dover.
Hillier, F. S., & Hillier, M. S. (2011). *Introduction to management science: A modeling and case studies approach with spreadsheets* (4th ed.). Burr Ridge, IL: Irwin/McGraw-Hill.
Hillier, F. S., & Lieberman, G. J. (2010). *Introduction to operations research* (9th ed.). New York: McGraw-Hill.
Holloran, T. J., & Byrne, J. E. (1986). United airlines station manpower planning system. *Interfaces, 16*(1), 39–50.
Infanger, G. (1993). *Planning under uncertainty: Solving large-scale stochastic linear programs*. Danvers, MA: Boyd and Fraser.
Ireland, P., Case, R., Fallis, J., Van Dyke, C., Kuehn, J., & Meketon, M. (2004). The Canadian pacific railway transforms operations by using models to develop its operating plans. *Interfaces, 34*(1), 5–14.
Karmarker, N. K. (1984). A New Polynomial-time Algorithm for Linear Programming. *Combinatorica, 4*, 373–395.
Khachiyan, L. G. (1979). A polynomial algorithm for linear programming. *SSSR Doklady Akademii Nauk, 244*, 1093–1096. Translated in Soviet Math. Doklady 20 (1979), 191–194.
Klingman, D., Phillips, N., Steiger, D., & Young, W. (1987). The successful deployment of management science throughout Citgo Petroleum Corporation. *Interfaces, 17*(1), 4–25.
Leachman, R. C., Kang, J., & Lin, Y. (2002). SLIM: Short cycle time and low inventory in manufacturing at Samsung Electronics. *Interfaces, 32*(1), 61–77.
Lee, E. K., & Zaider, M. (2008). Operations research advances cancer therapeutics. *Interfaces, 38*(1), 5–25.
Luenberger, D. G., & Ye, Y. (2008). *Linear and nonlinear programming* (3rd ed.). New York: Springer.
Marsten, R., Subramanian, R., Saltzman, M., Lustig, I., & Shanno, D. (1990). Interior point methods for linear programming: Just call Newton, Lagrange, and Fiacco and McCormick! *Interfaces, 20*(4), 105–116.
Mehrotra, S. (1992). On the implementation of a primal-dual interior point method. *SIAM Journal on Optimization, 2*(4), 575–601.
Murty, K. G. (2010). *Optimization for decision making: Linear and quadratic models*. New York: Springer.
Vanderbei, R. J. (2008). *Linear programming: Foundations and extensions* (3rd ed.). New York: Springer.
Ye, Y. (1997). *Interior point algorithms*. New York: Wiley.

Linear-Fractional Programming Problem

The linear-fractional programming problem is one in which the objective to be maximized is of the form $f(\boldsymbol{x}) = (\boldsymbol{cx} + \alpha)/(\boldsymbol{dx} + \beta)$ subject to $\boldsymbol{Ax} \leq \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$, where α and β are scalars, $\boldsymbol{c}$ and $\boldsymbol{d}$ are row vectors of given numbers, and $\boldsymbol{b}$ is the right-hand-side vector. The problem can be converted to an equivalent linear programming problem by the translation $\boldsymbol{y} = \boldsymbol{x}/(\boldsymbol{dx} + \beta)$, provided that $\boldsymbol{dx} + \beta$ does not change sign in the feasible region.

See

► Fractional Programming

Lipschitz Continuous

A function $f(x)$ is said to be Lipschitz continuous if there exists a real constant $K > 0$ (called the Lipschitz constant) such that for every pair of points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, $||f(\boldsymbol{x}_1) - f(\boldsymbol{x}_2)|| \leq K||\boldsymbol{x}_1 - \boldsymbol{x}_2||$. If $K < 1$, then the function is called a contraction.

Little's Law

Susan Albin
Rutgers, The State University of New Jersey, Piscataway, NJ, USA

Little's Law, among the most fundamental and useful formulas in queueing theory, relates the number of customers in a queueing system to the waiting time of customers for a system in steady state as

$$L = \lambda W$$

- $L =$ The average number of customers in the system including customers in service
- $\lambda =$ The average arrival rate of customers to the system; and
- $W =$ The average time a customer spends in the system including the time in service

An alternate form of Little's Law addresses only the customers in the waiting line, or queue, i.e.,

$$L_q = \lambda W_q$$

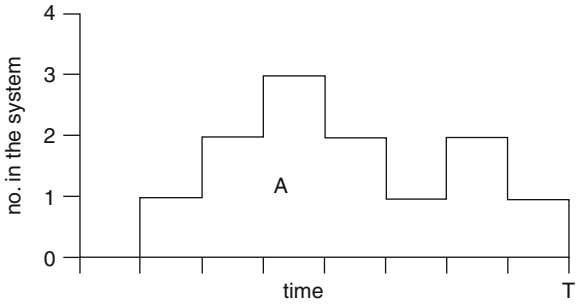
- L_q = The average number of customers in the queueing (excluding customers in service);
- λ = The average arrival rate of customers to the queueing system; and
- W_q = The average time that a customer spends in the queueing (excluding the time in service).

Little's Law, formally proven in Little (1961) and simplified in Stidham (1974), is remarkably general, requiring only that the queueing is ergodic and that no service needs are artificially created or destroyed (i.e., the system is work conserving). The result holds for any arrival process, service-time distribution, and number of servers. It applies for all queueing disciplines, with the customers not necessarily served in order of arrival, and for a specific class of customers that are distinguished from others by priority or some other characteristic. Little's formula holds for every infinite sample path realization of the queueing system, and it is approximately valid in finite intervals, with the accuracy increasing as the interval increases.

In the study of queueing, whether by mathematical analysis, simulation or direct data collection, it is often simpler to find either the average number in system or the average waiting time. Once the simpler one has been found, Little's Law gives the other. For example, in an operating manufacturing system, if average time in the system (lead time) is simpler to estimate from data, Little's Law can be used to estimate the average number of parts in the system (in process inventory).

An outline of a proof of Little's Law is based on depicting a sample path of the number in the system over an interval of time T for a steady-state queueing system with arrival rate λ (Fig. 1). The number of customer-minutes spent in the system equals A, the area under the curve. The average number of customers that arrive in the interval is λT (approximately); thus the average number of minutes in the system per customer is $W = A/(\lambda T)$. The average number of customers in the system $L = A/T$. Manipulating the two equations, taking limits, and accounting for end effects yields Little's Law.

An outline of a proof of Little's Law is based on depicting a sample path of the number in the system over an interval of time T for a steady-state queueing system with arrival rate λ (see Fig. 1). The number of customer-minutes spent in the system equals A, the area under the curve. The average number of customers that arrive in the interval is λT (approximately); thus the average number of minutes in the system per customer is $W = A/(\lambda T)$. The average number of customers in the system $L = A/T$. Manipulating the two equations, taking limits, and accounting for end effects yields Little's Law.



Little's Law, Fig. 1 Sample path realization of the number in the system over time

See

▶ Queueing Theory

References

Little, J. D. C. (1961). A proof for the queueing formula: $L = \lambda W$. *Operations Research, 9*, 383–387.

Little, J. D. C. (2011). Little's Law as viewed on its 50th anniversary. *Operations Research, 59*, 536–529.

Stidham, S., Jr. (1974). A last word on $L = \lambda W$. *Operations Research, 22*, 417–421.

Little's Law in Distributional Form

L. D. Servi
The MITRE Corporation, Bedford, MA, USA

Since Little's Law first appeared in 1961, its simplicity and importance have established it as a basic tool of queueing theory. Little's Law relates the average number of customers in a system, N, with the average

time in the system, T, under very broad conditions. For example, Keilson and Servi (1988) have demonstrated that for many systems, the relationship between the queueing length and the time in the system can be characterized beyond just their average value.

This is possible, however, if a class of customers arrives according to a Poisson process, is served first-in, first-out (FIFO) within the class, and is processed as either

1. An ordinary single-server queueing,
2. A single-server queueing with one or more classes of priority which processes each class according to a preemptive-resume, preemptive-repeat, or nonpreemptive discipline,
3. A vacation model system, where the server takes one or more vacations when the queueing is depleted,
4. A polling system, where a single server moves cyclically between (real or virtual) queueing, either serving the customers at the queueing to exhaustion, employing a Bernoulli schedule, or serving at most K customers at a queueing before moving on, or
5. An $M/G/G/\ldots G/1$ tandem queueing system, where the output of one queueing is the input of another and the service times at successive queueing are i.i.d. service times for successive arrivals.

More precisely, Keilson and Servi (1988) demonstrated that, if for a given class of customers,

(C-1) The arrival process is Poisson with rate λ,

(C-2) All arriving customers enter the system and remain in the system until served,

(C-3) The customers leave the system one at a time in order of arrival, and

(C-4) For any time t, the arrival process after time t, and the time in the system of any customer arriving before time t, are statistically independent,

then the relationship between the probability distribution of the number in the system and the time in the system follows the simple formula

$$\pi_N(u) = \alpha_T(\lambda - \lambda u) \tag{1}$$

where $\pi_N(u) = E[u^N]$ is the probability generating function of N and $\alpha_T(s) = E[e^{sT}]$ is the Laplace transform of the density of T.

Since $d^n\pi_N(u)/du^n = E[N(N-1)\ldots(N-n+1)]$ for $u = 1$ and $d^n\alpha_T(s)/ds^n = (-1)^n E[T^n]$ for $s = 0$, one can relate the moments of queueing lengths to the moments of the time in the system by computing successive derivatives of (1) with respect to u and then evaluating at $u = 1$. For example,

$$\begin{aligned}
E[N] &= E[\lambda T] \\
E\left[N^2\right] &= E\left[(\lambda T)^2\right] + E[\lambda T] \\
E\left[N^3\right] &= E[\lambda T] + 3E\left[(\lambda T)^2\right] + E\left[(\lambda T)^3\right] \\
E\left[N^4\right] &= E[\lambda T] + 7E\left[(\lambda T)^2\right] + 6E\left[(\lambda T)^3\right] + E\left[(\lambda T)^4\right] \\
E\left[N^5\right] &= E[\lambda T] + 15E\left[(\lambda T)^2\right] + 25E\left[(\lambda T)^3\right] + 10E\left[(\lambda T)^4\right]
\end{aligned} \tag{2}$$

The first of these equations is the familiar Little's Law. As is the case of the Pascal Triangle, there is a simple relation between the coefficients. Specifically, one can show that

$$E[N^n] = \sum_{m=1}^{n} S(n,m) E[\lambda T]^m \tag{3}$$

where $S(u, m)$ is a Stirling number of the second kind defined by the recursion $S(n+1,m) = mS(n,m) + S(n,m-1)$ for $n+1 \geq m \geq 1$, $S(n,0) = S(n,n+1) = 0$ for $n \geq 1$ and $S(1,1) = 1$ (Abramowitz and Stegun 1972).

Similarly,

$$E[(\lambda T)^n] = \sum_{m=1}^{n} \bar{S}(n,m) E[N^m]$$

where $\bar{S}(n,m)$ are Stirling numbers of the first kind which satisfy $\bar{S}(n,m-1) = \bar{S}(n,m-1) - n\bar{S}(n,m)$ for $n+1 \geq m \geq 1$, $\bar{S}(n,0) = \bar{S}(n,n+1) = 0$ for $n \geq 1$ and $\bar{S}(1,1) = 1$.

The first two equations of (2) imply the simple but non-intuitive formula

$$\frac{Var[N]}{E[N]} = \frac{Var[\lambda T]}{E[\lambda T]} + 1.$$

The system could refer to the queueing and the pool of customers in service or exclusively to the queueing. In the latter case, additional systems satisfy conditions (C-1)–(C-4). For example, for a multi-server system,

the customers do not leave the system consisting of the queueing and the pool of customers in service on a first-in, first-out basis [and hence violate condition (C-3)]. However, if the system refers exclusively to the queueing, then condition (C-3) is satisfied.

These results have been generalized, for example, to systems with non-Poisson arrivals (Bertsimas and Mourtzinou 1997), to systems operating under heavy traffic (Szczotka 1992), to systems having batch arrivals (Takahashi and Miyazawa 1994), and has been used as the basis to derive explicit formulae for the distribution of the number in the system (or queueing) as well as the time in the system (or queueing) for a number of more classical systems (Keilson and Servi 1990).

See

- ▶ Little's Law
- ▶ Queueing Theory

References

Abramowitz, M. & Stegun, I. A. (1972). *Handbook of mathematical functions with formulas, graphs, and mathematical tables*. National Bureau of Standards, U.S. Government Printing Office. 824–825.

Bertsimas, D., & Mourtzinou, G. (1997). Transient laws of non-stationary queueing systems and their applications. *Queueing Systems, 25*, 115–155.

Keilson, J., & Servi, L. D. (1988). A distributional form of Little's Law. *Operation Research Letters, 7*, 223–227.

Keilson, J., & Servi, L. D. (1990). The distributional form of Little's Law and the Fuhrmann-Cooper decomposition. *Operations Research Letters, 9*, 239–247.

Little, J. (1961). A proof of the theorem $L = \lambda W$. *Operations Research, 8*, 383–387.

Szczotka, W. (1992). A distributional form of Little's Law in heavy traffic. *Annals Probability, 20*, 790–800.

Takahashi, Y., & Miyazawa, M. (1994). Relationship between queueing-length and waiting time distributions in a priority queueing with batch arrivals. *Journal of the Operations Research Society of Japan, 37*, 48–63.

Local Balance Equations

- ▶ Detailed Balance Equations
- ▶ Queueing Theory

Local Improvement Heuristic

A heuristic rule which examines all the solutions that are closely related to a given initial solution and is guaranteed to reach at least a local optimum.

See

- ▶ Heuristic Procedure
- ▶ Local Optimum

Local Maximum

A function $f(x)$ defined over a set of points S is said to have a local maximum at a point $\boldsymbol{x}_0$ in S if $f(\boldsymbol{x}_0) \geq f(\boldsymbol{x})$ for all $\boldsymbol{x}$ in a neighborhood of $\boldsymbol{x}_0$ in S. The point $\boldsymbol{x}_0$ is referred to as a local optimum (maximum).

See

- ▶ Global Maximum (Minimum)
- ▶ Nonlinear Programming
- ▶ Quadratic Programming

Local Minimum

A function $f(\boldsymbol{x})$ defined over a set of points S is said to have a local minimum at a point $\boldsymbol{x}_0$ in S if $f(\boldsymbol{x}_0) \leq f(\boldsymbol{x})$ for all $\boldsymbol{x}$ in a neighborhood of $\boldsymbol{x}_0$ in S. The point $\boldsymbol{x}_0$ is referred to as a local optimum (minimum).

See

- ▶ Global Maximum (Minimum)
- ▶ Nonlinear Programming
- ▶ Quadratic Programming

Local Optimum

▶ Local Maximum
▶ Local Minimum

Local Solution

A best solution in a feasible neighborhood.

Location Analysis

Charles ReVelle[1] and Vladimir Marianov[2]
[1]The Johns Hopkins University, Baltimore, MD, USA
[2]Pontificia Universidad Católica de Chile, Santiago, Chile

Introduction

The term location analysis refers to the development of formulations and algorithms/methodologies to site facilities of diverse kinds in a spatial or geographic environment. The facilities may be sited with relation to demand points, supply points, or with respect to one another. Although facility layout falls within this definition, this topic is not generally considered under the rubric of location analysis. Common descriptive terms for location analysis are deployment, positioning, and siting, although these terms are actually the outcome that follows the execution of a formulation or algorithm.

Location settings may be classified into two broad categories: planar problems and network problems. Planar problems typically assume that the distances between facilities and demand points, supply points or other facilities are given by a metric, a formula that calculates distance between points based on their coordinates in space. Network problems, in contrast, assume that travel can only occur on an underlying network and that distances are the lengths (or cost) of the shortest paths between the particular points on the network. A further distinction between these categories is provided by the assumption in most planar problems of an infinite solution space, that is, that facilities can be sited anywhere on the plane, perhaps subject to exclusion areas or regions. These planar problems are most often non-linear optimization problems and more abstract in their application than network-based problems. In contrast to the infinite solution space assumed by most planar problems, all but a few network problems restrict facilities to sites that have been specified in advance as eligible to house those facilities. The network problems tend to be linear zero-one optimization problems and so pose challenges in their resolution to integers. First, planar problems and approaches to them will be discussed; followed by a discussion about network location formulations and their solution.

Planar Location Problems

The most famous of the planar problems and the first location problem to be posed historically is the minimum Euclidean single facility location problem first stated by Fermat as a mathematical problem: "Given three points in the plane, find a fourth such that the sum of its distances to the three given points is a minimum" (Kuhn 1967). It is often referred to as the Weber problem, after the German economist who first discussed it in economic terms (Weber 1909). The minimum problem considers points dispersed on the plane that send items to or receive finished product from some central factory or facility. The problem seeks the central point that minimizes the sum of weights (quantities) times the distances to all dispersed points. The problem assumes that the Euclidean distances separate the dispersed points and the central point, that the central point can be anywhere on the plane, and that a weight or loading is associated with each of the dispersed points. An iterative solution method that can be shown to converge to an optimal solution was offered in the 1930s, lost to view, and rediscovered in the early 1960s by several independent investigators. In the minisum multiple facility problem (the multi-Weber problem), a number of central facilities are to be sited, each one associated with a cluster or partition of the dispersed points.

An allocation problem arises, i.e., the problem of deciding which facility serves each dispersed point.

The history of the minisum problem is reviewed in Wesolowsky (1993). Only in the early 1990s has this problem yielded to exact methods, followed by heuristics and metaheuristics (Brimberg et al. 2008).

While the Weber problem in its single and multi-facility forms utilizes the Euclidean metric for distances, the minisum rectilinear problem utilizes the Manhattan or rectilinear metric for distances and minimizes the sum of weights times these distances to the central point. The rectilinear distance between two points is the sum of the horizontal and vertical separation of the points. Because the problem can be reduced to the choice among a set of eligible points, the multi-facility rectilinear minisum problem yields either to heuristics or to the linear integer-programming formulation used for the *p*-median problem, a problem that will be discussed under network location models. When the classic metrics are set aside, solution of the minisum problem generally becomes more difficult, except in the case of minimizing the weighted sum of squared distances, in which case the single facility minisum solution is simply the centroid.

A second important objective setting in planar problems is the siting of a single facility under the objective of minimizing the maximum distance that separates any demand/supply point from the central facility. The problem may utilize either of the two classic metrics, Euclidean or rectilinear. No matter the number of dispersed points, the minimax single facility location problem with rectilinear distances yields to either a geometric solution or to a four-constraint linear program. The minimax single facility location problem with Euclidean distances is a nonlinear-programming problem, but can also be solved by a geometric argument. Multi-facility versions of the planar minimax location problems may yield to heuristics resembling those applied to the *p*-median problem. A good general reference dealing in part with planar location problems is the text of Love et al. (1988), Plastria (1995) provides a comprehensive review for the researcher in planar location.

It is worth mentioning that researchers in continuous location, seeking a greater realism in their problems, have sought to project the most likely real distances on a road network between a pair of points given the spatial coordinates of these points. This literature is reviewed in Brimberg and Love (1995).

Network Location Problems

In contrast to the use of formula-based metrics for the siting of facilities on a plane, network location problems always measure distances across the links of the network. Interestingly, the assumption of an infinite solution space can be made in network-based location problems as well. That is, the infinite solution space would consist of all the points on every arc of the network. For some problems, including the *p*-median, the solution space can be reduced without loss of optimality from all the points on all the arcs to a limited number of eligible points when the triangle inequality holds throughout the network. Many network problems simply assume a prespecified set of eligible facility sites based on needed characteristics of such points, such as transportation infrastructure, availability of lots or warehouse space, etc.

Within network location research two distinct foci are found. The first is cost minimizing/profit maximizing siting that is goods-oriented, an activity especially of the manufacturing and distribution industries. The second is people or service-oriented siting, an activity mostly of government at a number of levels from local to national, but also of private companies. The divisions are not perfect, as it will be seen, but are, at the least, useful for discussion purposes. These two settings will be taken up in that order followed by presentations of some variations and adaptations of these classes.

Goods-Oriented Siting

By far, the problem setting considered most extensively in the goods oriented location category is the simple plant location problem (SPLP). The problem assumes that an unknown number of plants are to be sited to manufacture product for distribution to a number of spatially dispersed demand points. The plants have no limit as to the amount manufactured, and each point must be fully supplied with its demand. The objective is the

minimization of the total of manufacturing cost and distribution cost. Manufacturing includes a fixed opening cost and an expansion cost that can be linear or nonlinear. The problem may be stated mathematically as:

$$\text{minimize} \quad z = \sum_{i=1}^{m}\sum_{i=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} f_i y_i$$

subject to :

$$\sum_{i=1}^{m} x_{ij} = 1, \qquad j = 1, \ldots, n,$$

$$y_i - x_{ij} \geq 0, \qquad i = 1, \ldots, m; j = 1, \ldots, n,$$

$$x_{ij}, y_i \in \{0, 1\}, \quad i = 1, \ldots, m; j = 1, \ldots, n.$$

i = the index of eligible plant sites of which there are m;
j = index of demand points of which there are n;
f_i = opening cost for a plant at i;
c_{ij} = cost to deliver j's full demand from i, including the production cost at i;
$y_i \in \{0, 1\}$, it is 1 if a plant opens at i and 0 otherwise; and
$x_{ij} \in \{0, 1\}$, it is 1 if i delivers j's full demand and 0 otherwise.

The above problem formulation is due to Balinski (1965), and is one of several formulations possible for the SPLP. It is presented here because it is the basis for a number of solution methods.

The SPLP has attracted attention since the 1950s when heuristics were first suggested. In the 1960s, Balinski offered his formulation of the problem but dismissed it as unreliable. In addition, several branch and bound algorithms were created to solve the SPLP, but these algorithms proved impractical for large problems. In the mid-1970s, Bilde and Krarup (1977) and Erlenkotter (1978) both proposed dual ascent algorithms for the SPLP; the basic algorithm proposed by these two sets of investigators has proved to be capable of handling relatively large problems. Morris (1978) investigated 500 randomly generated plant location problems and found that if the formulation above were solved as a linear program (without integer requirements on any of the variables) that 96% of the problems so solved presented with all zero-one variables. Morris' experience thus suggested that linear programming alone was a powerful technique for the SPLP formulation that Balinski had abandoned. The problem has since been successfully pursued by Lagrangian relaxation by Galvão (1989) and Korkel (1989), who modified the dual ascent algorithm referred to above to solve remarkably large problems.

While the SPLP has attracted considerable attention, a related form, the capacitated plant location problem (CPLP), languished until the late 1980s. The CPLP sets limits on the amount that could be manufactured at any site, but in all other respects is the same as the SPLP. First attacked by Davis and Ray (1969), the problem later received attention from Pirkul (1987), who provided both references to prior work and a solution algorithm based on Lagrangian relaxation. The CPLP also describes a problem in solid waste management in which waste is generated at population nodes and must be disposed of at sanitary landfills with limited capacities. Landfills are to be sited in this problem statement.

Many other plant location style problems can be stated. A maximum profit version of the SPLP is one such statement. The time dimension has been incorporated in a number of models, Melo et al. (2005). Multiple products can be treated as well. Another line of research focuses on the representation of the cost, since in many cases there are economies of scale or costs that are piecewise linear. Inventory, as well as other logistics costs can be also integrated in these models, see Snyder et al. (2007). Finally, demands, prices, and costs can be viewed as random, leading to stochastic versions of the plant location problem. The SPLP has been not only used for goods-oriented siting, but also for the design of telecommunications networks; in particular, for solving a problem called the Concentrator Location Problem, whose mathematical structure is identical to that of the warehouse or plant location problem. Shen (2007) surveys integrated supply chain design models.

Public Service-Oriented Siting

Nearly all of the plant location problems – excluding the concentrator location problem – emphasize the flow/movement of goods. In contrast, service oriented siting problems focus on the accessibility of people to services or services to people. Flow/movement is part of the equation in some of the models, but simple geographic coverage can suffice in others.

The same two objectives treated under planar problems, minisum and minimax, have also been considered for network location problems of service

siting. The minisum network location problem is known as the *p*-median problem; the minimax network location problem is known as the *p*-center problem. Both were posed together in seminal papers by Hakimi (1964, 1965). He also proved that there is always an optimal solution considering location only at nodes of the network.

The *p*-median problem, which seeks the minimum cost assignment of each population node to one of *p* facilities, resembles the SPLP in all but one modeling aspect. Indeed, so strong is the resemblance of *p*-median to the simple plant location model that the same algorithms may be used for solution of both with minor adaptation, Galvão (1989). The single difference between the two models is easy to explain once a mathematical-programming formulation of the *p*-median is offered. The *p*-median problem seeks to site *p* facilities in such a way that the least total of people times distance traveled to the assigned facility is achieved. Division of this objective by the total of population reveals that minimization of the total population-miles objective also minimizes the average distance that people travel to service. Travel/assignment is always assumed to the closest among the *p* facilities.

The *p*-median problem may be formulated as:

$$\text{minimize} \quad Z = \sum_{i=1}^{n} \sum_{j=1}^{n} a_i d_{ij} x_{ij}$$

subject to :

$$\sum_{j=N_i}^{n} x_{ij} = 1, \quad i = 1, 2, \ldots, n,$$

$$x_{jj} - x_{ij} \geq 0, \quad i, j = 1, 2, \ldots, n; i \neq j,$$

$$\sum_{j=1}^{n} x_{jj} = p$$

$$x_{ij} \in \{0, 1\}, \quad i, j = 1, 2, \ldots, n,$$

a_i = relevant population at demand node i;
d_{ij} = shortest distance from node i to node j;
N = number of nodes;
P = number of facilities; and
$x_{ij} \in \{0, 1\}$; it is 1 if node i assigns to a facility at j and 0 otherwise.

It can be seen from a comparison of the *p*-median formulation and that of the SPLP that the objectives differ only in the presence or absence of fixed opening costs and their opening variables, and that the constraints differ only in the presence or absence of a constraint on the number of facilities. In all other respects, the formulations look virtually identical. If the constraint on the number of facilities in the *p*-median formulation is brought to the objective with a multiplier λ, the objective becomes

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_i d_{ij} x_{ij} + \sum_{j=1}^{n} \lambda x_{jj}.$$

The subscripts reflect flow between central facilities and demand points. The *p*-median is now fully equivalent to an SPLP with equal opening costs, thus making all the techniques for solution of the SPLP available for solution of the *p*-median. Ranging the multiplier λ in the *p*-median is equivalent to trading off people miles against the number of facilities by use of the weighting method of multi-objective programming. Among the methods available for the SPLP that can be used for the *p*-median are relaxed linear programming (ReVelle and Swain 1970), the dual ascent methodology (Bilde and Krarup 1977; Erlenkotter 1978) and Lagrangian relaxation (Galvão 1989). A number of other researchers have used heuristics for the *p*-median problem; a listing of many of the early methods for the *p*-median problem appeared in ReVelle et al. (1977). Newer and more effective heuristic and metaheuristic methods are reviewed in Mladenovic et al. (2007) and Reese (2006). As the SPLP, the *p*-median also has a capacitated version in which each facility can serve up to a certain number of people.

While the *p*-median problem attracted considerable attention, researchers found its focus on the average condition of population accessibility to be limiting. Concern for those worst off relative to their distance to the nearest facility, that is, for the maximum distance or time separating population centers from service, gave rise to another concept, that of coverage. A population node is considered to be covered, i.e., adequately served, if it has a facility sited within some maximum distance or time; that is, sited within a time standard. Coverage can either be required for all demand points within the standard, or maximization of demand covered can be sought, giving rise to a host of new problems, the earliest of which is the location set covering problem (LSCP).

The LSCP seeks to position the least number of facilities so that every point of demand has at least

one facility sited within the time or distance standard. The problem can be stated as a linear zero-one programming problem as follows:

$$\text{minimize} \quad z = \sum_{j \in J} x_j$$

$$\text{subject to :} \sum_{j \in N_i} x_j \geq 1 \quad \forall i \in I,$$

$$x_j \in \{0, 1\} \quad \forall j,$$

i, I = index and set of demands;
j, J = index and set of eligible sites for facilities;
$x_j \in \{1, 0\}$, 1 if a facility placed at j and 0 otherwise;
d_{ji} = the shortest distance (or time) from site j to demand point i;
S = the maximum distance (or time) that a demand point can be from its nearest facility; and
$N_i = \{j | d_{ji} \leq S\}$ = the set of facility sites eligible to serve demand point i, by virtue of being within S of i.

While general set covering problems may require integer-programming algorithms to solve them, the LSCP appears to possess special properties. In particular, solution of the linear-programming formulation on data from a geographic problem without any zero-one requirements produces all zero-one answers with remarkable regularity (over 95% of the time). If a set of eligible facility sites is specified in advance, the LSCP can be used to derive solutions to the p-center problem as well. The p-center problem seeks to position p facilities in such a way that the maximum distance that separates any population node from its nearest facility is as small as possible. Solutions to this problem can be found by solving a sequence of LSCP problems, with decreasing distance standards. As the distance decreases, the number of facilities required to cover all demands increases. The minimum distance standard that makes total coverage feasible with p facilities is the solution of the p-center problem (Minieka 1970). If, however, any point on any link of the network is eligible to house a facility (the infinite solution space case), the solution of the p-center problem remains open and challenging.

The LSCP, however, has several shortcomings as a meaningful problem statement. First, population is absent from the problem statement; proximity and population are not linked even though they should be. Second, all population nodes require coverage within the standard, a requirement that could and often proves very costly in terms of the number of facilities/servers required.

Recognizing these shortcomings of the LSCP, several researchers have created new models for siting that utilized the coverage concept not as a requirement but as a goal. The most widely known of these models is referred to as the maximal covering location problem (MCLP) or the partial covering problem, depending on the specific formulation. The MCLP seeks the positions for p facilities among a prespecified set of eligible points that maximize the population that has a facility sited within a distance or time standard S, that is, that maximizes the population covered. The MCLP can be stated as:

$$\text{maximize} \quad z = \sum_{i \in I} a_i y_i$$

$$\text{subject to :} \quad y_i \leq \sum_{j \in N_i} x_j \, \forall i \in I,$$

$$\sum_{j \in J} x_j = p,$$

$$x_j, y_i \in \{0, 1\}, \forall i, j,$$

where additional notation is
a_i = the population at demand node i;
$y_i \in \{1, 0\}$, it is 1 if demand i is covered by a facility within N_i and 0 otherwise; and
p = the number of facilities that can be sited.

Basically, while the LSCP is attempting to find the least resources to cover all demand nodes within the distance goal, the MCLP is attempting to distribute lesser and limited resources to achieve as much population coverage as possible (Church and ReVelle 1974).

Related Research and Extensions

The basic models described above have caught the interest of a number of researchers. The literature on the subject keeps growing.

Drezner (1987) addressed the unreliable p-median in which facilities can become inactive. Marianov and

Serra (1998) proposed models that include the effect of queuing at the facilities, while Marianov (2003) maximized the amount of people willing to get service from a facility when there is demand elasticity to travel distance and queueing. The user point of view has been embedded in the p-median by Drezner and Drezner (2007), who investigated the effect on location of considering customers' behavior, represented through gravity models.

Uncertainty has also been considered in covering models. In probabilistic covering models, the presence or availability of a vehicle or server within a time standard is not guaranteed. The probabilistic models suggest a chance constraint on vehicle availability, that is, a requirement that a vehicle be available within the time standard with a specified level of reliability, see ReVelle and Marianov (1991). The chance constraint may be a strict requirement or may be treated as a goal for each population demand node. Many of the probabilistic, as well as redundant/backup coverage models, and multiple vehicle type models were reviewed by Marianov and ReVelle (1995). A review of the applications of probabilistic coverage models to emergency systems is provided by Goldberg (2004).

A number of other lines of research within the network location setting have been pursued. Among these are hierarchical location models, models in which a hierarchy of interacting/interrelated facility types are sited. One example is the health care hierarchy in developing nations, that consists of hospitals, clinics, and remote doctors. Another is a banking system consisting of central banks, branch banks, and teller machines. Morphological relations in hierarchical systems is reviewed by Narula (1986), with a brief treatment of the topic given in Daskin (1995). Serra and ReVelle (1994) provide algorithms for the median version of these hierarchical problems where coherence of assignments is enforced. Church and Eaton (1987) present an interesting set of hierarchical models with referral between levels.

The concept of coverage has been challenged, since in some situations it does not seem reasonable to consider a demand as covered if it is within, say, 500 m from a facility, but not covered if it is at 500.1 m. Models using a gradual coverage have been reviewed by Eiselt and Marianov (2009a). In these models, the coverage function, originally a step function, can take different shapes, representing quality of coverage as a function of the distance.

Another significant line of siting research is embodied in the competitive location models in which facilities are sited in a competitive market environment with goals of capturing market share from other retailers or manufacturers, or maximizing profit in the presence of competitors. Two problems are usually solved: the follower's problem, which is to locate facilities in such a way that the market capture from existing competitors is maximized; and the leader's problem, which is to locate first in a virgin market, anticipating possible followers that will try to cannibalize the leader's market share. A review of competitive location models in continuous and discrete space is provided by Dasci (2011).

Another line of location research involves the siting of noxious facilities. Such facilities may be undesirable in of themselves and should be distant from population centers or may be required to be distant from one another. However, they usually cannot be too far, since operation costs can be prohibitive, as in garbage processing plants or jails. Several approaches have been proposed for these facilities: maximizing their distance to population; maximizing the minimum facility-population distance; compensating the population that is affected by such a facility; and expropriation. A review of obnoxious facility location problems can be found in Melanchrinoudis (2011). Another line of research addresses both location of obnoxious facilities and routing of hazardous waste (Nagy and Salhi 2007).

A problem of increasing interest is the location of hubs. As airlines and courier companies focus on logistic improvements, the location of these traffic concentration points becomes more relevant. This line of research was started by (O'Kelly 1986) and has grown towards several fronts. Hub problems can be classified into the same categories as the original location problems: hub-median, hub-location, hub-covering and hub-center problems. They can be solved on the plane (O'Kelly 1986), or on networks. Campbell et al. (2002) provide a taxonomy of hub problems. Competition and queuing effects have also been considered when locating hubs (Marianov and Serra 2003; Eiselt and Marianov 2009b).

Finally, the tools developed for location in a geographical setting can be also used in very

different spaces: to locate employees and tasks in a skill space, finding the best measurement points in the eye for glaucoma detection, and locating candidates and voters in an issue space.

Concluding Remarks

The wide variety of important applications and modeling challenges are reported in many OR/MS journals, including *Computers & Operations Research* (including *Location Science*); *European Journal of Operational Research*, *Journal of the Operational Research Society*; *IIE Transactions* and *Papers in Regional Science*. In addition, the proceedings of the triennial International Symposium on Locational Decisions (ISOLDe) have appeared in separate volumes of *Annals of Operations Research*, beginning with 1984 Boston/Martha's Vineyard conference.

See

- ▶ Facility Location
- ▶ Integer and Combinatorial Optimization
- ▶ Network
- ▶ Shortest-Route Problem
- ▶ Stochastic Programming

References

Balinski, M. (1965). Integer programming: Methods, uses and computations. *Management Science, 12*, 253–313.

Bilde, O., & Krarup, J. (1977). Sharp lower bounds and efficient algorithms for the simple plant location problem. *Annals Discrete Mathematics, 1*, 79–97.

Brimberg, J., & Love, R. (1995). Estimating distances. In Z. Drezner (Ed.), *Facility location: A survey of applications and methods* (pp. 9–31). New York: Springer.

Brimberg, J., Hansen, P., Mladenovic, N., & Salhi, S. (2008). A survey of solution methods for the continuous location-allocation problem. *International Journal of Operations Research, 5*, 1–12.

Campbell, J. F., Ernst, A. T., & Krishnamoorthy, M. (2002). Hub location problems. In Z. Drezner & H. W. Hamacher (Eds.), *Facility locations applications and theory* (pp. 373–407). New York: Springer.

Church, R., & Eaton, D. (1987). *"Hierarchical location analysis using covering objectives", in spatial analysis and location models*. New York: Van Nostrand-Rheinhold.

Church, R., & ReVelle, C. (1974). The maximal covering location problem. *Papers Regional Science Association, 32*, 101–118.

Dasci, A. (2011). Conditional location problems on networks and the plane. In H. A. Eiselt & V. Marianov (Eds.), *Foundations of location analysis* (pp. 179–206). New York: Springer.

Daskin, M. (1995). *Network and discrete location*. New York: Wiley.

Davis, P., & Ray, T. (1969). A branch-and-bound algorithm for the capacitated facilities location problem. *Naval Research Logistics Quarterly, 16*, 331–344.

Drezner, Z. (1987). Heuristic solution methods for two location problems with unreliable facilities. *Journal of the Operational Research Society, 38*, 509–514.

Drezner, Z. (Ed.). (1995). *Facility location: A survey of applications and methods*. New York: Springer.

Drezner, T., & Drezner, Z. (2007). The gravity p-median model. *European Journal of Operational Research, 179*, 1239–1251.

Eiselt, H. A., & Marianov, V. (2009a). Gradual location set covering with service quality. *Socio-Economic Planning Sciences, 43*(2), 121–130.

Eiselt, H. A., & Marianov, V. (2009b). A conditional p-hub location problem with attraction functions. *Computers and Operations Research, 36*, 3128–3135.

Erlenkotter, D. (1978). A dual-based procedure for uncapacitated facility location. *Operations Research, 26*, 992–1009.

Galvão, R. (1989). A method for solving optimality uncapacitated location problems. *Annals of Operations Research, 18*, 225–244.

Goldberg, J. B. (2004). Operations research models for the deployment of emergency services vehicles. *EMS Management Journal, 1*(1), 20–39.

Hakimi, S. L. (1964). Optimal location of switching centers and the absolute centers and medians of a graph. *Operations Research, 12*, 450–459.

Hakimi, S. L. (1965). Optimum distribution of switching centers in a communication network and some related graph theoretic problems. *Operations Research, 13*, 462–475.

Korkel, M. (1989). On the exact solution of large-scale simple plant location problems. *European Journal of Operational Research, 39*, 157–173.

Kuhn, H. (1967). On a pair of dual nonlinear programs. In J. Abadie (Ed.), *Nonlinear programming* (pp. 39–54). Amsterdam: North-Holland.

Love, R., Morris, J., & Wesolowsky, G. (1988). *Facilities location: Models and methods*. New York: North Holland.

Marianov, V. (2003). Location of multiple-server congestible facilities for maximizing expected demand, when services are non-essential. *Annals of Operations Research, 123*, 125–141.

Marianov, V., & ReVelle, C. (1995). "Siting emergency services," chapter 10 in facility location. In Z. Drezner (Ed.), *A survey of applications and methods*. New York: Springer.
Marianov, V., & Serra, D. (1998). Probabilistic maximal covering location-allocation models for congested systems. *Journal of Regional Science, 13*, 401–424.
Marianov, V., & Serra, D. (2003). Location models for airline hubs behaving as M/D/c queues. *Computers and Operations Research, 30*, 983–1003.
Melanchrinoudis, E. (2011). The location of undesirable facilities. In H. A. Eiselt & V. Marianov (Eds.), *Foundations of location analysis* (pp. 207–240). New York: Springer.
Melo, M. T., Nickel, S., & da Gama, F. (2005). Dynamic multi-commodity capacitated facility location: A mathematical modeling framework for strategic supply chain planning. *Computers and Operations Research, 33*, 181–208.
Minieka, E. (1970). The M-centre problem. *SIAM Review, 12*, 138–141.
Mladenovic, N., Brimberg, J., Hansen, P., & Moreno-Perez, J. (2007). The p-median problem: A survey of metaheuristic approaches. *European Journal of Operational Research, 179*, 927–939.
Morris, J. (1978). On the extent to which certain fixed charge depot location problems can be solved by LP. *Journal of the Operational Research Society, 29*, 71–76.
Nagy, G., & Salhi, S. (2007). Location-routing: Issues, models and methods. *European Journal of Operational Research, 177*, 649–672.
Narula, S. (1986). Minisum hierarchical location-allocation problems on a network: A survey. *Annals of Operations Research, 6*, 257–272.
Nickel, S., & Puerto, J. (2005). *Location theory: A unified approach*. New York: Springer.
O'Kelly, M. E. (1986). The location of interacting hub facilities. *Transportation Science, 20*(2), 92–106.
Pirkul, H. (1987). Efficient algorithms for the capacitated concentrates location problem. *Computers and Operations Research, 14*, 197–208.
Plastria, F. (1995). Continuous location problems. In Z. Drezner (Ed.), *Facility location: A survey of applications and methods*. New York: Springer. Chapter 11.
Reese, J. (2006). Solution methods for the p-median problem: An annotated bibliography. *Networks, 48*, 125–142.
ReVelle, C., Bigman, D., Schilling, D., Cohon, J., & Church, R. (1977). Facility location: A review of context-free and EMS models. *Health Services Research, Summer, 12*, 129–146.
ReVelle, C., & Marianov, V. (1991). A probabilistic FLEET model with individual reliability requirements. *European Journal of Operational Research, 53*, 93–105.
ReVelle, C., & Swain, R. (1970). Central facilities location. *Geographical Analysis, 2*, 30–42.
Serra, D., & ReVelle, C. (1994). Location and districting of hierarchical facilities—II heuristic solution methods. *Location Science, 2*, 63–82.
Shen, Z. J. M. (2007). Integrated supply chain design models: A survey and future research directions. *Journal of Industrial and Management Optimization, 3*, 1–27.
Snyder, L. V., Daskin, M. S., & Teo, C. P. (2007). The stochastic location model with risk pooling. *European Journal of Operational Research, 179*, 1221–1238.
Weber, A. (1909). *Uber den standort der industrien, Tübingen [Translation: (1929). Theory of the location of industries]*. Chicago: University of Chicago Press.
Wesolowsky, G. (1993). The Weber problem: History and perspectives. *Location Science, 1*, 5–23.

Logic Programming

Logic programming deals with the use of symbolic logic for problem representation and inferential reasoning. A popular logic programming language is Prolog (PROgrammation en LOGique), developed in the early 1970s by the French computer scientists, Alain Colmerauer and Philippe Roussel. Prolog has been used to develop a man-machine communication system in natural language.

See

► Artificial Intelligence

References

Bergin, T. J., Jr., & Gibson, R. G., Jr. (Eds.). (1996). *History of Programming Languages—II*. New York: ACM Press.
Cohen, J. (1988). A view of the origins and development of Prolog. *Communications of the ACM, 31*(1), 26–36.

Logical Variables

In a linear-programming problem, the set of variables that transform a set of inequalities to a set of equations are called logical variables.

See

► Linear Inequality
► Slack Variable
► Structural Variables
► Surplus Variable

Logistics and Supply Chain Management

Marius M. Solomon
Northeastern University, Boston, MA, USA

Introduction

For quite some time, logistics has accounted for a significant percentage of the U.S. gross domestic product (GDP). The Council of Supply Chain Management Professionals estimated that in 2008 the country's logistics costs were about $1.3 trillion, or 9.4% of the $13.8 trillion GDP. Year-to-year carrying costs decreased by 13.2% due to smaller inventories and lower interest rates while transportation costs rose by 2% as a result of higher fuel prices. These figures and a number of other key economic developments highlight logistics and supply chains as areas where large productivity improvements have and continue to be attained. Given the intrinsic complexity of logistics problems in today's global supply chains, such improvements could not have been achieved without the use of analytical tools, including operations research/management science (OR/MS) methodologies.

The mathematical difficulty of strategic, tactical, and operational logistics decisions and the magnitude of the potential cost savings to be achieved by utilizing OR/MS models and algorithms have attracted researchers since the early days of the field. Witness to this are the pioneering efforts of researchers in 1950s, 1960s, and 1970s. Most of the methods developed made extensive use of network models and algorithms coupled with different types of inventory techniques.

Over the last twenty five years, fueled by major developments in modeling and algorithmic methodology, constant breakthroughs in computer technology, and web-based applications, operations researchers have found logistics to be a very fertile design and implementation area. They addressed an ever increasing variety of problems with escalating complexity and size. The body of supply chain applications of OR/MS techniques also expanded at a progressively swifter pace. In what follows, the focus will be on some of the more important areas in logistics and supply chain management and, where possible, on OR/MS applications in large-scale logistics systems.

Networking and Routing

Network design and freight routing have been addressed by Braklow et al. (1992) in the context of less-than-truckload (LTL) transportation. The authors formulate the problem as a nonlinear, multicommodity network design problem. Its solution is based on a hierarchical decomposition of the overall problems into a series of optimization subproblems. The network design problem is solved using interactive optimization, where the user guides the search performed by a local improvement heuristic which adds (drops) links to (from) the load planning network. The subproblems involve the routing of the LTL shipments, of truckload shipments and of empty trailers. The former two problems are solved using shortest path algorithms, while the latter problem involves the solution of a classical linear transhipment problem. They must be reoptimized every time a change is made in the load planning network. This is performed sufficiently fast to make interactive optimization possible. The model has been used as a tactical decision tool for load planning by one of the largest LTL motor carriers. It has also been used at the strategic level to determine the location and size of new terminals.

The research of Simão et al. (2010) is illustrative of the evolution of the OR/MS methodology which had to match the increasing complexity of real-world problems due to their size and dynamism. The authors address the problem faced by a major transportation company that wanted the ability to significantly improve how it managed the dynamics of its fleet of over 6,000 long haul drivers. The issues under consideration were how to handle hiring, changes in work rules, and examine scenarios permitting the drivers to spend more time at home. Simão et al. used approximate dynamic programming (ADP) to solve this problem. ADP is a simulation-based algorithm that optimizes complex stochastic problems through iterative learning. This approach was capable to deal with both complex dynamics and multiple forms of uncertainty regarding drivers and loads and to anticipate the future impact of decisions. The model allowed the company to avoid costs and achieve savings in the millions of dollars and, at the same time, substantially improve its customer service.

While logistics encompasses a broad set of activities, two key elements are transportation

and storage. Generally, very intricate trade-offs occur between these two areas. The first focus will be on transportation issues and then address inventory matters. Transportation is in fact the most costly component of many logistics systems and supply chains. A very important segment of transportation management is the routing and scheduling of vehicles. This facet is of significant importance across land, air, and water transportation. Similar problems are also encountered in a variety of manufacturing, warehousing and service sector environments.

This area has been reviewed in several insightful surveys, including that written by Laporte (2009). The author highlights the major developments in the OR/MS methodology for the vehicle routing problem (VRP) over the last fifty years. He reviews successful exact algorithms and heuristics introduced in the literature ranging from extremely sophisticated optimal decomposition algorithms to powerful metaheuristics. His work is complemented by the books edited by Toth and Vigo (2002) and Golden et al. (2008) who put together articles spanning a multitude of VRP variants. All these sources also provide a wealth of references to research conducted over the years in the ever increasing universe of VRP problems.

While Laporte highlights the outstanding progress made by optimal algorithms, he also notes that such methods have their limitations with respect to increasing larger problems. Certainly, they can be transformed into optimization-based heuristics which can solve larger problems. However, when it comes to huge instances, heuristics are still the answer. Laporte also observed that over time the research community has designed metaheuristics that have become more and more over-engineered at the expense of computation time. He suggests researchers should consider producing simpler and more flexible algorithms capable of faster handling of a broader variety of constraints, even if they cause a slight decrease algorithmic effectiveness.

The application of OR/MS methods in this area has lead to significant achievements in practice. Kant et al. (2008) report on a very successful implementation undertaken by Coca-Cola Enterprises (CCE), the world's largest bottler and distributor of Coca-Cola products. The CCE fleet in the U.S. is only surpassed in size by that of the U.S. Postal Service. The software developed is very flexible and handles a variety of practical constraints in determining the truck routes from each distribution center to the retail outlets. Hundreds of dispatchers use this software daily to plan the routes for tens of thousands of trucks. The deployment of the software has resulted in annual cost savings of tens of millions of dollars. In addition, CCE has experienced fewer missed deliveries and gained the ability to deal with tighter time windows, thereby substantially enhancing its customer service. Given the success of the software, Coca-Cola decided to roll it out in other parts of its business.

A variety of routing settings also involve the temporal aspect in the form of customer imposed time windows. A unified framework for all time constrained vehicle routing and crew scheduling problems was developed by Desaulniers et al. (1998). This paper presents a more general model than previously considered which integrates all the different time constrained vehicle routing and crew scheduling problem types examined up that point in the literature. The model extends well-known generic formulations to allow the modeling of all real-world circumstances encountered to date in this environment. This enables the reader to understand the common structure of these problems. It also allows one to perceive the relations between the various problems, the different forms of the model used previously in the literature, and assorted applications across a unified formulation. This also permits the reader to note the diversity of specialized algorithms that have been designed to solve them, and to comprehend the difficulties inherent in certain modeling aspects.

The common structure of these problems is a multi-commodity network flow model with additional resource constraints. Time is one example of a resource. Resource variables help manage complex nonlinear cost functions and difficult local constraints (e.g., time windows, vehicle capacity, and union rules). To solve the nonlinear multi-commodity problems in this class, the paper presents a branch-and-bound framework. It shows that a variety of strategies and algorithms can be utilized for the computation of lower bounds and for devising branching schemes. The lower bounds are derived by using a decomposition approach. In their paper, Desaulniers et al. focus on an extension of the Dantzig-Wolfe decomposition principle and establish that this is valid even for nonlinear objective functions and constraints.

They also illustrate that it embeds the column generation-based methods using set partitioning formulations previously suggested in the literature as special cases. The branching module used to obtain integer solutions compatible with column generation is more general, but yet simpler than other prior strategies. Branching decisions and cuts appear either in the master problem or in the subproblem structures. Finally, the authors examine the constrained shortest path problems that appear at the subproblem level of the decomposition. The paper displays the variety of specialized dynamic programming algorithms that have been developed to solve these and more general single commodity problems and the aspects which have not yet received attention.

Optimal algorithms stemming from the above framework have emerged as the most preferred solution methodologies. These branch, price, and cut algorithms have been widely applied not only in a variety of routing and scheduling transportation contexts, but also in crew scheduling, network design, production, and telecommunications, as well as other areas. These algorithms have become even more powerful due to different classes of strong cutting planes that have been proposed to tighten the lower bounds. Significant improvements in the quality of the lower bounds computed in the search tree have also resulted from utilizing the elementary shortest path problem with resource constraints at the subproblem level.

Crew Scheduling

Two notable application areas of the above framework are the urban transit crew scheduling problem and the airline crew scheduling problem. Blais et al. (1990) describe a software package to handle the former problem. It consists of several modules. The first uses standard network flow methodology to solve the bus scheduling problem. Next, crew scheduling is handled in two steps. In the first, several approximations are used to permit the fast derivation of a linear- programming solution. Using this solution, specific driver assignments are then obtained in step two by means of solving a quadratic-integer program heuristically and using an optimal matching algorithm. Finally, a shortest path algorithm utilizing the marginal costs from the matching problem is used to improve the solution. The software has been successfully implemented in a number of cities worldwide.

With respect to exact algorithms, very large multiple-depot vehicle scheduling problems can be solved to optimality in reasonable times. The same holds true for practical crew scheduling problems encountered in urban mass transit and in air transportation. However, the joint consideration of these two problems proved to be much more challenging. Haase et al. (2001) address this simultaneous vehicle and crew scheduling problem in urban mass transit systems. They propose an optimization algorithm based on the above Dantzig-Wolfe column-generation framework for the problem variant involving a single depot case and a homogeneous vehicle fleet. The authors take a crew-first, vehicle-second approach where decision variables are defined only for the scheduling of drivers. The bus routes are handled within constraints. These constraints ensure that optimal bus itineraries can be obtained in polynomial time once the crews have been scheduled. The authors provide computational results that indicate that this technique was capable to optimally solve larger problems than previously reported in the literature. An easily achieved optimization-based heuristic version of the method is was able to solve even larger instances.

The evolution in airline crew scheduling from the manual methods of the early 1970s to the powerful OR/MS based software now in use mirrors the developments that have occurred in many other logistics areas. In addition, research in crew scheduling is part of the stream of research spearheading the development of optimization methods capable of handling practical size problems. This new generation of optimal algorithms discussed above blends the effectiveness of advanced optimization methods, designed to take advantage of special problem structures, with the efficiency of sophisticated computer science techniques, and the computing power of workstations.

Air transport carriers use a five-phase tactical planning and scheduling process. The schedule planning phase first determines all flight segments, or legs, to be flown during a given period, according to the forecasted demand, the time slots that the company owns at different airports, and the competition. The next phase is fleeting, where each equipment type or

fleet is assigned to individual legs. The fleeting solution provides a decomposition for the problems to be considered in the next three phases. For each fleet, the flight legs with their corresponding scheduled departure and arrival times become inputs to the aircraft routing phase. At this stage, for each type of aircraft, routes are build that must encompass all legs to be flown and satisfy maintenance requirements. The fourth phase builds valid crew pairings, also known as crew rotations, to minimize crew cost. A pairing is a detailed schedule of activities, such as flight legs, deadhead legs (crew members fly as passengers), briefings and debriefings, breaks and nighttime rests that start and end at the same crew base. In the fifth phase, employees are assigned to monthly blocks where each block describes the activities of a crew member during the month. When this process accounts for employee preferences it is called rostering. When blocks are built without regard to crew members' desires, the process is called bidding, in which case, crew members choose blocks according to seniority.

Butchers et al. (2001) provide a historical account and discuss the OR/MS techniques developed for crew scheduling and rostering at a major airline over a fifteen year period. It highlights the fact that the use of such methodologies created major savings for the company, while at the same time providing rosters that benefited the crew members. The account is also illustrative of the advantages to be derived from close collaborations between industry and academia. Nevertheless, the airline planning process phases considered had to be treated sequentially due to the size of the problems involved. The fact that in this planning process the output of an earlier phase provides the input to the next later phase generally leads to suboptimal policies.

Researchers have started to solve selected subsets of planning problems such as fleeting and aircraft routing and aircraft routing and crew pairing simultaneously. Representative of this line of work is that of Sandhu and Klabjan (2007) that addresses the fleeting, aircraft routing, and crew pairing phases in an integrated fashion. The maintenance requirements that must be satisfied in the aircraft routing phase are, however, not considered. The authors propose two optimal algorithms, one using a Benders decomposition approach and the other involving a combination of Lagrangian relaxation and column generation. Based on computational experiments conducted using data from a major carrier, they conclude that if improvements are sought in a short amount of time, the former method should be used. However, if sufficient computing time is available, the usual case in this planning environment, then the latter technique should be utilized. In addition, the authors found the Lagrangian relaxation/column generation approach more robust and practical.

Real-Time Logistics

While the size of problems solved by optimization algorithms increases constantly, heuristics remain a viable tool for very large-scale and/or very complex problems. Dispatching, an intricate activity given the need for a solution in real-time to large-scale problems, lends itself naturally to heuristic solutions. The use of fast route construction/route improvement heuristics to deal with the practical complexities of the problem typifies the kind of research conducted in the 1980s. The highly dynamic character of dispatching is also apparent in truckload transportation. In this environment characterized by high demand uncertainty, a motor carrier must continuously manage the assignment of drivers to loads across the country. Stochastic network optimization models exemplify the type of methodology developed to solve this dynamic vehicle allocation problem. Powell et al. (1995) provide an extensive survey of this problem area.

When shipments could not be forecasted with accuracy, Moore et al. (1991) report having built mixed-integer programming (MIP) and simulation models. The use of these techniques for operational purposes has stemmed from the successful solution of a strategic decision through similar methods. This decision involved the significant reduction in the number of carriers used and the creation of partnerships with them. To solve the carrier selection problem for a global, integrated aluminum company, the authors developed an MIP and further analyzed its results using simulation. This problem represented an important part of a redesign effort aimed at centralizing previously decentralized transportation and purchasing decisions. In particular, by creating a central dispatch center and supporting decisions with OR/MS methodologies, the company improved on time delivery and reduced annual freight costs by

millions of dollars. Overall, this implementation was a reflection of the lean manufacturing philosophy extended to logistics. Furthermore, as logistics has evolved into an information technology centric environment, partnerships with carriers now involve electronic data interchange and web based information sharing.

Supply chains have become a competitive weapon in the global economy. The remarkable advances in telecommunications and information technology have enabled companies to focus on velocity and timeliness throughout the supply chain. To achieve these competitive advantages, they must be able to make effective use of the vast amount of real-time information now available to them. The Dynamic Vehicle Routing Problem (DVRP) is a prime example of a distribution context where intelligent use of real-time information can differentiate one company from another by means of superior on-time service. The DVRP is the dynamic counterpart of the VRP. In the latter problem, the objective is generally to minimize the travel cost for several vehicles that must visit and service a number of customers. Constraints specifying capacity restrictions, time windows within which to start service at customers, and additional requirements on the drivers and vehicles restrict the optimization space. In the VRP all routing and demand information is known with certainty prior to the day of operations, so routes can be planned ahead. In contrast, in the DVRP part or all of the necessary information becomes available only during the day of operation. In other words, not all information relevant to the planning of the routes is known by the planner when the routing process begins and information can change after the initial routes have been constructed.

The practical significance of the DVRP is highlighted by the variety of environments it can model. An important application is the pickup and delivery of overnight mail. Other scenarios include the distribution of heating oil or liquid gas to private households, residential utility repair services, such as cable and telephone, and appliance repair. Additional settings are the transportation of the elderly and physically disabled, taxi cab services, and emergency services, such as police, fire, and ambulance dispatching.

Gendreau and Potvin (2004) have edited a special issue of *Transportation Science* dealing with many issues in real-time fleet management. These were created by the consideration of transportation and fleet management activities as an integral part of the supply chain, their coordination with other aspects of the supply chain, and the explosive growth of web-based logistics services. The paper by Larsen et al. (2004) is illustrative of this type of research. The authors examine the traveling salesman problem with time windows for various degrees of dynamism. The objective is to minimize lateness and examine the impact of this criterion choice on the distance traveled. The focus on lateness is motivated by the problem faced by overnight mail service providers. A real-time solution method is proposed that requires the vehicle, when idle, to wait at the current customer location until it can service another customer without being early. In addition, the authors develop several enhanced versions of this method that may reposition the vehicle at a location different from that of the current customer based on a priori information on future requests. The results obtained on both randomly generated data and on a real-world case study indicate that all policies proved capable of significantly reducing lateness. The results also show that this can be accomplished with only small distance increases.

Another important setting for the application of OR/MS methodologies to support real-time decisions is in the airline industry. Airlines must build aircraft routes and crew rotations to provide scheduled service while maximizing profits. This objective must be achieved in an environment that is difficult to predict. Hence, planning decisions– made in advance– may have to be altered by real-time decisions when perturbations occur in order to minimize customer inconvenience and costs to the airline. Changes made on the day of operations result from bad weather conditions, headwinds on route, technical difficulties with aircraft, crew and passenger delays, and peak-hour congestion at airports. This challenging problem is very important in practice since perturbations are costly in terms of rescheduling issues and especially in terms of loss of traveler goodwill. This is because they can lead to delaying or canceling flights, swapping aircraft among flights or using spare aircraft (if any exist), which in turn affect future deployment of aircraft and crews. Dispatchers usually adjust the planed schedules as soon as a perturbation occurs. They have little time to analyze cost-effective scheduling alternatives. Therefore, it is important to find a good balance between the optimality of a proposed solution and the speed with which it is obtained.

Historically, the day of operations solutions have relied mainly on management information systems and graphical user interfaces, and on simple heuristics to support the decision process. Exact algorithms also have been deployed in practice to provide optimal or near-optimal solutions. Yu et al. (2003) present an optimization based decision support system developed for a large air carrier that provides crew-recovery solutions. The software proved capable of handling major disruptions and in turn it allowed the airline to recover quickly and derive benefits in the millions of dollars.

Inventory in the Supply Chain

The fundamental and often complex trade-offs between transportation and inventory costs are a central issue in supply chain management. Blumenfeld et al. (1987) present an ingenious analysis of the production network of a manufacturer of vehicle components. Their bottom-up approach begins with the analysis of the trade-offs on a single link. These are obtained using a standard economic order quantity (EOQ) model. Using several realistic approximations, the authors are then able to extend their analysis to much more complex networks. In particular, one approximation allows the decomposition of a large network into a number of small independent subnetworks, where shipment sizes can be computed using the single link model. This work involving simple, easy to understand models, supplemented by insightful graphical information, is representative of a line of research complementary to combinatorial optimization.

In light of intense global competitive pressures, many companies have tried to decrease their inventory investment while maintaining or improving customer service in their vital business processes. Yet, the implementation of lean manufacturing has lead to significant increases in product variety. In turn, this has augmented the complexity of the after-sales service logistics networks. Cohen et al. (1990) describe the design of a spare parts inventory control system capable of supporting multiple service levels. The building block of their approach is a periodic review, stochastic model for the one-part, one-location case. This model is then extended to a multi-product, one-location case, called the service allocation problem. This is solved using a greedy heuristic. A decomposition approach is utilized for the overall multi-product, multi-echelon problem. It involves a bottom-up procedure which begins by solving the service allocation problems at the lowest echelon. The solutions are then used to deal with the next higher echelon. The algorithm proceeds in this fashion, level-by-level up to the highest echelon. The model has been implemented by a global computer manufacturer. It has found applicability both as a strategic network redesign tool and as a weekly operational device.

Inventory investment becomes progressively more substantive with increases in the size of companies holding it. While enterprise resource planning software has provided much needed inventory visibility in the supply chain, these systems do not optimize inventory levels. OR/MS methods do, but as they have become increasingly sophisticated over time, the scale and complexity of supply chains has also augmented. The paper by Farasyn et al. (2011) is representative of these issues. It discusses the implementation of various inventory management solutions at Procter and Gamble (P&G). Given the company has 500 different supply chains, it chose a two pronged approach to realize improvements in inventory levels. P&G first focused on the wide-ranging use of spreadsheet-based inventory models throughout its supply chains. This part of the implementation involved four methods that can locally optimize different parts of the supply chains. The next step dealt with the deployment of integrated multi-echelon inventory software in the company's more complex supply chains. The use of OR/MS technologies led to savings of $1.5 billion in 2009, while service levels were maintained or increased. The authors also highlight the fact that this successful implementation did not rely on tools alone. A buy-in from the various entities involved was of equal importance and so was the fit between the necessities of a business unit and the inventory techniques it will use.

Supply Chain Management

Corporations have evolved from the vertical management of separate individual functions to the horizontal management across all functions. Many of

the old conflicts among business units, including transportation versus inventory have given way to the concept of the total logistics cost. Supply chain management is the natural progression of applying these concepts throughout distribution channels by means of pipeline inventory management and information sharing by all involved parties.

The implementation of a comprehensive set of OR/MS tools in a variety of business areas of a large oil company is discussed in Klingman et al. (1987). It is not surprising to see that this industry was at the leading edge of computer integrated horizontal management across functional areas. OR/MS techniques such as linear programming have been utilized in the oil industry since the 1950s. The work of the above authors included such tools as mathematical programming, statistics, forecasting, expert systems, artificial intelligence, organizational theory, cognitive psychology and information systems. A core element was the optimization-based integrated system for supply, distribution, and marketing. This strategic tool is used to make a number of decisions including how much product to buy or trade, how much to hold in inventory, and how much to ship by each mode of transportation. The system is based on the minimum-cost flow network model.

Since then, supply-chain management has become a key application area for OR/MS methodologies, with an explosive growth in the development of models and algorithms and their implementation. Some researchers took an economics perspective, including game theory and information management approaches, while others examined inventory models. Supply chain configuration has also been at the forefront of research in this area. Researchers have examined the integration and coordination between production and distribution, location and routing, routing and inventory, and routing and crew scheduling. They have proposed a vast assortment of heuristic and optimal methods for these aspects of supply chains and a variety of single and multi-objective decision support systems for the overall system design, (Simchi-Levi et al. 2004).

Sophisticated OR/MS models and algorithms are only part of successful implementations. Ulstein et al. (2006) drive home the idea of the collaboration between business and academia, and business and the community as additional necessary ingredients. Their work was conducted for Elkem's silicon division which is the largest supplier of silicon metal and ferrosilicon in the world. With the slowdown in the global economy that started in 2000, the corporation realized the need to improve the efficiency of its supply chain network and evaluate its product portfolio. To help the division to manage this process, the authors developed a strategic planning model. This mathematical-programming model addresses decisions pertaining to future plant structure, including possible closures, new plant acquisitions, and investments in production equipment. The silicon division has used the model and its scenario analysis capabilities extensively to obtain important benefits. The company agreed to a restructuring process, that included reopening a closed furnace and investing $17 million in equipment conversion. Overall, as a result of the restructuring plan, Elkem has achieved significant and sustained improvements in yearly revenue for the silicon division. Many companies face supply-chain design problems with a similar level of complexity. They can benefit from following the close collaborative process described in this paper and from using optimization tools to solve their decision problems.

Sustainability issues are becoming a requisite part of a supply chain studies. For example, Nagurney and Nagurney (2010) consider a company's multicriteria decision problem that attempts to minimize the total costs associated with its supply chain activities, along with the emissions generated by its manufacturing, storage and distribution facets. The business incurs both capital and operational costs. The authors propose a network optimization framework and illustrate an algorithm applied to a number of sustainable supply chain examples. Carter and Easton (2011) trace the evolution of the field from the original research on social and environmental areas, to issues of corporate social responsibility, and the eventual realization that sustainability is part of the bottom line. They provide a comprehensive review of the sustainable supply-chain management literature. One of the salient features of the paper is the relationship between supply chain risk management and contingency planning and sustainable supply chains.

See

- Airline Industry Operations Research
- Crew Scheduling
- Facility Location
- Inventory Modeling
- Multicommodity Network Flows
- Network
- Scheduling and Sequencing
- Supply Chain Management
- Vehicle Routing

References

Blais, J. Y., Lamont, J., & Rousseau, J.-M. (1990). The HASTUS vehicle and manpower scheduling system at the societé de transport de la communauté urbaine de Montréal. *Interfaces, 20*(1), 26–42.

Blumenfeld, D., et al. (1987). Reducing logistics costs at general motors. *Interfaces, 17*(1), 26–47.

Braklow, J., et al. (1992). Interactive optimization improves service and performance for yellow freight system. *Interfaces, 22*(1), 147–172.

Butchers, E., et al. (2001). Optimized crew scheduling at Air New Zealand. *Interfaces, 31*(1), 30–56.

Carter, C., & Easton, P. (2011). Sustainable supply chain management: Evolution and future directions. *International Journal of Physical Distribution and Logistics Management, 41*(1), 46–62.

Cohen, M., et al. (1990). Optimizer: IBM's multi-echelon inventory system for managing service logistics. *Interfaces, 20*(1), 65–82.

Desaulniers, G., et al. (1998). A unified framework for deterministic time constrained vehicle routing and crew scheduling problems. In T. Crainic & G. Laporte (Eds.), *Fleet management and logistics* (pp. 57–93). Norwell, MA: Kluwer.

Farasyn, I., et al. (2011). Inventory optimization at Procter & Gamble: Achieving real benefits through user adoption of inventory tools. *Interfaces, 41*(1), 66–78.

Gendreau, M., & Potvin, J.-Y. (2004). Issues in real-time fleet management. *Transportation Science, 38*(4), 397–398.

Golden, B., Raghavan, S., & Wasil, E. (Eds.). (2008). *The vehicle routing problem: Latest advances and new challenges*. New York: Springer.

Guide, D., & Van Wassenhove, L. (2009). The evolution of closed-loop supply chain research. *Operations Research, 57*(1), 10–18.

Haase, K., Desaulniers, G., & Desrosiers, J. (2001). Simultaneous vehicle and crew scheduling in urban mass transit systems. *Transportation Science, 35*(3), 286–303.

Johnson, M. (2006). Supply chain management: Technology, globalization, and policy at a crossroads. *Interfaces, 36*(3), 191–193.

Kant, G., Jacks, M., & Aantjes, C. (2008). Coca-cola enterprises optimizes vehicle routes for efficient product delivery. *Interfaces, 38*(1), 40–50.

Klingman, D., et al. (1987). The successful deployment of management science throughout citgo petroleum corporation. *Interfaces, 17*(1), 4–25.

Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science, 43*(4), 408–416.

Larsen, A., Madsen, O., & Solomon, M. M. (2004). The a priori dynamic traveling salesman problem with time windows. *Transportation Science, 38*(4), 459–472.

Moore, E., Warmke, J., & Gorban, L. (1991). The indispensable role of management science in centralizing freight operations at Reynolds metals company. *Interfaces, 21*(1), 107–129.

Nagurney, A., & Nagurney, L. (2010). Sustainable supply chain network design: A multicriteria perspective. *International Journal of Sustainable Engineering, 3*(3), 189–197.

Powell, W., Jaillet, P., & Odoni, A. (1995). Stochastic and dynamic routing and networks. In M. Ball et al. (Eds.), *Handbooks in operations research/management science* (Vol. 8, pp. 141–295). Amsterdam, The Netherlands: Elsevier.

Sandhu, R., & Klabjan, D. (2007). Integrated airline fleeting and crew-pairing decisions. *Operations Research, 55*(3), 439–456.

Simão, H., George, A., & Powell, W. (2010). Approximate dynamic programming captures fleet operations for Schneider national. *Interfaces, 40*(5), 342–352.

Simchi-Levi, D., Chen, X., & Bramel, J. (2004). *The logic of logistics: Theory, algorithms, and applications for logistics and supply chain management* (2nd ed.). New York: Springer.

Toth, P., & Vigo, D. (eds) (2002). The vehicle routing problem. *SIAM Monographs on discrete mathematics and applications, society for industrial and applied mathematics*, Philadelphia.

Ulstein, N., et al. (2006). Elkem uses optimization in redesigning its supply chain. *Interfaces, 36*(4), 314–325.

Yu, G., et al. (2003). A new era for crew recovery at continental airlines. *Interfaces, 33*(1), 5–22.

Log-Linear Model

- Learning Curves
- Regression Analysis

Longest-Route Problem

In a directed network, the finding of the longest route between two nodes is the longest-route problem. In an acyclic network, one that represents the precedence

relationships between activities in a project, the longest route in the network represents the critical path, with the value of the longest route equal to the value of the earliest completion time of the project.

See

- ▶ Critical Path Method (CPM)
- ▶ Program Evaluation and Review Technique (PERT)

Long-Tailed Distribution

- ▶ Heavy-Tailed Distribution

Loss Function

- ▶ Decision Analysis
- ▶ Total Quality Management

Lottery

In utility theory and decision analysis, a lottery consists of a finite number of alternatives of prizes $A_1 \ldots A_n$ and a chance mechanism such that prize A_i will be an outcome of the random experiment with probability $p_i \geq 0, \Sigma_i p_i = 1$.

See

- ▶ Decision Analysis
- ▶ Utility Theory

Lower-Bounded Variables

The condition $l_j \leq x_j$, $l_j \neq 0$, defines x_j as a lower-bounded variable. Such conditions are often part of the constraint set of an optimization problem. For linear programming, these conditions can be removed explicitly by appropriate transformations, given that the problem is feasible when $x_j = l_j$ for each j.

Lowest Index Anticycling Rules

- ▶ Bland's Anticycling Rules

LP

- ▶ Linear Programming

LU matrix decomposition

The decomposition of a matrix into the product of a lower- and an upper-triangular matrix. This is similar to an $\boldsymbol{LDU}$ decomposition in which the $\boldsymbol{D}$ and $\boldsymbol{U}$ matrices have been combined.

See

- ▶ LDU Matrix Decomposition

M

Machine Learning

A term used in the artificial intelligence community to indicate automated improvement based on experience or empirical data in accomplishing a given task such as optimizing an objective function.

See

► Artificial Intelligence

MAD

Mean absolute deviation.

Maintenance

Maintenance is the support of successful system operation during long periods of usage by means of: (1) regular or sample check-ups; (2) planned or preventive replacement of the system's units; (3) failure diagnosis; and/or (4) spare units supply. Operations research models for a system maintenance analysis are represented mainly by optimization models for the improvement of system and equipment reliability.

For (1) and (2), one usually uses methods of controlled stochastic processes. For (3), one uses special methods based on mathematical logic, while (4) is considered in the scope of optimal redundancy and inventory control.

See

► Airline Industry Operations Research
► Inventory Modeling

References

Ushakov, I. A. (Ed.). (1994). *Handbook of reliability engineering*. New York: Wiley.

Makespan

► Scheduling and sequencing

Malcolm Baldrige Award

► Total Quality Management

Manhattan Metric

► Location Analysis

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Manpower Planning

David J. Bartholomew
The London School of Economics and Political Science, London, UK

Introduction

Manpower (or, human resource) planning is concerned with the quantitative aspects of the supply of and demand for people in employment. At one extreme this might include the whole working population of a country, but it has been most successful when applied to smaller, more homogeneous systems like individual firms or professions. The term manpower planning appears to date from the 1960s though many of the ideas can be traced back much further. In recent years terms such as Workforce Panning and Personnel Planning have been used in the same sense. A history of the subject up to the 1980s,from a U.K. perspective, will be found in Smith and Bartholomew (1988). The literature of the subject is very scattered reflecting the diverse disciplinary origins of the practitioners, but most of the technical material is to be found in the journals of operations research, probability, and statistics. There was an initial surge of publication in the late 1960s and early 1970s and since then book length treatments include Grinold and Marshall (1977), Vajda (1978), and Bennison and Casson (1984). Bartholomew, Forbes and McClean (1991) gives a thorough coverage of the technical material and contains an extensive bibliography. Since then there has been a period of consolidation. The earlier theoretical work has largely proved adequate for practical needs, though there have been developments in closely related areas. See, for example, Kalamatianou and McLean (2003).

The essence of manpower planning is summed up in the aphorism that its aim is to have the right numbers of people of the right kinds in the right places at the right time. The basic approach is first to classify the members of a system in relevant ways. These will often be on the basis of such things as grade, salary level, sex, qualifications, and job title. The state of the system at any point in time can then be described by the numbers in these categories, often referred to as the stocks. Over time, changes occur as individuals join, leave the system or move within it. The numbers making these transitions are called the flows. The factors giving rise to change may be predictable or unpredictable but will include such things as individual decisions to leave, changes in demand for goods, management decisions on promotion or organizational structure and so on. The operations researcher's role is to describe and model the system as a basis for optimizing its performance.

Stochastic Models

The presence of uncertainty in so many aspects of the functioning of a manpower system means that any adequate model has to be stochastic. Two probability processes, in particular, have proved to be both flexible and realistic. These are the absorbing Markov chain and the renewal process. The former is appropriate in systems where the stocks are free to vary over time under the impact of constant flow rates, or probabilities. The art of successful application is to define the classification of individuals that all those within a category have approximately the same probability of moving to any other category. Loss from the system corresponds to absorption, and the theory of Markov chains can then be used to predict future stock numbers for various sets of transition probabilities. Later work has extended these methods by allowing the intervals between transitions to be random variables in which case a semi-Markov process or a Markov renewal process results.

When the numbers in the categories are fixed, as they often are when the categories are grades or based on job function, a different approach must be used. Transitions cannot then be regarded as generated by fixed probabilities, but arise in response to the occurrence of vacancies. The result is a replacement, or renewal process, where movement is driven by wastage (or the creation of new places). It was shown in Bartholomew, Forbes and McClean (1991) that the flows of vacancies could be modelled by a Markov chain in a manner very similar to that used for the modeling of the flows of people.

If a system is relatively small or if the rules governing its operation are complex, the only realistic way to model it may be to use a computer-based simulation model. The term simulation is commonly

used in two distinct senses in this context. Primarily it means that each individual movement is generated in the model by a random mechanism. Secondly, it is sometimes used of any algorithm for computing the aggregate properties of a system treated deterministically.

Forecasting and Control

Broadly speaking all models may be used in two modes for forecasting or control. In the early stages of a study one usually wishes to forecast the future state of the system if current trends continue. Next, it will usually be desirable to carry out a sensitivity analysis to explore the consequences of variations from present conditions. This leads on to questions of control where the question is how those parameters under management control should be chosen to achieve some desired goal. The distinction between forecasting and control can be illustrated using a simple form of the Markov model. According to this model successive vectors of expected stocks are related by an equation of the form

$$\boldsymbol{n}(T+1) = \boldsymbol{n}(T)\boldsymbol{P} + \boldsymbol{R}$$

where T represents time, $\boldsymbol{P}$ is a matrix of transition probabilities, and $\boldsymbol{R}$ is a vector of recruitment numbers. In forecasting mode, estimated or guessed values of $\boldsymbol{P}$ and $\boldsymbol{R}$ could be used to predict future values of $\boldsymbol{n}(T)$. In principle, $\boldsymbol{P}$ and $\boldsymbol{R}$ could both depend on T. In control mode, one would be asking how some or all of the elements of $\boldsymbol{P}$ and $\boldsymbol{R}$ should be chosen to attain a given $\boldsymbol{n}$ within a specified time. This gives rise to questions of attainability (whether the problem is solvable) and maintainability (whether an $\boldsymbol{n}$ can be maintained once it is reached). These matters have led to an interesting set of theoretical questions about the solvability of such problems in deterministic or stochastic environments. At a more practical level it has led to the formulation of optimization problems expressed in goal programming and/or network analysis terms (Gass 1991; Klingman and Phillips 1984).

The wastage flow (also known as attrition or turnover) is an important element in a manpower system both because it is highly variable and, largely, beyond the control of management. It has been intensively studied mainly through the survivor function or, equivalently, the frequency distribution of completed length of service. In practice the analysis is complicated by the fact that the data are usually censored and sometimes truncated also. This work has three main objectives: measurement, prediction, and gaining insight into the factors determining wastage.

The demand side of the manpower equation has proved to be less tractable. Demand for people is equivalent to the supply of jobs and this depends on technological, political, social, and economic factors many of which may be specific to particular organizations or industries. To take only one example, the demand for qualified medical manpower will depend on such varied things as demographic changes, the willingness of government or users of the service to pay, and the appearance and spread of new diseases like AIDS. The methods used have been, and have to be, as diverse as the fields of application. Because of the considerable uncertainties involved it is important to monitor constantly the changing environment and to adjust plans accordingly. A once-and-for-all plan has no place in manpower planning.

See

- ▶ Goal Programming
- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Network

References

Bartholomew, D. J., Forbes, A. F., & McClean, S. I. (1991). *Statistical techniques for manpower planning* (2nd ed.). Chichester: Wiley.

Bennison, M., & Casson, J. (1984). *The manpower planning handbook*. London: McGraw-Hill.

Gass, S. I. (1991). Military manpower planning models. *Computers and OR, 18*(1), 65–73.

Grinold, R. C., & Marshall, K. T. (1977). *Manpower planning models*. New York/Amsterdam: North-Holland.

Kalamatianou, A. G., & McClean, S. (2003). The perpetual student: Modeling duration of undergraduate studies based on lifetime-type educational data. *Data Analysis, 9*, 311–330.

Klingman, D., & Phillips, N. (1984). Topological and computational aspects of preemptive multicriteria military personnel assignment problems. *Management Science, 30*, 1362–1375.

Smith, A. R., & Bartholomew, D. J. (1988). Manpower planning in the United Kingdom: An historical review. *Journal of Operational Research Society, 9*, 235–248.
Vajda, S. (1978). *Mathematics of manpower planning*. Chichester: Wiley.

Manufacturing

- ▶ Flexible Manufacturing Systems
- ▶ Operations Management
- ▶ Production Management

MAP

Markov arrival process.

See

- ▶ Matrix-Analytic Stochastic Models

Marginal Value

The marginal value is the extra cost of producing one extra unit of output. Similarly, marginal revenue is the extra revenue resulting from selling an extra unit of goods. From the economics of a firm, when marginal revenue equals marginal costs, the firm is in an equilibrium optimal condition in terms of maximizing profits. Depending on the application, the dual variables of a linear-programming problem can be interpreted as marginal values. The economic interpretation of the dual variables is complicated by alternate optimum solutions (corresponding to different bases) that may yield different values of the dual variables. Thus, there may be two or more marginal values for the same constraint. Such multiple values must be interpreted with care.

See

- ▶ Dual Linear-Programming Problem
- ▶ Duality Theorem

Marketing

Yoram (Jerry) Wind[1], Eric T. Bradlow[1], Jehoshua Eliashberg[1], Gary L. Lilien[2], Jagmohan Raju[1], Arvind Rangaswamy[2] and Berend Wierenga[3]
[1]University of Pennsylvania, Philadelphia, PA, USA
[2]The Pennsylvania State University, University Park, PA, USA
[3]Erasmus University Rotterdam, Rotterdam, The Netherlands

Introduction

Marketing offers a rich and unique domain for applications of operations research (OR) methods, models, and approaches. Not only does the marketing area offer opportunities to develop and apply OR models and methods to increasingly important decisions affecting ALL companies, nonprofits, governments, societies, and individuals, but also unique opportunities to further the much needed collaboration between academics and practitioners, and for bridging the silos between marketing and the other management disciplines and functions.

Since customers (individuals or groups) are at the heart of the marketing system, OR modeling approaches help characterize, understand, and predict their behaviors. For consumers and organizational buyers, that behavior involves the search for solutions to a want or desire, the screening or evaluation of alternatives, the selection of a best alternative, the act of purchase, the post-purchase feedback to the firm as well as to other customers and learning that affects future purchasing behavior. In fact, such applications of OR to marketing problems have become even more prevalent, with website morphing (Hauser et al. 2009), Netzer's work on optimal email campaigns, and optimal in-store movement using the traveling salesman paradigm (Hui et al. 2009).

Firms and other non profit organizations (such as museums, politicians, government organizations) capitalize on that knowledge or model of individual behavior by focusing on such decisions as product/service design, pricing, distribution, promotion, advertising, personal selling, and the likely customer responses to them. In addition, at a higher level, these

decisions must be integrated and coordinated with the activities of other management functions (finance, manufacturing, R&D, etc.) and linked to other product and market decisions of the organization, including the critical resource allocation decisions among products, markets, distribution options, and businesses. Such critical decisions are evaluated based on their return on investment (ROI) under alternative scenarios reflecting different views of the future.

The external scenarios range from pessimistic views of recurring financial crisis, catastrophic natural disasters, continued terrorist activities and political unrest around the world, through continuation of the status quo, to optimistic scenarios of growth and prosperity driven by the fast growing economies of Asia and a recovery of the West. For marketers, these scenarios lead to consideration of strategic alternatives derived from a narrow view of modeling, e.g., the impact of a specific marketing activity (such as advertising expenditures) through an integrated view of all marketing touch points and product/service/solutions/customer experience, to the design of full strategy integration across the various management functions, incorporating multiple short and long-term performance measures.

Background

The American Marketing Association defines marketing as: "... the process of planning and executing the conception, pricing, promotion, and distribution of ideas, goods, and services to create exchanges that satisfy individual and organizational objectives."

As a management function, marketing includes such activities as advertising, sales and marketing research. Or, more simply put, marketing's organizational role is the interface between a firm and its customers. It is also a critical participant in cross-functional processes aimed at developing and launching new products and services that create customer value, i.e., products and services that customers want.

As a philosophy, marketing views the need to understand, anticipate and meet customer needs as the key to organizational success. As such, the customer is the final arbitrator of the value of any product or service offering. Marketing philosophy also extends the concept of customer orientation to internal customers and other stakeholders.

Thus, marketing is concerned with anticipating and understanding human needs and wants and translating those needs and wants into the demand (as economists use the term) for products and services. Those needs and wants are satisfied with products and services that are increasingly being developed in collaboration with empowered consumers. Businesses that exemplify this view include Build-a-Bear, Dell, and others offering opportunities for customization of the products and services, as well as firms that now scrape blogs, discussion forums, and other user-generated content to bring the digital voice of the customer into the firm, and help determine the appropriate responses (Ghose and Han 2011).

Products and services have functional as well as image characteristics. They are made available to the customer through a variety of channels ranging from physical retail stores to online websites, to mail order to social network platforms (e.g., Facebook). In order to effect an exchange, individuals have to be aware of, emotionally engaged, and understand the product (through advertising or other communication media), find the product worth their money (by comparing the product's total cost — its purchase price adjusted by any promotional offerings plus the cost of maintaining, using and disposing of the product — with the benefits promised in terms of performance and image), and participate in the exchange process.

While historically marketing models of behavior saw a product's value as consisting of the sum of the utilities of the features and benefits of which it is comprised (Green et al. 1973), and that is still a significant part of marketing modeling, newer conceptual models also take into account the perceived value of others, the recommendation of others, and the ability to share those experiences with others via one's social network (Stephen and Toubia 2010), or via the network externalities generated by other adopters.

Exchanges have typically been aggregated to the context of a market segment, which consists of the customers sharing a particular and similar need and who are willing to engage in exchange to satisfy that need. However, it is no longer uncommon to see exchange activities take place between the firm and individual consumers rather than at the level of a market segment (which represents higher level of aggregation). In essence, technology has allowed marketing in the 21st century to be infinitely tailored

because of the wealth of individual-level data that is now tractable due to the advances in the interactive media, and consumers' motivation and ability to customize the offerings.

OR Marketing Model Types

OR in marketing helps decision makers by harnessing measurement models and theoretical models and embedding them within a decision model (or more generally, within a decision-support system). The corresponding models are called **measurement models**, **stylized theoretical models**, and **decision-making models**, respectively (although it may be equally helpful to interpret these categories as classification dimensions for interpreting the multiple purposes of models).

Measurement Models — The purpose of measurement models is to describe and predict a current or anticipated either an individual consumer or the market reaction to a product or service as a function of various independent variables. The phrase "market reaction" here should be interpreted broadly. It is not necessarily units demanded but could be some other related variables. For example, in Guadagni and Little's (1983) model, the dependent (reaction) variable is the probability that the individual will purchase a given brand on a given purchase occasion. Choice models often have several independent variables including whether the brand was on sale (deal) at a given purchase occasion, regular price of the brand, deal price (if any), brand loyalty of the individual, etc. In addition, sometimes the focus of such models may be on certain variables preceding the steady-state demand (e.g., awareness, first-trial, repeat purchase). These examples suggest that measurement models can deal with individual (disaggregate) demand or aggregate (segment or market-level) demand as well as transitory or steady-state demand. Note that advances in measurement models can be due to better data (e.g., scanner data) or better estimation methods and procedures (maximum-likelihood methods for generalized logit models, for example). In traditional marketing problems such as customer satisfaction and customer-defined quality, OR measurement models have greatly enhanced the relatively simplistic survey-based approaches to the measurement of these constructs. Relying on advances in structural equation modeling. as well as the new area of empirical industrial organization, allows researchers to address more realistic and rich problems, such as competitive pricing behavior in markets with a large number of products (e.g., Sudhir 2001).

Stylized Theoretical Models — The purpose of stylized theoretical models is to explain and provide insights into marketing phenomena: a stylized theoretical model typically begins with a set of assumptions that describes a particular marketing environment. Some of these assumptions may be purely mathematical, but are also intuitively logical with the objective of making the analysis tractable. Others are substantive assumptions with real empirical grounding. Two well-known theoretical modeling efforts are Bell, Keeney and Little (1975), who show what functional forms of market share models are consistent with a certain set of reasonable criteria, and Basu et al. (1985), who show what form of sales force compensation plan is optimal under a set of assumptions about firm and salesperson objectives and behavior.

Such stylized theoretical models have helped improve the ability to design optimal product lines, issues related to specialization versus vertical integration (McGuire and Staelin 1983), aligning the incentives between manufacturers and retailers (Jeuland and Shugan 1983), designing pricing strategies for traditional goods, and also information goods. Stylized models have helped improve how companies offer short-term price discounts (Raju et al. 1990), how such short-term price discounts pass through to the consumer, and how retailers might improve their private label offerings. As marketing systems evolved, especially with the advent of new technologies, such stylized models have significantly improved understanding of new platforms and mechanisms for interactions between buyers and sellers. Stylized theoretical models have also helped in the understanding of the role brands play in a competitive market, including the symbolic role that brands play in social interactions, and how firms may improve their advertising and communications strategies (Chen et al. 2009).

While the emphasis in this work is on developing stylized theoretical models, most work in this area also rigorously tests the ability of these models to predict firm and market behavior. Recent empirical work in

Marketing, Table 1 The frontiers of decision models (DM)

	DM frontiers today	DM frontiers tomorrow
Time Scale	Days and weeks, if not months	Moving toward real time in data entry, data access, data analysis, implementation, and feedback
Focus of DM	Support strategic decisions	Support both strategic and operational decisions
Mode of Operation	Individual and PC-centric	Organization and Network centric – support multiple employees in multiple locations on multiple devices
Decision Domain	Marketing	Marketing and other functions, such as Supply Chain and Finance
Company Interface	Loosely coupled to company's IT systems	Woven into IT-supported company's operations and decision processes
Intervention Opportunities	Discrete, Problem-driven	Continuous, Process-driven
DM Goal	Support analysis and optimization	Support robust and adaptive organizational decision processes
DM System Design	As a tool to understand information and enhance decisions	As tool to enhance productivity and success of business models
DM System Operation	Interactive (User interacts with model)	Interactive as well as autonomous (embedded)
DM Outputs	Recommended actions; What if analyses	Visualization of markets and their behavior (e.g., Dashboard), Extended reality (e.g., Business model simulation), Explanation (Why?), Automated implementation (e.g., create alerts, automate actions)
DM Implementation Sequence	Intervention Opportunity → Implementation of decisions → Integration with IT Systems	Integration with IT → Intervention Opportunity → Implementation of decisions

the structural economics also contains stylized theoretical models where observed outcomes are assumed to arise from equilibrium actions taken by agents, modulo stochastic error (Dube et al. 2010). In this manner, joint theory and empirical work has begun to play a larger role. Distinguishing features of stylized theoretical models, especially the ones that use economic modeling and game-theory as tools, are that they explicitly recognize that companies must make decisions in a competitive environment and recognize that they compete with other firms who also are capable of making sound decisions. It is through this explicit recognition that these models are able to provide companies with a theoretically sound and empirically grounded means of improving strategic marketing decisions.

Decision-Making Models — These models are designed to directly help marketing managers make better decisions. They incorporate measurement models as building blocks, but go beyond measurement models in recommending specific actions (e.g., optimal marketing-mix decisions) for the manager. The techniques used to derive the optimal policies vary across applications, and include calculus, dynamic programming, optimal control, and calculus of variations techniques, as well as linear and integer programming. These models have been developed for each marketing variable and for the entire marketing mix program (i.e., a product and service offering including pricing, distribution, etc.). Little's BRANDAID is a classical example of such a model. Lilien et al. (2011) elaborate on the impact such models have had.

Since 2000, many enhanced decision-making models have been developed that are embedded inside enterprise information systems. Examples include revenue management systems used by airlines and hotels and recommender systems used by web sites such as Amazon.com and netflix.com. Table 1, adapted from Lilien and Rangaswamy (2006), summarizes the many ways that decision models are evolving to provide enterprises with real-time and automated decision making capabilities.

The Emergence of Marketing Science

By most accounts, OR in marketing began its growth in the 1960s and 1970s. The literature used a variety of OR methods to address marketing problems: those

problems included product design/development decisions, distribution system decisions, sales force management decisions, advertising and mass communication decisions, and promotion decisions (Kotler 1971). The OR tools that were most prevalent in the 1960s and earlier included mathematical programming, simulation, stochastic processes applied to models of consumer choice behavior, response function analysis, and various forms of dynamic modeling (difference and differential equations, usually of the first order). Some uses of game theory were reported, but most models that included competition used decision analysis, risk analysis, or market simulation games.

Nearly three times the number of marketing articles appeared in the OR literature in the 1970s as appeared in the period from 1952 though 1969. In addition to the increase in the number of articles, reviews by Lilien and Kotler (1983) showed that a number of new areas had begun to emerge. These included descriptive models of marketing decisions, the impact of and interaction of marketing on organizational design, subjective decision models, strategic planning models, models for public and non-profit organizations, organizational buying models, and the emergence of the concept of the Marketing Decision Support System (MDSS). In addition, while the number of published articles rose dramatically, the impact on organizational performance did not appear to be equally significant, raising questions about effective implementation. Much of the literature of the 1970s pointed to the need to expand the domain of application. The "limitations" sections of some of the papers in the 1970s pointed out that many important phenomena that were being overlooked (such as competition, dynamics, and interactions amongst marketing decision variables) were both important and inherently more complex to model. Hence, the level of model complexity and the insightfulness of the analyses in marketing seemed destined to escalate in the 1980s and beyond.

The 1980s saw another more-than doubling of the number of published OR articles in marketing compared to the earlier decade. Two of the areas that produced much of this growth were stylized theoretical models and process-oriented models. The shortening of product life cycles and the impact of competitive reactions in the market place preclude most markets from approaching steady state or equilibrium. Areas of special research focus in that decade included extensive focus on consumer choice models (focusing on the dynamics and heterogeneity of the choice process and the implications for decision making) and the new product area (where the moves and countermoves of competitors keep the marketplace in a constant state of flux).

The 1990s saw new trends in marketing science (and in marketing in general), with the electronic marketplace changing the locus and the nature of the transaction. The concept of the physical marketplace is being replaced by that of market space, and marketing science has found new territories to develop theories and applications. Most of this, of course, is due to the applied nature of the marketing discipline in which solutions to problems emanate from the data and the problem at hand. As the physical marketplace is being replaced by the physical in conjunction with digital marketplace, OR methods that allow for cross-channel optimization are being developed.

The first decade of the 21st century has seen the marketspace/customer centricity trend continue, as customers have gained increased influence and power in all areas of marketing. User-generated content and customers as co-producers and co-marketers are increasingly accepted. Understanding and monitoring these new market structures are central to the new view of marketing. Markets are now made up of customer-networks, and models for understanding and managing such networks are being developed. And the study of the role of the marketing manager and the related decision support systems has evolved from Little's (1979) perspective to a domain of mainstream interest both to academics and practitioners (see Wierenga 2011).

Another important trend is the emergence of two-sided and multi-sided platforms, wherein a business builds a platform that enables many distinct audiences to engage with the business as well as interact with each other, to create economic value, often in the presence of network externalities (Eisenmann et al. 2006). Typically, value appropriation occurs through cross-subsidies, wherein the costs of acquiring one group (e.g., consumers) are subsidized by another group (e.g., advertisers), and the platform itself retains part of the value created. eBay (buyers and sellers), Amazon.com (consumers and affiliates), HMO (patients and doctors), and credit-card payment systems (merchants and consumers) are

often given as examples of platforms. Even many traditional businesses are transforming into platforms that connect players in a complex eco-system (e.g., iPhone as a device connecting application developers with consumers). Cross-subsidies between audiences creates complex transaction flows that offer opportunities for OR modelers to help in carefully managing prices, revenues, and subsidies to optimize business performance.

Trends of OR Use in Marketing

The OR literature in marketing is vast, as reviewed in Lilien and Rangaswamy (2008). Models have been used to explore most facets of marketing and the marketplace, and increasingly marketing research is integrated with appropriate modeling. Some key trends include the following:

1. **OR in marketing is having important impact both on academic development in marketing and in marketing practice**. During the 1980s two new and important journals were started that emphasize the OR approach: *Marketing Science* and the *International Journal of Research in Marketing* (IJRM). Both are healthy, popular, and extremely influential, especially among academics. Another journal, *Quantitative Marketing and Economics* was started in 2003. Together, they reflect the developments of marketing models.
2. **Digital marketing represents vast area of opportunity for OR**. By transforming the market place into marketspace, the revolution in the marketplace brings a host of modeling opportunities and challenges, such as: How are new products and ideas generated, diffused, and discussed in a digital environment? How can a firm manage the natural conflict in physical and electronic distribution channels? How can firms offer different prices to different groups of customers in an electronically linked world? How and when word-of-mouth among consumers evolves? When marketing, manufacturing and the customer are interlinked in the digital environment, what opportunities emerge in the marketing-manufacturing interface? Digital marketing has other major implications, such as the development of new markets (on-line auctions, electronic bargaining) and the possibility of involving customers directly in the development of information products (Dell stores, IBM Jam, and others). More recently, it has become feasible to model large-scale social networks consisting of millions of nodes and billions of links, such as for example to link in near real-time a TV event (e.g., Super Bowl ad) with the Twitter and Facebook feeds triggered by the ad, to potential impact on market outcomes. These provide opportunities to apply OR modeling for analyzing flows of information and influence in such networks to link those to consequences for the firm (e.g., profit) or adverse spread of word of mouth in the marketplace.
3. **New data sources are having a major impact on marketing modeling**. One of the most influential developments of the 1980s and 1990s has been the impact of scanner data on the marketing models field. Scanner data and the closely related single source data (of communication and consumption data) have enabled marketing scientists to develop and test models with much more precision than ever before. Indeed, the very volume of new data has helped spawn tools to help manage the flow of new information inherent in such data. Data mining methods applied to some of the new, massive direct-response data bases has resulted in much more precise customer targeting and promotion-selection procedures. Two new data sources are providing opportunities for OR modelers in marketing: (1) Large integrated data warehouses created by companies to feed enterprise systems, such as CRM, are creating opportunities for developing more fine-grained models that integrate traditional demand side modeling undertaken by marketing modelers with supply side modeling issues such as inventory management, multi-channel logistics, and the like. (2) User-generated data (e.g., online product reviews posted by consumers, social media activities such as twitter feeds) that provide information in real-time about market sentiments offer opportunities for modelers to develop new tools for supporting marketing decision makers. New models for text analysis and synthesis (e.g., to convert reviews into numeric scores representing valence and volume

of sentiments) developed by computer scientists represent a start, but many new opportunities exist in this nascent area to translate huge volumes of raw data into insights for action. Traditional quantitative data sources have been employed by marketing modelers extensively, but more and more attention is now being given to analyzing qualitative and textual data through data and text mining as well as sentiment analysis software packages developed in computer sciences. While the 1990s presented the land of promise for these methods, the 2000s saw it materialize. Thus, the number of people in the information systems area working on traditional marketing problems has increased dramatically, blurring the lines between these related disciplines.

4. **Stylized theoretical modeling is still a mainstream research tradition in marketing**. Stylized models allow researchers to state explicitly as set of assumptions or axioms and then derive theoretical propositions with respect to the phenomena being considered. Such propositions provide valuable managerial insights.
5. **Competition and interaction are major thrusts of marketing models today**. The saturation of markets and the economic fights for survival has changed the focus of interest in marketing models, probably forever. A key-word search of past volumes of *Marketing Science*, *Journal of Marketing Research*, and *Management Science* (marketing articles only) reveals multiple entries for "competition," "competitive strategy," "non-cooperative games," "competitive entry," "late entry," and "market structure." These terms are largely missing in a comparable search in the 1960s and early 1970s.
6. **Marketing research and modeling are facing new challenges**. Both marketing research and modeling, especially as applied to new product development, have to be reformed to address such issues as global scope, electronically interconnected product development sites, the potential for mass customization and rapid prototyping/testing. These issues drive the development of models that incorporate nontraditional customer information, including trade show-participant feedback, user co-development, lead user methods, data and text mining, and Internet panels. Similarly, advertising and marketing mix modeling face comparable challenges, which have led to numerous efforts to develop single source data, related modeling, experiments, and dashboards. Another challenge is to develop models, beyond those developed for the consumer package good industry, that capture adequately various idiosyncratic characteristics of industries such as financial services, entertainment, life sciences, and B2B industries.
7. **Beyond Marketing Analytics—Marketing Engineering**. Marketing analytics, a term that refers to any systematic analysis of marketplace behavior and transactions is giving way to advance marketing analytics or marketing engineering, a term Lilien and Rangaswamy (2006) have popularized to refer to the use of decision models for making marketing decisions. Many of these decisions are now being automated, with decision models making routine pricing and promotion decisions in low-risk stable environments. But the confluence of new data sources, theories, hardware and software, and computer networks has now put these decision models on the desktop of marketing executives everywhere. The use of OR in marketing through marketing engineering is accelerating because of at least six trends (Lilien and Rangaswamy 2008):
   - Investments in infrastructure firms need to maintain extensive, integrated corporate information warehouses (also called data warehouses).
   - The use of On-Line Analytic Processing (OLAP — or just-in-time OR!) to integrate modeling capabilities with data bases.
   - Deploying intelligent systems to automate many modeling tasks.
   - Developing computer simulations for decision training and for exploring multiple options.
   - Installing groupware systems to support group decision making.
   - Enhancing user interfaces to make the use of even complex modeling systems accessible to a wide range of users.
8. **Marketing Management Support Systems and Artificial Intelligence**. A marketing management support systems (MMSS) is defined as any device, combining information technology, analytical

capabilities, marketing data, and marketing knowledge, made available to marketing decision makers with the objective to improve the quality of marketing management (Wierenga and Van Bruggen 2000). Marketing models, with their origin in OR constitute the analytical part of MMSS. However, in marketing there are also many weakly-structured problem areas, where qualitative considerations and judgment are more important. Here, the knowledge and the expertise of the marketer are key resources. Therefore, marketing management support systems not only include the primarily quantitative, data-driven decision-support systems, but also support technologies that are aimed at supporting marketing decision making in weakly structured areas.

9. **Expert Systems**. Marketing expert systems have been developed for many domains of marketing, e.g., (i) to find the most suitable type of sales promotion; (ii) to recommend the execution of advertisements (positioning, message, presenter) (Burke et al. 1990); (iii) to screen new product ideas, and (iv) to automate the interpretation of scanner data, including writing reports. For an overview, see Wierenga and Van Bruggen (2000, Chapter 5). An example of a system especially developed for supporting a particular marketing function is BRANDFRAME. This system supports the decision making of a product or brand manager, which is a typical marketing job. More recently, expert systems in marketing are less often stand-alone systems, but are woven into the company's overall IT systems (Lilien and Rangaswamy 2008).
10. **Neural Networks and Predictive Modeling**. As mentioned earlier, in marketing companies can work more and more with data about individual customers. As a consequence of this development, customer relationship management systems (CRM) became important. An essential element of CRM is the customer database that contains information about each individual customer. This information may refer to socio-economic characteristics (age, gender, education, income), earlier interactions with the customer (e.g., offers made and responses to these offers, complaints, service), and information about the purchase history of the customer (i.e., how much purchased and when). This data can be used to predict the response of customers to a new offer or to predict customer retention/churn. Such predictions are very useful, for example, for selecting the most promising prospects for a mailing or for selecting customers in need of special attention because they have a high likelihood of leaving the company (campaign optimization). A large set of techniques is available for this kind of predictive modeling. Prominent examples are neural networks and classification and regression trees. Both techniques are rooted in artificial intelligence. CRM is a quickly growing area of marketing. Companies want to achieve maximum return on their often large investments in customer databases. (Van Bruggen and Wierenga 2010).
11. **Analogical Reasoning and Case-Based Reasoning (CBR)**. Analogical reasoning plays an important role in human perception and decision making. When confronted with a new problem, people seek similarities with earlier situations and use previous solutions as the starting point for dealing with the problem at hand. Analogical reasoning is also the principle behind the field of case-based reasoning (CBR) in Artificial Intelligence. A CBR system comprises a set of previous cases from the domain under study and a set of search criteria for retrieving cases for situations that are similar (or analogous) to the target problem. Applications of CBR can be found in weakly-structured domains such as architecture, engineering, law, and medicine. By their nature, many marketing problems have a good fit with CBR. A recent application uses CBR as a decision-support technology for designing creative sales promotion campaigns (Van Bruggen and Wierenga 2010).
12. **Adaptive Experimentation**. While OR applications in marketing have been focused on models, given the increased uncertainty, complexity and speed of change of the business environment, it is unlikely that one can model optimal strategies. The alternative to the search for a silver bullet is the adoption of an adaptive experimentation philosophy (Wind 2007) that allows experimentation with a number of innovative strategies, facilitates learning, helps create an innovative organizational culture that reduces the pressures for risk averse decisions, encourages relevant measurement and provides

a competitive advantage. As sophisticated firms such as Google and most direct response companies Increasingly engage in adaptive experimentation, a new role for many of the OR marketing models (including marketing mix models) is in suggesting hypotheses that guide the experimental variables and design. Adaptive experimentation is consistent with the philosophy of OR and should be considered in any portfolio of approaches to aid decision makers in making better decisions.

13. **Documentation of the Impact of OR Marketing Models on the Organization is Now Mainstream**. The emergence of the INFORMS Society for Marketing Science Practice Prize and work by Lilien (2011) and Wierenga (2011) have underscored the need to study how marketing integrated with the concepts of OR can become a mainstream research domain for marketing academics while having a greater impact on the operations of firms. According to a *Business Week* article in 2010, the Fortune 1,000 companies spend over $1 trillion in marketing annually. Yet, according to a McKinsey report (2009), most of these companies do not use marketing models to improve their marketing investment related decision making, even though the small percentage of companies that do (17% of B2C and 7% of B2B) seem to realize considerable benefits from their use. In a controlled experimental study, Lilien shows that the managers using decision models realize measurable improvements in decision performance when compared to managers who have access to the same data, but without a decision-support model to optimally interpret the data. Research is ongoing on what factors influence companies to deploy marketing models, under what conditions their impact is maximized, and how decision tools should be designed to enhance their usability and impact.

Concluding Remarks

OR/marketing models and approaches have had significant impact on academic research and practice. Marketing science has also been used to address important societal problems, e.g., Bradlow (2009) discusses the use of marketing science to aid in creatively solving problems related to the financial crisis. Developments in constructing, testing and applying new marketing science models will continue to benefit management and society.

See

- ▶ Advertising
- ▶ Data Mining
- ▶ Decision Analysis
- ▶ Electronic Commerce
- ▶ Game Theory
- ▶ Linear Programming
- ▶ Operations Management
- ▶ Retailing

References

Basu, A., Lal, R., Srinivasan, V., & Staelin, R. (1985). Sales force compensation plans: An agency theoretic perspective. *Marketing Science, 4*, 267–291.

Bell, D. E., Keeney, R. L., & Little, J. D. C. (1975). A market share theorem. *Journal of Marketing Research, 12*, 136–141.

Bradlow, E. T. (2009). Marketing science and the financial crisis. *Marketing Science, 28*(2), 201.

Burke, R. R., Rangaswamy, A., Wind, J., & Eliashberg, J. (1990). A knowledge-based system for advertising design. *Marketing Science, 9*(3), 212–229.

Chen, Y., Yogesh, J., Raju, J. S., & Zhang, J. (2009). A theory of combative advertising. *Marketing Science, 28*, 1–19.

Dube, J. P., Hitsch, G. J., & Chintagunta, P. K. (2010). Tipping and concentration in markets with indirect network effects. *Marketing Science, 29*(2), 216–249.

Eisenmann, T., Parker, G., & Van Alstyne, M. W. (2006). Strategies for two-sided markets. *Harvard Business Review*, October 2006, 1–10.

Ghose, A., & Han, S. (2011). An empirical analysis of user content generation and usage behavior on the mobile internet. *Management Science, 57*(9), 1671–1691.

Green, P. E., Wind, Y., & Carroll, D. (1973). *Multi-attribute decisions in marketing: A measurement approach*. Hinsdale: The Dryden Press.

Guadagni, P., & Little, J. D. C. (1983). A Logit model of brand choice calibrated on scanner data. *Marketing Science, 2*, 203–238.

Hauser, J. R., Urban, G. L., Liberali, G., & Braun, M. (2009). Website morphing. *Marketing Science, 28*(2), 202–223.

Hui, S., Bradlow, E., & Fader, P. (2009). The traveling salesman goes shopping: The systematic deviations of grocery paths from TSP-optimality. *Marketing Science, 28*(3), 566–572.

Jeuland, A. P., & Shugan, S. M. (1983). Managing channel profits. *Marketing Science, 2*(3), 239–272.

Kotler, P. (1971). *Marketing decision making: A model building approach*. New York: Holt, Rinehart and Winston.

Lilien, G. L. (2011). Bridging the academic-practitioner divide in marketing decision models. *The Journal of Marketing, 75*, 196–210.
Lilien, G. L., & Kotler, P. (1983). *Marketing decision making: A model building approach*. New York: Harper and Row.
Lilien, G. L., & Rangaswamy, A. (2006). Marketing decision support models: The marketing engineering approach. In R. Grover & M. Vriens (Eds.), *The handbook of marketing research: Uses, misuses, and future advances*. Thousand Oaks, CA: Sage Publications.
Lilien, G. L., & Rangaswamy, A. (2008). Marketing engineering: Models that connect with practice. In B. Wierenga (Ed.), *Handbook of marketing decision models: International series in operations research & management science*. New York: Elsevier Press.
Little, J. D. C. (1979). Decision support systems for marketing managers. *Journal of Marketing, 43*(3), 9–27.
McGuire, T. W., & Staelin, R. P. (1983). An Industry equilibrium analysis of downstream vertical integrations. *Marketing Science, 2*(2), 161–192.
Raju, J. S., Srinivasan, V., & Lal, R. (1990). Effects of brand loyalty on competitive price promotional strategies. *Management Science, 36*, 276–304.
Stephen, A., & Toubia, O. (2010). Deriving value from social commerce networks. *Journal of Marketing Research, 47*(2), 215–228.
Sudhir, K. (2001). Competitive pricing behavior in the auto market: A structural analysis. *Marketing Science, 20*(1), 42–60.
Van Bruggen, G. H., & Wierenga, B. (2010). Marketing decision making and decision support: Challenges and perspectives for successful marketing management support systems. *Foundations and Trends in Marketing, 4*(4), 126.
Wierenga, B. (2011). Managerial decision making in marketing: The next research frontier. *International Journal of Research in Marketing, 28*(2), 89–101.
Wierenga, B., & Van Bruggen, G. H. (2000). *Marketing management support systems: Principles, tools, and implementation*. Boston: Kluwer Academic.
Wind, J. (2007). Marketing by experiment. *Marketing Research*, Spring 2007, 10–16.

Markov Chain Equations

William J. Stewart
North Carolina State University, Raleigh, NC, USA

Introduction

For a continuous-time Markov chain, the probability distribution at any time t, $\boldsymbol{\pi}(t)$, is calculated from the Chapman-Kolmogorov differential equation,

$$\frac{d\boldsymbol{\pi}(t)}{dt} = \boldsymbol{\pi}(t)\boldsymbol{Q}. \tag{1}$$

where the vector $\boldsymbol{\pi}(t)$ is of length n, the number of possible states in the Markov chain, and its ith component, $\pi_i(t)$, expresses the probability that the Markov chain is in state i at time t, and $\boldsymbol{Q}$ is the infinitesimal generator or transition rate matrix, a square matrix of order n whose elements satisfy

$$q_{ij} \geq 0, \quad i \neq j;$$
$$q_{ii} = \sum_{j=1,\, j\neq i}^{n} q_{ij}, \quad \text{for all } i = 1,\ 2,\ \ldots,\ n.$$

When the number of states in the Markov chain is relatively small (e.g., less than a thousand), computing numerical solutions of the chain equations is generally easy, and (1) can be solved readily by software such as MATLAB. But two difficulties arise when the number of states is large: The first is the sheer size of the matrices involved; the second is how well-conditioned or how ill-conditioned the equations are. These difficulties exist even in the simpler setting considered here when all that is required is the stationary solution of the Markov chain obtained by setting the left-hand side of (1) to zero and solving the linear system of equations that results.

It is not unusual for the number of states in a Markov chain model to exceed the millions. Such size impacts both the storage of the matrix and the number of vectors needed to compute the solution. Very large matrices cannot be stored in the usual two-dimensional array format; there is simply not enough storage space available. In addition, this would be very wasteful, since most of the matrix elements are zero. In general, each state communicates directly with only a small number of states and so the number of nonzero elements in the matrix is usually equal to a small multiple of the number of states. If the states can be ordered sequentially so that each communicates only with its closest neighbors, then the nonzero elements of Q lie close to the diagonal and a banded storage technique can be used. Otherwise, it is usual to store only the nonzero elements in a double-precision one-dimensional array and use two integer one-dimensional arrays to indicate the position of each nonzero element in the matrix. In addition to storing the transition matrix, a certain

number of double-precision vectors, of size equal to the number of states, is also needed. In the simplest numerical methods, two such vectors suffice. In other more sophisticated methods, many more (possibly in excess of 50) may be needed.

A second difficulty in solving Markov chains numerically is that of the degree of ill-conditioning of Q. In certain models, the difference in the rates at which events can occur may be many orders of magnitude, as is the case when a model allows for both human interaction and electronic transactions. These differences in magnitude may lead to ill-conditioned systems, that is to say, a small change in one of the parameters can result in a large change in the solution. It is appropriate to distinguish between numerical conditioning and numerical stability; the first has already been described and is a function of the problem itself; the second describes the behavior of an algorithm in attempting to compute solutions. A stable algorithm will not allow the error to grow out of proportion to the degree of ill-conditioning of the problem. In other words, a stable algorithm will give as good a solution as can be expected for the particular problem to be solved. A further effect of large differences in transitions rates is that they can create convergence problems for iterative solution methods.

Numerical Methods for Computing Stationary Distributions

The goal is to solve the matrix equation

$$\boldsymbol{\pi Q} = 0. \tag{2}$$

By setting $\boldsymbol{P} = \boldsymbol{Q}\Delta t + \boldsymbol{I}$, where $\Delta t \leq (\max_i |q_{ii}|)^{-1}$, this equation may be written as

$$\boldsymbol{\pi P} = \boldsymbol{\pi}. \tag{3}$$

In carrying out this operation, the continuous-time system represented by the transition rate matrix, $\boldsymbol{Q}$, is essentially converted to a discrete-time system represented by the stochastic transition probability matrix, $\boldsymbol{P}$. In the discrete-time system, transitions take place at intervals of time Δt, this parameter being chosen so that the probability of two transitions taking place in time Δt is negligible. The stationary distribution π may be computed from either of these equations.

Direct Methods — Since Eq. (2) is a homogeneous system of linear equations, one may use standard linear solution methods based on Gaussian elimination. Assume that the Markov chain is ergodic. In this case, the fact that the system of equations is homogeneous does not create any problems, because any of the n equations can be replaced by the n normalizing equation, $\sum_{j=1}^{n} \pi_j = 1$, and thereby convert it into a nonhomogeneous system with nonsingular coefficient matrix and nonzero right hand side. The solution in this case is well defined. It turns out that replacing an equation with the normalizing equation is not really necessary.

The usual approach taken is to construct an $\boldsymbol{LU}$ decomposition of $\boldsymbol{Q}$ and replace the final zero diagonal element of $\boldsymbol{U}$ with an arbitrary value. The solution computed by back substitution on $\boldsymbol{U}$ must then be normalized. Furthermore, since the diagonal elements are equal to the negated sum of the off-diagonal elements ($\boldsymbol{Q}$ is, in a restricted sense, diagonally dominant), it is not necessary to perform pivoting while computing the $\boldsymbol{LU}$ decomposition. This simplifies the algorithm considerably.

The problems of the size and nonzero structure (the placement of the nonzero elements within the matrix) still remain. Obviously this method works well when the number of states is small. It will also work well when the nonzero structure of $\boldsymbol{Q}$ fits into a narrow band along the diagonal. In these cases, a very stable variant, referred to as the GTH (Grassmann, Taskar, and Heyman) algorithm, may be used. In this variant, all subtraction is avoided by computing diagonal elements as the sum of off-diagonal elements. This is possible since the zero-row-sum property of an infinitesimal generator is invariant under the basic operation of Gaussian elimination, namely adding a multiple of one row into another. For an efficient implementation, the GTH variant requires convenient access to both the rows and the columns of the matrix. This is the case when a banded structure is used to store $\boldsymbol{Q}$, but is generally not the case with other compact storage procedures. When the number of states becomes large and the structure in not banded, the direct approach loses its appeal and one must resort to other methods.

Iterative Methods — For iterative methods, the first approach is to solve Eq. (3) in which $\boldsymbol{P}$ is a matrix of transitions probabilities. Let the initial probability distribution vector be given by $\boldsymbol{\pi}^{(0)}$. After the first transition, the probability vector is given by $\boldsymbol{\pi}^{(1)} = \boldsymbol{\pi}^{(0)}\boldsymbol{P}$; after k transitions it is given by $\boldsymbol{\pi}^{(k)} = \boldsymbol{\pi}^{(k-1)}\boldsymbol{P} = \boldsymbol{\pi}^{(0)}\boldsymbol{P}^{\boldsymbol{k}}$. If the Markov chain is ergodic, then $\lim_{k\to\infty}\boldsymbol{\pi}^{(k)} = \boldsymbol{\pi}$. This method of determining the stationary probability vector, by successively multiplying some initial probability distribution vector by the matrix of transition probabilities, is called the Power method. Observe that all that is required is a vector–matrix multiplication operation. This may be conveniently performed on sparse matrices that are stored in compact form. Because of its simplicity, this method is widely used, even though it often takes a very long time to converge. Its rate of convergence is a function of how close the subdominant eigenvalue of P is to its dominant unit eigenvalue. In models in which there are large differences in the magnitudes of transition rates, the subdominant eigenvalue can be pathologically close to one, so that for all intensive purposes the Power method fails to converge.

It is also possible to apply iterative equation solving techniques to the system of equations given by (2). The well-known Jacobi method is closely related to the Power method, and it also frequently takes very long to converge. A better iterative method is Gauss-Seidel. Unlike the previous two methods, in which the equations are only updated after each completed iteration, the Gauss-Seidel method uses the most recently computed values of the variables as soon as they become available and, as a result, almost always converges faster than Jacobi or the Power method. All three methods can be written so that the only numerical operation is that of forming the product of a sparse matrix and a probability vector, so all are equal from a computation per iteration point of view.

Block Methods — In Markov chain models, it is frequently the case that the state space can be meaningfully partitioned into subsets. Perhaps the states of a subset interact only infrequently with the states of other subsets, or perhaps the states possess some property that merits special consideration. In these cases, it is possible to partition the transition rate matrix accordingly and to develop iterative methods based on this partition. In general, such block iterative methods require more computation per iteration, but this is offset by a faster rate of convergence.

If the state space of the Markov chain is partitioned into N subsets of size $n_1, n_2, \ldots, n_N$ with $\sum_{i=1}^{N} n_i = n$, then block iterative methods essentially involve the solution of N systems of equations of size n_i, $i = 1,2,\ldots, N$, within a global iterative structure, such as Gauss-Seidel, for instance: thus the Block Gauss-Seidel method. Furthermore, these n systems of equations are nonhomogeneous and have nonsingular coefficient matrices and either direct or iterative methods may be used to solve them. It is not required that the same method be used to solve all the diagonal blocks, so that it is possible to tailor methods to the particular block structures.

If a direct method is used, then a decomposition of the diagonal block may be formed once and for all before initializing the global iteration process. In each subsequent global iteration, solving for that block then reduces to a forward and backward substitution operation. The nonzero structure of the blocks may be such that this is a particularly attractive approach. For example, if the diagonal blocks are themselves diagonal matrices, or if they are upper or lower triangular matrices or even tridiagonal matrices, then it is very easy to obtain their LU decomposition, and a block iterative method becomes very attractive.

If the diagonal blocks do not possess such a structure, and when they are of large dimension, it may be appropriate to use an iterative method to solve each of the block systems. In this case, there are many inner iterative methods (one per block) within an outer (or global) iteration. A number of tricks may be used to speed up this process. First, the solution computed for any block at global iteration k should be used as the initial approximation to the solution of this same block at iteration $k + 1$. Second, it is hardly worthwhile computing a highly accurate solution in early (outer) iterations. Only a small number of digits of accuracy should be required until the global process begins to converge. One convenient way to achieve this is to carry out only a fixed small number of iterations for each inner solution.

Iterative Aggregation/Disaggregation Methods — Related to block iterative methods, these methods are particularly powerful when the Markov chain is nearly completely decomposable, as the partitions are chosen based on how strongly the states of the Markov chain interact with one another.

Projection Methods — An idea that is basic to sparse-linear systems and eigenvalue problems is that of projection processes. Whereas iterative methods begin with an approximate solution vector that is modified at each iteration and which (supposedly) converges to a solution, projection methods create vector subspaces and search for the best possible approximation to the solution that can be obtained from that subspace. With a given subspace, for example, it is possible to extract a vector $\hat{\pi}$ that is a linear combination of a set of basis vector for that space and which minimizes $|\hat{\boldsymbol{\pi}}\boldsymbol{Q}|$ in some vector norm. This vector $\hat{\pi}$ may then be taken as an approximation to the solution of $\boldsymbol{\pi Q} = 0$. This is the basis for the Generalized Minimal Residual (GMRES) algorithm. Another popular projection method is the method of Arnoldi. The subspace most often used is the Krylov subspace, $K_m = \text{span}\{\boldsymbol{v}_1, \boldsymbol{v}_1\boldsymbol{Q}, \ldots, \boldsymbol{v}_1\boldsymbol{Q}^{m-1}\}$, constructed from a starting vector v_1 and successive iterates of the power method. The computed vectors are then orthogonalized with respect to one another. It is also possible to construct iterative variants of these methods. When the subspace reaches some maximum size, the best approximation is chosen from this subspace and a new subspace generated using this approximation as the initial starting point.

Preconditioning techniques are frequently used to improve the convergence rate of iterative Arnoldi and GMRES. This typically amounts to replacing the original system $\boldsymbol{\pi Q} = 0$ by $\boldsymbol{\pi Q M}^{-1} = 0$, where $\boldsymbol{M}$ is a matrix whose inverse is easy to compute. The objective of preconditioning is to modify the system of equations to obtain a coefficient matrix with a fast rate of convergence. It is worthwhile pointing out that preconditioning may also be used with the basic power method to improve its rate of convergence. The inverse of the matrix $\boldsymbol{M}$ is generally computed from an incomplete *LU* factorization of the matrix $\boldsymbol{Q}$.

Stochastic Automata Networks

Stochastic Automata Networks (SANs) provide a means of performing Markov chain modeling without the problem of having to store huge transition matrices. A SAN consists of a number of individual stochastic automata that operate more or less independently of each other. Each individual automaton is represented by a number of states and rules that govern the manner in which it moves from one state to the next. The state of an automaton at any time t is just the state it occupies at time t, and the state of the SAN at time t is given by the state of each of its constituent automata. An automaton may be thought of as a component in a Markov chain state descriptor.

The use of SANs is important in the performance modeling of parallel and distributed systems, since such systems are often viewed as collections of components that operate more or less independently, requiring only infrequent interaction such as synchronizing their actions or operating at different rates depending on the state of parts of the overall system. This is exactly the viewpoint adopted by SANs. Furthermore, the state space explosion problem associated with Markov chain models is mitigated by the fact that the state transition matrix is not stored, nor even generated. Instead, it is represented by a number of much smaller matrices, one for each of the stochastic automata that constitute the system, and from these all relevant information may be determined without explicitly forming the global matrix. A considerable saving in memory is realized by storing the matrix in this fashion.

The compact form in which the transition matrix that characterizes the model is kept (called the SAN Descriptor) is written as

$$\sum_{j=1}^{N+2E} \otimes_{i=1}^{N} \boldsymbol{Q}_j^{(i)},$$

where N is the number of automata in the SAN, E is the number of synchronizing events and $\boldsymbol{Q}_j^{(i)}$ is a square matrix of low dimension. In order to benefit from this compact form, the descriptor is never expanded into a single large matrix. Consequently, all subsequent operations must necessarily work with the model in its descriptor form, and hence, numerical operations on the underlying Markov chain infinitesimal generator become more costly. Research efforts directed at reducing these costs include the development of a generalized tensor algebra to permit functional transitions to be handled at the same low costs as constant transitions, design of algorithms to reduce the amount of computation involved in forming the product of a vector and a SAN descriptor, and finding suitable preconditioners with which to speed up iterative methods.

See

- Markov Chains
- Markov Processes
- Numerical Analysis
- Queueing Theory
- Stochastic Process

References

Berman, A., & Plemmons, R. J. (1994). *Nonnegative matrices in the mathematical sciences*. Philadelphia: SIAM.

Fernandes, P., Plateau, B., & Stewart, W. J. (1998). Efficient descriptor-vector multiplication in stochastic automata networks. *Journal of the Association for Computing Machinery, 45*, 381–414.

Saad, Y. (1996). *Iterative solution of sparse linear systems*. New York: PWS Publishing.

Stewart, W. J. (1976). MARCA: Markov chain analyzer. IEEE Computer Repository, No. R76 232 (See the URL: http://www.csc.ncsu.edu/faculty/WStewart).

Stewart, W. J. (1994). *An introduction to the numerical solution of Markov chains*. New Jersey: Princeton University Press.

Markov Chain Monte Carlo

Michel Wedel[1] and Peter Lenk[2]
[1]University of Maryland, College Park, MD, USA
[2]University of Michigan, Ann Arbor, MI, USA

Introduction

Markov chain Monte Carlo (MCMC) methods numerically approximate the integral or expectation, $E[g(Y)] = \int g(y) f(y|\Theta) dy$, where Y is a random variable with distribution $f(y|\Theta)$, which is parameterized by Θ, and $g(Y)$ is an integrable function of Y, where the integral is with respect to either Lebesgue measure for continuous random variables or counting measure for discrete ones. A simple way to compute $E[g(Y)]$ is through Monte Carlo (MC) simulation, which approximates the integral as an average of $g(Y)$ across a random sample from $f(y|\Theta)$: $\overline{g(Y)} = \frac{1}{n} \sum_{i=1}^{n} g(y_i)$. The estimation variance is proportional to n^{-1}, regardless of the dimension of Y, and the estimator can be made arbitrarily accurate by letting the size of the sample $n \to \infty$ by the strong law of large numbers. MCMC addresses settings where random variates for $f(y|\Theta)$ cannot be generated easily, e.g., through the inverse transform method, the acceptance-rejection method (also called rejection sampling), or importance sampling. These methods generally rely on independent and identically distributed (i.i.d.) random draws to approximate the integral.

MCMC methods relax this independence assumption to construct a Markov chain of draws $\{y_i, i = 1, \ldots, n\}$, with a stationary distribution equal to. $f(y|\Theta)$. MCMC uses recursive simulation where the random number generator for Y_i depends on the previous draw y_{i-1}, hence the name Markov chain Monte Carlo. MCMC's range of applications is astonishing, and continues to expand. A large part of these applications have been in Bayesian statistics, but MCMC originated in image processing and physics and continues to be used in these fields, as well as in biology, engineering, demography, finance and marketing. MCMC was started by the work of Metropolis et al. (1953) and Hastings (1970). Gibbs sampling as a special case developed through the work of Besag (1974), Geman and Geman (1984), and Gelfand and Smith (1990). Important extensions were developed by Albert and Chib (1993), Green (1995), Richardson and Green (1997) and Neal (2003). Texts include Gill (2008), Press (2003), Gelman et al. (2003), and Zellner (1971). Essential MCMC methods are reviewed here, while details can be found in the references above.

Discussion

Metropolis-Hastings Sampler: The Metropolis-Hastings (MH) sampler is very general and sparked the MCMC revolution. For $i = 1, \ldots, n$, it generates a candidate sample y_i from a proposal distribution $h(y|y_{i-1}, \Phi)$ and transforms it to make it behave as if it came from $f(y|\Theta)$ (The support of h is a subset of that of f). If the proposal distribution h depends on the previous value y_{i-1}, the algorithm is called Random Walk Metropolis-Hastings (rMH), while if it does not depend on previous values, it is called Independence Metropolis-Hastings (iMH). The algorithm works as follows, for $i = 1, 2, \ldots, n$

1. Initialize the chain at y_0 that is in the support of f.
2. Given that a prior value y_{i-1} has been obtained, sample a candidate value $y^* \sim h(y|y_{i-1};\Phi)$ and sample $u_i \sim U(0,1)$.
3. Calculate $\alpha(y_{i-1}, y^*) = \frac{f(y^*|\Theta)}{f(y_{i-1}|\Theta)} \cdot \frac{h(y_{i-1}|y^*,\Phi)}{h(y^*|y_{i-1},\Phi)}$.
4. Accept the candidate $y_i = y^*$, if $\alpha(y_{i-1}, y^*) > u_i$, otherwise set $y_i = y_{i-1}$.

The normalizing constants for f and h cancel in Step 3, so that they only need to be known up to such constants. The MH algorithm creates a Markov chain with transition function $q(y_{i-1}, y_i) = h(y_i|y_{i-1},\Phi)\tau(y_{i-1}, y_i)$ where $\tau(y_{i-1}, y^*) = \min[\alpha(y_{i-1}, y^*), 1]$ is the acceptance probability from Step 4. The chain is reversible because $f(y_{i-1}|\Theta)q(y_{i-1}, y_i) = f(y_i|\Theta)q(y_i, y_{i-1})$, and is therefore ergodic with stationary distribution f. This has the crucial implication that regardless of the initial value in Step 1, the draws from the Markov chain will eventually be from f. Monte Carlo is a special case if the candidate distribution h is equal to f: then $\alpha(y_{i-1}, y^*) = 1$. If $h(y|y_{i-1},\Phi)$ is symmetric in $(y - y_{i-1})$, e.g. a normal distribution with mean y_{i-1}, then the ratio in h cancels in Step 3.

The performance of MH depends on the proposal distribution. In rMH if the proposal distribution is too tight around the last value of the chain, then the candidate is highly likely to be accepted, and the Markov chain will tour the support of f very slowly, so n will have to be quite large to obtain reliable MCMC estimates. Conversely, if the variance of the proposals is too large, the MCMC algorithm will reject most of the candidate values, and the chain will hardly budge. For the estimator to be valid, the chain needs to visit areas of the support of f with non-negligible probabilities.

Convergence: Starting from an arbitrary y_0, the chain passes through a transitory period, say $i = 1, \ldots, l$ for $l < n$, where the draws are not from f. These initial draws are not used in the MCMC approximation of $E[g(Y)]$: $\overline{g(Y)} = \frac{1}{n-l} \sum_{i=l+1}^{n} g(y_i)$. In theory, under very general conditions the rate of convergence is geometric in the second eigen value of the transition function. Problems can occur if the target distribution is multimodal, and f is zero between modes, so that subsets of the support do not communicate with each other. Then the chain can become stuck in isolated regions of the support unless the proposal distribution h is sufficiently broad to bridge the gaps. In practice, it may be difficult to conclusively determine l. One procedure for monitoring convergence is to run multiple chains from different initial values and to compute multiple estimates. If the between-chain variance of the estimators is small relative to the within-chain variance, then the chain has likely converged. A host of other diagnostic measures are available, that may help identify likely convergence of the chain.

Blocked MH Sampler: Depending on the structure of f, it may be convenient to block Y into sub-vectors Y_s for $s = 1, 2, \ldots, S$. The distribution of each sub-vector is conditioned on all others to obtain the full conditional distributions: $f(y_s|y_{-s};\Theta)$, with y_{-s} denoting y with y_s omitted $h_s(y_s|y_{-s};\Phi)$ is the proposal distribution for Y_s. This leads to the following algorithm for $i = 1, 2, \ldots, n$ and $s = 1, 2, \ldots, S$:

1. Initialize y_0 in the support of f.
2. Sample a candidate value $y_s^* \sim h_s\left(y_s|y_{-s,i-1};\Phi\right)$ and $u_i \sim U(0,1)$.
3. Calculate $\alpha_{s,i} = \frac{f(y^*|\Theta)}{f(y_{i-1}|\Theta)} \cdot \frac{h_s(y_{s,i-1}|y^*;\Phi)}{h_s(y_s^*|y_{s,i-1};\Phi)}$, where y^* is identical to y_{i-1}, except for sub-vector s, which equals y_s^*.
4. Accept the candidate $y_{s,i} = y_s^*$, if $\alpha_{s,i} > u_i$, otherwise keep $y_{s,i-1}$.

This algorithm cycles through the s sub-vectors (in arbitrary order, systematically or randomly) and updates them separately though MH-steps. Not every sub-vector needs to be updated at every iteration i.

Gibbs Sampler: In many applications, some or even all of the full conditional target distributions $f(y_s|y_{-s};\Theta)$ can be sampled directly, which greatly simplifies the Blocked MH algorithm. This can be seen by substituting the full conditional distributions for the proposal distributions in Step 3 of the Blocked MH: $\alpha_{s,i} = \frac{f(y^*|\Theta)}{f(y_{i-1}|\Theta)} \cdot \frac{f(y_{s,i-1}|y^*;\Theta)}{f(y_s^*|y_{s,i-1};\Theta)}$. Because $f(y|\Theta) = f(y_s, y_{-s}|\Theta)$, and $y_{-s}^* = y_{-s,i-1}$, it holds that $\alpha_{s,i} = \frac{f\left(y_{-s,i-1}|\Theta\right)}{f\left(y_{-s,i-1}|\Theta\right)} = 1$. The algorithm for the Gibbs sampler modifies the Blocked MH by replacing Step 2 with directly drawing $y_{s,i} \sim f\left(y_{s,i}|y_{-s,i};\Theta\right)$ and skipping Steps 3 and 4 for these blocks.

Modifications of the Gibbs sampler have been proposed to speed up convergence and provide the chains with better properties. For three sub-vectors y_1, y_2, y_3, for example, the Collapsed Gibbs Sampler

draws from the unconditional joint distribution $y_{1:2,i} \sim f(y_1, y_2|\Phi)$, and the full conditional distribution $y_{3,i} \sim f(y_3|y_{1,i}, y_{2,i}; \Phi)$. The Grouped Gibbs Sampler on the other hand, groups two sub-vectors and draws from the full conditionals $y_{1:2,i} \sim f\left(y_1, y_2|y_{3,i-1}; \Phi\right)$ and $y_{3,i} \sim f(y_3|y_{1,i}, y_{2,i}; \Phi)$. The relative simplicity of the Gibbs sampling algorithms has contributed to their popularity, and many extensions, three important ones being the Auxiliary Variable, Slice and Reversible Jump Samplers.

Auxiliary Variable Sampler: Introducing an auxiliary random variable Z can simplify MCMC if there is a joint distribution $h(y, z|\theta, \Psi)$ such that $f(y|\Theta) = \int h(y, z|\theta, \Psi)dz$, and both $h(y|z, \Theta, \Psi)$ and $h(z|y, \Theta, \Psi)$ are easy to sample. Using these two full conditional distributions, it is then straightforward to sample from $h(y, z|\theta, \Psi)$, using Gibbs sampling. The Auxiliary Variable Gibbs Sampler is then, for $i = 1, 2, \ldots, n$:

1. Sample $y_i \sim h(y|z_{i-1}, \Theta, \Psi)$.
2. Sample $z_i \sim h(z|y_i, \Theta, \Psi)$.

An additional advantage is that the introduction of the augmented variable helps mixing.

Slice Sampler: A special case of the Auxiliary Variable Sampler arises if $f(y|\Theta)$ an be factored as $f(y|\Theta) \propto k(y|\Theta) \cdot h(y|\Theta)$. The auxiliary variable Z in this case is chosen such that the joint density $f(y, z) \propto I[0 < z < k(y|\Theta)] \cdot h(y|\Theta)$. The resulting sampler is called the Slice Sampler and iterates between the following full conditional distributions, for $i = 1, 2, \ldots, n$:

1. Sample $z_i \sim U(0, k(y_{i-1}|\Theta))$, from a uniform distribution on 0 and $k(y_{i-1}|\Theta)$.
2. Sample $y_i \sim h(y|\Theta)I[0 < z_i < k(y|\Theta)]$, from the distribution $h(y|\Theta)$ truncated on the set $\{y : z_i < k(y|\Theta)\}$.

Slice sampling is applicable in cases where $k^{-1}(y|\Theta)$ can be analytically obtained, and the truncated distribution $h(y|\Theta)I[0 < z_i < k(y|\Theta)]$ can be sampled from, often by using the inverse transform method. The extension to distributions that factor as $f(y|\Theta) \propto h(y|\Theta) \cdot \prod_t k_t(y|\Theta)$ is straightforward if all $k_t^{-1}(y|\Theta)$ an be obtained, now by sampling multiple $z_{i,t} \sim U(0, k_t(y_{i-1}|\Theta))$.

Reversible Jump Sampler: The above algorithms assume that the dimension of Y is constant. The Reversible Jump (RJ) sampler is an extension of MH that constructs a Markov chain that transverses spaces of different dimensions. The spaces are labeled m, and $Y^{(m)}$ is the random variable Y restricted to space m. The dimension of $Y^{(m)}$ or $\dim\left(Y^{(m)}\right)$ depends on m (In Bayesian statistics – details below – RJ is used to transverse different models where m indicates the model, and then simulate $Y^{(m)}$ given model m). The state space for the Markov chain is $\left(M, Y^{(M)}\right)$ with joint distribution $f\left(m, y^{(m)}|\Theta\right) = f\left(y^{(m)}|m, \Theta_y\right)f(m|\Theta_m)$ where $P(M = m) = f(m|\Theta_m)$ is a discrete distribution, and $f\left(y^{(m)}|m, \Theta_y\right)$ is the distribution of Y restricted to space m. RJ is a strategy to simulate $\left(M, Y^{(M)}\right)$ when a convenient random number generator for $f\left(m, y^{(m)}|\Theta\right)$ does not exist.

As with MH, the goal is to construct a reversible Markov chain with stationary distribution $f\left(m, y^{(m)}|\Theta\right)$. Reversible moves between any $\left(m, y^{(m)}\right)$ and $\left(m', y^{(m')}\right)$ require a bijective mapping, which does not exist when the spaces have different dimensions. The trick is to augment $y^{(m)}$ with a random variable $u^{(m)}$ so that $\dim\left(y^{(m)}\right) + \dim\left(u^{(m)}\right)$ is constant across all m: $\dim\left(y^{(m)}\right) + \dim\left(u^{(m)}\right) = \dim\left(y^{(m')}\right) + \dim\left(u^{(m')}\right)$. RJ requires a bijective, differentiable function $\left(y^{(m')}, u^{(m')}\right) = T_{m,m'}\left(y^{(m)}, u^{(m)}\right)$ that uniquely maps $\left(y^{(m)}, u^{(m)}\right)$ to $\left(y^{(m')}, u^{(m')}\right)$ with reverse mapping $T_{m',m} = T_{m,m'}^{-1}$. Given the current state $\left(m, y^{(m)}\right)$ of the Markov chain, candidate values are generated by: (1) selecting a new value m' according to the proposal distribution $q\left(m'|m, \Psi\right)$; (2) generating $u^{(m)}$ from $h_{m,m'}\left(u^{(m)}|y^{(m)}, \Phi\right)$; and (3) computing the candidate $\left(y^{(m')}, u^{(m')}\right) = T_{m,m'}\left(y^{(m)}, u^{(m)}\right)$. For the Markov chain to be reversible, the implied distribution of $u^{(m')}$, $h_{m',m}\left(u^{(m')}|y^{(m')}, \Phi\right)$, is required to move from $\left(y^{(m')}, u^{(m')}\right)$ to $\left(y^{(m)}, u^{(m)}\right)$ using the reverse mapping $T_{m',m}$. Implementation details of the RJ are as much art as science, because the construction of $\left\{T_{m,m'}\right\}$ for all m and m' and the selection of proposal distributions are tailored specifically for each application. The RJ algorithm for $i = 1, 2, \ldots, n$ is:

1. Initialize the chain at $\left(m_0, y_0^{m_0}\right)$ in the support of f.
2. Given m_{i-1} and $y_{i-1}^{(m_{i-1})}$ are obtained, set $m = m_{i-1}$ and $y = y_{i-1}^{(m_{i-1})}$ and
   a. Sample $m' \sim q(m'|m, \Psi)$;
   b. Sample $u \equiv u^{(m)} \sim h_{m,m'}\left(u^{(m)}|y, \Phi\right)$;
   c. Compute proposal $y' \equiv y^{(m')}$ from $(y', u') = T_{m,m'}(y, u)$.

3. Calculate
$$\alpha(y,y') = \frac{f\left(m',y'|\Theta\right)}{f(m,y|\Theta)} \cdot \frac{h_{m',m}\left(u'|y',\Phi\right)}{h_{m,m'}(u|y,\Phi)} \cdot \frac{q(m|m',\Psi)}{q(m'|m,\Psi)} \cdot \left|\frac{\partial T_{m,m'}(y,u)}{\partial y \partial u}\right|.$$
4. Sample $v_i \sim U(0,1)$ and accept the candidate $\left(m_i, y_i^{(m_i)}\right) = (m', y')$ if $\alpha(y,y') > v_i$, otherwise set $\left(m_i, y_i^{(m_i)}\right) = \left(m_{i-1}, y_{i-1}^{(m_{i-1})}\right)$.

In step 3, $\left|\frac{\partial T_{m,m'}(y,u)}{\partial y \partial u}\right|$ is the Jacobian of the transformation $T_{m,m'}$, which is needed because it is a deterministic function for the change in variables from $\left(y^{(m)}, u^{(m)}\right)$ to $\left(y^{(m')}, u^{(m')}\right)$. As in the MH algorithm, the distributions $f\left(m, y^{(m)}|\Theta\right)$, $q\left(m'|m,\Psi\right)$, and $h_{m,m'}\left(u^{(m)}|y,\Phi\right)$ only need to be known up to normalizing constants which cancel in step 3. It should be noted that while $\alpha(y,y')$ in its general form provided in step 3 is somewhat complex, in a wide range of practical applications it simplifies considerably, for example when the proposal distributions are symmetric (see above), when $\dim\left(y^{(m')}\right) > \dim\left(y^{(m)}\right)$, in which case the mapping reduces to $\left(y^{(m')}\right) = T\left(y^{(m)}, u^{(m)}\right)$, and when moves are limited to $m' \in \{(m_{i-1}-1), m_{i-1}, (m_{i-1}+1)\}$.

***Example*:** In Bayesian statistics the parameters of a model are considered random variables, reflecting a priori uncertainty on the part of the researcher that is reduced a posteriori after the data are observed. Inference focuses on their posterior distribution, which summarizes all information about the parameters. According to Bayes Theorem, the posterior distribution is proportional to the prior distribution of the parameters times the distribution of the data given the parameters. Bayesian estimation and inference has gained great popularity in business, in particular in marketing and finance, because even without strictly accepting the (attractive) fundamental properties of Bayesian inference, pragmatic Bayesians have found great value in MCMC algorithms to estimate complex models, especially as uninformative prior distributions can be used. Simpler illustrative examples follow.

***Example 1*:** The Weibull distribution is used in duration analysis applications to bankruptcy in finance, and in customer relationship management (CRM) in marketing. The observations $\{x_j\}$ for $j = 1,\ldots,J$ are a random sample of durations from a Weibull distribution: $f(x|\theta,\delta) = \theta\delta x^{\delta-1}\exp(-\theta x^{\delta})$ for $x > 0$. The prior distributions of the parameters are Gamma distributions: $p(\theta) = \frac{s_0^{r_0}}{\Gamma(r_0)}\theta^{r_0-1}\exp(-s_0\theta)$ and $p(\delta) = \frac{a_0^{b_0}}{\Gamma(b_0)}\delta^{a_0-1}\exp(-b_0\delta)$. The joint posterior distribution of the parameters is: $\pi\left(\theta,\delta|\{x_j\}\right) \propto p(\theta)p(\delta)\prod_{j=1}^{J} f\left(x_j|\theta,\delta\right)$, which does not have a convenient random number generator and can be sampled with MH within Gibbs. The full conditional distribution of θ given the data and δ_{i-1} is also a Gamma distribution: $\pi\left(\theta|\delta_{i-1},\{x_j\}\right) \propto \theta^{r_0+n-1}\exp\left(-\theta\left[s_0 + \sum_{j=1}^{J} x_j^{\delta_{i-1}}\right]\right)$. The full conditional distribution of δ given the data and θ_i does not have a known distributional form: $\pi\left(\delta|\theta_i,\{x_j\}\right) \propto \delta^{a_0+n-1}\left[\prod_{j=1}^{J} x_j^{\delta-1}\right]\exp\left(-b_0\delta - \theta_i\sum_{j=1}^{J} x_j^{\delta}\right)$. Thus, rMN can be used to generate the candidate δ^*. The MCMC algorithm to approximate the posterior distribution of the parameters is, for $i = 1,2,\ldots,n$:

1. Initialize the chain at (θ_0,δ_0).
2. Draw θ_i from a Gamma distribution $\theta_i = G\left(r_0 + n, s_0 + \sum_{j=1}^{J} x_j^{\delta_{i-1}}\right)$.
3. Sample $u_i \sim U(0,1)$, and generate a candidate δ^* from a log-normal distribution: $g(\delta^*|\delta_{i-1},\sigma) \propto \frac{1}{\delta^*}\exp\left[-\frac{1}{2\sigma^2}\left(\ln(\delta^*) - \ln(\delta_{i-1})\right)^2\right]$.
4. Compute $\alpha(\delta_{i-1},\delta^*) = \frac{\pi\left(\delta^*|\theta_i,\{x_j\}\right)\delta_{i-1}}{\pi\left(\delta_i|\theta_i,\{x_j\}\right)\delta^*}$.
5. Accept $\delta_i = \delta^*$ if $\alpha(\delta_{i-1},\delta^*) > u_i$, otherwise set $\delta_{i-1} = \delta_{i-1}$.

Extensions involve the parameterization of θ in terms of predictor variables $\theta_j = w_j\beta$, and the case where the durations are censored by the observation time; the estimations of the models in question involve extensions of the algorithms above.

***Example 2*:** Change-point regression models are popular in finance to describe financial time series data with a structural change, and used in marketing in models of stochastic preference and market shares. Here, the data $\{x_t\}$ are observed for time points $t = 1,\ldots,T$, and assumed to follow a binomial distribution: $f(x_t|\pi_t) = \pi_t^{x_t}(1-\pi_t)^{1-x_t}$; for $x_t \in \{0,1\}$. Two regression functions are

separated in time by an unknown switch-point $\tau : \pi_t = \Phi\left(w_t^{'}\beta_k\right)$, with $\beta_k = \beta_1$ for $t \leq \tau$, and $\beta_k = \beta_2$ for $t > \tau$. Φ is the Normal CDF used as an inverse link function, w_t is a vector of regressors, and $\beta_k \sim N(b_0, B_0)$ the prior distributions of its coefficients. The 'switch-point' has a uniform discrete prior on a subset of the observed timepoints: $\tau \sim U(c, d)$. The MCMC algorithm simplifies through the introduction of an auxiliary variable $z_t \sim N\left(w_t^{'}\beta_k, 1\right)$, with, $x_t = I(z_t > 0)$, and $I(\cdot)$ the indicator function. The MCMC algorithm to approximate the posterior distribution of the parameters is, for $i = 1, 2, \ldots, n$:

1. Sample for: $k = 1, 2:\ \beta_{k,i} \sim N(b_{k,i}, B_{k,i})$, with $b_{k,i} = B_{k,i}\left(B_0^{-1}b_0 + \sum_{t=L_{k,i}}^{t=U_{k,i}} w_t^{'}z_{t,i}\right)$, $B_{k,i} = \left(B_0^{-1} + \sum_{t=L_{k,i}}^{t=U_{k,i}} w_t w_t^{'}\right)$, and $L_{k,i} = 1 + (k-1)\tau_i$, $U_{k,i} = \tau + (k-1)(T - \tau_i)$.
2. Sample, for $L_{k,i} < t < U_{k,i}$: $z_{t,i} \sim N\left(w_t^{'}\beta_{k,i}, 1\right)I(z_{t,i} < 0)$ if $x_t = 0$, and $z_{t,i} \sim N\left(w_t^{'}\beta_{k,i}, 1\right)I(z_{t,i} > 0)$ if $x_t = 1$.
3. Sample τ_i using $Pr(\tau_i = r) = \frac{\prod_{t\leq r} f\left(x_t|w_t^{'}\beta_{1,i}\right)\prod_{t>r} f\left(x_t|w_t^{'}\beta_{2,i}\right)}{\sum_{s=c}^{d}\prod_{t\leq s} f\left(x_t|w_t^{'}\beta_{1,i}\right)\prod_{t>s} f\left(x_t|w_t^{'}\beta_{2,i}\right)}$.

Extensions of this MCMC procedure for multiples witch points are available, and extensions to an unknown number of switch points require RJMCMC.

***Example 3*:** Mixture models are used in finance to describe financial returns during different economic regimes, and are popular in marketing to identify unobserved heterogeneity in response-based market segmentation. The data $\{x_j\}$ are observed for individuals $j = 1, \ldots, J$, and assumed to follow a mixture Normal distribution with K classes and probabilities δ_k for which $0 < \delta_k < 1$ and $\sum_{k=1}^{m} \delta_k = 1$. Thus, $x_j \sim \sum_{k=1}^{m} \delta_k N\left(w_j^{'}\beta_k, \sigma_k^2\right)$. Here, β_k are class-specific regression coefficients associated with the vector of regressors w_t, with prior distributions $\beta_k \sim N(b_0, B_0)$. Further Inverse Gamma and Dirichlet priors are specified for: $\sigma_k^2 \sim IG\left(\frac{a_0}{2}, \frac{A_0}{2}\right)$and $\delta_{1:m} \sim D(c_0, \ldots, c_0)$. The MCMC algorithm simplifies by introducing an auxiliary variable with a multinomial prior distribution: $z_j \sim M(\delta_{1:m})$ that indicates the membership of individual j in class k, that is $z_j = 1, \ldots, m$. The MCMC algorithm is, for $i = 1, 2, \ldots, n$:

1. Sample, for $k = 1, \ldots, m : \beta_{k,i} \sim N(b_{k,i}, B_{k,i})$, with $b_{k,i} = B_{k,i}\left(B_0^{-1}b_0 + \sum_{\{j:z_j=k\}} w_j^{'}x_j\right)$, and $B_{k,i} = \left(B_0^{-1} + \sum_{\{j:z_j=k\}} w_j w_j^{'}\right)$.
2. Sample, for $k = 1, \ldots, m :$ $\sigma_{k,i}^2 \sim IG\left(\frac{a_0+n_k}{2}, \frac{A_0+\sum_{\{j:z_j=k\}}\left(x_j - w_j^{'}\beta_{k,i}\right)^2}{2}\right)$, with $n_k = \sum_{\{j:z_j=k\}} 1$.
3. Sample $\delta_{1:m} \sim D(c_0 + n_1, \ldots, c_0 + n_m)$.
4. Sample z_j using $Pr(z_j = k) = \frac{\delta_k f\left(x_j|w_t^{'}\beta_{k,i}, \sigma_{k,i}^2\right)}{\sum_s \delta_s f\left(x_j|w_t^{'}\beta_{s,i}, \sigma_{s,i}^2\right)}$.

This sampler, like that for many mixture models, suffers from "label switching," a problem in which the class parameters switch across the class labels during the iterations. Several solutions are available, including ordering the mixture probabilities or post-processing of the draws.

Furthermore, the above algorithm can be extended to include the number of classes $m = 1, \ldots, m_{\max}$, using RJMCMC. A step is added to the algorithm in which two randomly chosen classes (k_1 and k_2) are merged (k_*), or one randomly chosen class is split. A splitting decision is usually made with probability $\eta_m = 0.5$, a merging decision with $(1 - \eta_m)$, for, $m = 2, \ldots, (m_{\max} - 1)$, and $\eta_1 = 0$ and $\eta_{m_{\max}} = 1$. The merge move involves matching of moments of the class-distributions, involving the computation of β_{k_*} such that the mean $\mu_{k_*} = w^{'}\beta_{k_*}$ of the new class matches that of k_1 and k_2, as does the variance $\sigma_{k_*}^2$:

M1. Randomly select $k_1 \propto 1/m$ and find k_2 'most similar' to k_1.
M2. Compute $\delta_{k_*} = \delta_{k_1} + \delta_{k_2}$.
M3. Match $\mu_{k_*} = \frac{\delta_{k_1}}{\delta_{k_*}}\mu_{k_1} + \frac{\delta_{k_2}}{\delta_{k_*}}\mu_{k_2}$.
M4. Compute $\sigma_{k_*}^2 = \frac{\delta_{k_1}}{\delta_{k_*}}\left(\mu_{k_1}^2 + \sigma_{k_1}^2\right) + \frac{\delta_{k_2}}{\delta_{k_*}}\left(\mu_{k_2}^2 + \sigma_{k_2}^2\right) - \mu_{k_*}^2$.
M5. Recompute z_j using step 4 above.

The split move operates as follows, and again involves matching of the first two moments of the class-distributions, of the old and new classes:

S1. Randomly select $k_* \propto 1/m$, and draw the auxiliary variables $u_{1:3} \sim Beta(a, b)$.
S2. Compute $\delta_{k_1} = u_1\delta_{k_*}$, and $\delta_{k_2} = (1 - u_1)\delta_{k_*}$.
S3. Match $\mu_{k_1} = \mu_{k_*} - u_2\sigma_{k_*}\sqrt{\frac{\delta_{k_2}}{\delta_{k_1}}}$, and $\mu_{k_2} = \mu_{k_*} + u_2\sigma_{k_*}\sqrt{\frac{\delta_{k_1}}{\delta_{k_2}}}$.

S4. Compute $\sigma_{k_1}^2 = u_3\left(1-u_2^2\right)\sigma_{k_*}^2 \frac{\delta_{k*}}{\delta_{k_1}}$, and $\sigma_{k_2}^2 = (1-u_3)\left(1-u_2^2\right)\sigma_{k_*}^2 \frac{\delta_{k*}}{\delta_{k_2}}$.

S5. Recompute z_j using step 4 above.

The split/merge proposal is accepted with probability $\min(\alpha(y,y'),1)$, computed as outlined in the RJ algorithm above (and the split is rejected if k_2 is not 'most similar' to k_1 to ensure reversibility). Here $h_m\left(u^{(m)}|y,\Phi\right) = Beta\,(a,b)$, and $q\left(m'|m,\Psi\right) = P\left(m'|m_{i-1}\right) = 0.5$ in the RJ algorithm described above. The split/merge moves are reversible, as $T_{m,m'}$ is defined in S2-S4, and $T_{m,m'}^{-1}$ in M2-M3. The split/merge moves may be combined with "birth/death" moves, randomly chosen with probabilities 0.5/0.5. In a birth move the parameters of a new class are drawn at random from proposal distributions on the appropriate support (e.g., $\delta_{k_*} \sim Beta, \beta_k \sim MVN, \sigma_k^{-2} \sim Gamma$), and the weights are rescaled so that they sum to one. In a death move an empty class is deleted, and the remaining weights are rescaled (Richardson and Green 1997).

See

- Acceptance-Rejection Method
- Importance Sampling
- Inverse Transform Method
- Markov Chains
- Monte Carlo Simulation
- Reversible Markov Chain/Process
- Simulation of Stochastic Discrete-Event Systems
- Simulation Optimization
- Variance Reduction Techniques in Monte Carlo Methods

References

Albert, J. H., & Chib, S. (1993). Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association, 88*(422), 669–679.

Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems (with discussion). *Journal of the Royal Statistical Society, Series B, 41*, 143–168.

Gelfand, A. E., & Smith, A. F. M. (1990). Sampling based approaches to calculating marginal densities. *Journal of the American Statistical Association, 85*, 398–409.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2003). *Bayesian data analysis*. London: Chapman and Hall.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 6*, 721–741.

Gill, J. (2008). *Bayesian methods: A social and behavioral sciences approach*. New York: Chapman &Hall.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika, 82*(4), 711–732.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains, and their applications. *Biometrika, 57*, 97–109.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics, 21*, 1087–1091.

Neal, R. M. (2003). Slice sampling. *Annals of Statistics, 31*(3), 705–767.

Press, S. J. (2003). *Subjective and objective Bayesian statistics* (2nd ed.). New York: Wiley.

Richardson, S., & Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society, Series B, 59*, 731–792.

Zellner, A. (1971). *An introduction to Bayesian inference in econometrics*. New York: Wiley.

Markov Chains

Carl M. Harris
George Mason University, Fairfax, VA, USA

Introduction

A Markov chain is a Markov process $\{X(t), t \in T\}$ whose state space S is discrete, while its time domain T may be either continuous or discrete. Only considered here is the countable state-space problem. Classic texts treating Markov chains include Breiman (1986), Çinlar (1975), Chung (1967), Feller (1968), Heyman and Sobel (2004), Isaacson and Madsen (1976), Iosifescu (1980), Karlin and Taylor (1975), Kemeny and Snell (1976), Kemeny, Snell and Knapp (1976), and Meyn and Tweedie (2009).

As a stochastic process of the Markov type, chains possess the Markov or lack-of-memory (memoryless) property, which means that the probabilities of future events are completely determined by the present state of the process and the probabilities of its behavior from the present point on. In other words, the past behavior of the process provides no additional information in

determining the probabilities of future events if the current state of the process is known. Thus, the discrete process $\{X(t), t \in T\}$ is a Markov chain if, for any $n > 0$, any $t_1 < t_2 < \ldots < t_n < t_{n+1}$ in the time domain T, any states $i_1, i_2, \ldots, i_n$ and any state j in the state space S,

$$\Pr\{X(t_{n+1}) = j | X(t_1) = i_1, \ldots, X(t_n) = i_n\} = \Pr\{X(t_{n+1}) = j | X(t_n) = i_n\}.$$

The conditional transition probabilities on the right-hand side of this equation can be simplified by mapping the n time points directly into the nonnegative integers and renaming state i_n as i. Then the probabilities are only a function of the pair (i, j) and the transition number n. Oftentimes, it is assumed that the transition probabilities are stationary, i.e., time invariant, resulting in a square (possibly infinite) matrix $\boldsymbol{P} = [p_{ij}]$ (viz., the single-step transition matrix), which gives all conditional probabilities of moving to state j in a transition, given that the chain is currently in state i. (Any matrix with the property that its rows are nonnegative numbers summing to one is called a stochastic matrix, whether or not it is associated with a particular Markov chain).

Examples of Markov Chains

1. *Random Walk*. In its simplest form, an object moves to the left one space at each transition time with probability p or to the right with probability $1 - p$. The problem can be kept finite by requiring reflecting barriers at fixed left-and right-hand points, say M and N, such that the transition probabilities send the chain back to states $M + 1$ and $N - 1$, respectively, whenever it reaches M or N. One important variation on this problem allows the object to stay put with non-zero probability.
2. *Gambler's Ruin*. A gambler makes repeated independent bets and wins \$1 on each bet with probability p or loses \$1 with probability $1 - p$. The gambler starts with an initial stake and will play repeatedly until all money is lost or until the fortune increases to \$$M$. Let X_n equal the gambler's wealth after n plays. The stochastic process $\{X_n, n = 0, 1, 2, \ldots\}$ is a Markov chain with state space $\{0, 1, 2, \ldots, M\}$. The Markov property follows from the assumption that outcomes of successive bets are independent events. The Markov model can be used to derive performance measures of interest for this situation, such as the probability of losing all the money, the probability of reaching the goal of \$$M$, and the expected number of bets before the game terminates. All these performance measures are functions of the gambler's initial state x_0, probability p and goal \$$M$. (The gambler's fortune is thus a random walk with absorbing boundaries 0 and M). The gambler's ruin problem is a simplification of more complex systems that experience random rewards, risk, and possible ruin, such as insurance companies.
3. *Coin Toss Sequence*. Consider a series of independent tosses of a fair coin. One Markov chain is obtained by associating state 1, 2, 3 or 4 at time n depending on whether the outcomes of tosses $n - 1$ and n are (H,H), (H,T), (T,H) or (T,T), respectively.

Define the n-step transition probability $p_{ij}^{(n)}$ as the probability that the chain moves from state i to state j in n steps, and write

$$P_{ij}^{(n)} = \Pr\{X_{m+n} = j | X_m = i\} \quad \text{for all } m \geq 0 \;\; n > 0.$$

Then it follows that the n-step transition probabilities can be computed using the Chapman-Kolmogorov equations

$$P_{ij}^{(n+m)} = \sum_{k=0}^{\infty} p_{ik}^{(n)} \, p_{kj}^{(m)} \text{ for all } n, m, i, \; j \geq 0.$$

In particular, for $m = 0$,

$$P_{ij}^{(n)} = \sum_{k=0}^{\infty} p_{ik}^{(n-1)} p_{kj} = \sum_{k=0}^{\infty} p_{ik} p_{kj}^{(n-1)}, \qquad n = 2, 3, \ldots; i, j \geq 0.$$

Denoting the matrix of n-step probabilities by $\boldsymbol{P}^{(n)}$, it follows that $\boldsymbol{P}^{(n)} = \boldsymbol{P}^{(n-k)} \boldsymbol{P}^{(k)} = \boldsymbol{P}^{(n-1)} \boldsymbol{P}$ and that $\boldsymbol{P}^{(n)}$ can be calculated as the nth power of the original single-step transition matrix $\boldsymbol{P}$.

To calculate the unconditional distribution of the state at time n requires specifying the initial probability distribution of the state, namely, $\Pr\{X_0 = i\} = p_i, i \geq 0$. Then the unconditional distribution of X_n is given by

$$\Pr\{X_n = j\} = \sum_{i=0}^{\infty} \Pr\{X_n = j | X_0 = i\} \Pr\{X_0 = i\}$$
$$= \sum_{i=0}^{\infty} p_i p_{ij}^{(n)}$$

which is equivalent to multiplying the row vector $\boldsymbol{p}$ by the jth column of $\boldsymbol{P}$.

Properties of a Chain

The ultimate long-run behavior of a chain is fully determined by the location and relative size of the entries in the single-step transition matrix. These probabilities determine which states can be reached from which other ones and how long it takes on average to make those transitions. More formally, state j is said to be reachable from state i, written $i \to j$, if it is possible for the chain to proceed from i to j in a finite number of transitions, i.e., if $p^{(n)} > 0$ for some $n \geq 0$. If, in addition, $j \to i$, then the two states are said to communicate with each other, written as $i \leftrightarrow j$. If every state is reachable from every other state in the chain, the chain is said to be irreducible, i.e., the chain is not reducible into subclasses of states that do not communicate with each other.

Furthermore, the period of state i is defined as the greatest common divisor, $d(i)$, of the set of positive integers n such that $p_{ii}^{(n)} > 0$ (with $d(i) \equiv 0$ when $p_{ii}^{(n)} = 0$ for all $n \geq 1$). If $d(i) = 1$, then i is said to be aperiodic; otherwise, it is periodic with period $d(i)$. Clearly, any state with $p_{ii}^{(n)} \geq 0$ is an aperiodic state. All states in a single communicating class must have the same period, and the full Markov chain is said to be aperiodic if all of its states have period 1.

For each pair of states (i, j) of a Markov chain, define $f_{ij}^{(n)}$ as the probability that a first return from i to j occurs in n transitions and f_{ij} as the probability of ever returning to j from i. If $f_{ij} = 1$, the expectation m_{ij} of this distribution is called the mean first passage time from i to j. When $j = i$, write the respective probabilities as $f_i^{(n)}$ and f_i, and the expectation as m_i, which is called the mean recurrence time of i. If $f_i = 1$ and $m_i < \infty$, state i is said to be positive recurrent or nonnull recurrent; if $f_i = 1$ and $m_i = \infty$, state i is said to be null recurrent; if $f_i < 1$, state i is said to be transient.

A major result that follows from the above is that if $i \leftrightarrow j$ and i is recurrent, then so is j. Furthermore, if the chain is finite, then all states cannot be transient and at least one must be recurrent; if all the states in the finite chain are recurrent, then they are all positive recurrent. More generally, all the states of an irreducible chain are either positive recurrent, null recurrent, or transient.

Example: Reflecting Random Walk

Consider such a chain with movement between its four states governed by the single-step transition matrix

$$\boldsymbol{P} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1/3 & 1/3 & 1/3 & 0 \\ 0 & 2/3 & 0 & 1/3 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (1)$$

All the states communicate since there exists a path with non-zero probability from state 1 back to state 1 hitting all the other states in the interim. All the states are recurrent and aperiodic, as well.

If the random walk were infinite instead and without reflecting barriers (on either side), then the chain would be recurrent if and only if it is equally probable to go from right to left from each state; for otherwise the system would drift to $+\infty$ or $-\infty$ without returning to any finite starting point.

Limiting Behavior

The major characterizations of the stochastic behavior of a chain are typically stated in terms of its long-run or limiting behavior. Define the probability that the chain is in state j at the nth transition as $\pi_j^{(n)}$, with the initial distribution written as $\pi_j^{(0)}$. A discrete Markov chain is said to have a stationary distribution $\boldsymbol{\pi} = (\boldsymbol{\pi_0}, \boldsymbol{\pi_1}, \ldots)$ if these (legitimate) probabilities satisfy the vector–matrix equation $\boldsymbol{\pi} = \boldsymbol{\pi P}$. When written out in simultaneous equation form, the problem is equivalent to solving

$$\pi_j = \sum_i \pi_i p_{ij}, \quad j = 0, 1, 2, \ldots, \text{ with}$$
$$\sum_i \pi_i = 1.$$

The chain is said to have a long-run, limiting, equilibrium, or steady-state probability distribution $\boldsymbol{\pi} = (\boldsymbol{\pi_0}, \boldsymbol{\pi_1}, \ldots)$ if

$$\lim_{n\to\infty} \pi_j^{(n)} = \lim_{n\to\infty} \Pr\{X_n = j\} = \pi_j, \;\; j = 0, 1, 2, \ldots$$

A Markov chain that is irreducible, aperiodic and positive recurrent is said to be ergodic, and the following theorem relates these properties to the existence of stationary and/or limiting distributions.

Theorem: If $\{X_n\}$ is an irreducible, aperiodic, time-homogeneous Markov chain, then limiting probabilities

$$\pi_j = \lim_{n\to\infty} \Pr\{X_n = j\}, \;\; j = 0, 1, 2, \ldots$$

always exist and are independent of the initial state probability distribution. If all the states are either null recurrent or transient, then $\pi_j = 0$ for all j and no stationary distribution exits; if all the states are instead positive recurrent (thus the chain is ergodic), then $\pi_j > 0$ for all j the set $\{\pi_j\}$ also forms a stationary distribution, with $\pi_j = 1/m_j$.

It is important to observe that the existence of a stationary distribution does not imply that a limiting distribution exists. An example is the simple Markov chain

$$\boldsymbol{P} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

For this chain, it is easy to show that the vector $\boldsymbol{\pi} = (1/2, 1/2)$ solves the stationary equation. However, since the chain is oscillating between states 1 and 2, there will be no limiting distribution. The chain clearly has period 2, which violates the sufficient conditions for the above ergodic theorem. Combined with the earlier discussion, this implies that an irreducible finite-state chain needs to be aperiodic to be ergodic. Note that the stationary distribution (1/2, 1/2) still has meaning because it gives the fraction of time the chain spends in each state in the limit, even though there is periodic oscillation.

More on the Reflecting Random Walk

The example Markov chain with single-step transition matrix given by (1) is ergodic, so its steady-state probabilities are found by solving $\boldsymbol{\pi} = \boldsymbol{\pi P}$, written out as the simultaneous system

$$\begin{aligned}
\pi_1 &= \frac{1}{3}\pi_2 \\
\pi_2 &= \pi_1 + \frac{1}{3}\pi_2 + \frac{1}{3}\pi_3 \\
\pi_3 &= \frac{1}{3}\pi_2 + \pi_4 \\
\pi_4 &= \frac{2}{3}\pi_3.
\end{aligned}$$

When these equations are solved and normalized (to sum to 1), a unique solution is found $\boldsymbol{\pi} = (1/9, 3/9, 3/9, 2/9)$. Furthermore, the limiting n-step matrix, $\lim_{n\to\infty} \boldsymbol{P}^n$, would have identical rows all equal to the vector $\boldsymbol{\pi}$.

More on the Gambler's Ruin Problem

For the Gambler's Ruin, there are three classes of states, $\{0\}$, $\{1, 2, \ldots, M-1\}$, and $\{M\}$. After a finite time, the gambler will either reach the goal of M units or lose all the money. Of particular interest is the probability that the gambler's fortune will grow to M before all the resources are lost, denoted here by p_i, $i = 0, 1, \ldots, M$. It is not too difficult to show that

$$p_i = \begin{cases} \dfrac{1 - [(1-p)/p]^i}{1 - [(1-p)/p]^M} & \text{if } p \neq \dfrac{1}{2} \\ \dfrac{i}{M} & \text{if } p = \dfrac{1}{2}. \end{cases}$$

More on the Coin Toss Sequence Problem

For the coin toss sequence example, the single-step transition matrix is given by

$$\boldsymbol{P} = \begin{bmatrix} 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 \\ 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 \end{bmatrix}.$$

This particular matrix is very special since its columns also add up to 1; such a matrix is said to be doubly stochastic. It can be shown that any doubly stochastic transition matrix coming from a recurrent and aperiodic finite chain has the discrete uniform stationary probabilities $\pi_j = 1/M$.

Concluding Remarks

For continuous-time Markov chains, the analog for the single-step transition matrix is the transition rate matrix (infinitesimal generator), where the matrix entries of probabilities are replaced by rates of exponentially distributed random variables. The holding time in a state in a continuous-time Markov chain is exponentially distributed, the analog to the geometric holding time in a state of a discrete-time Markov chain. Well-known examples of continuous-time Markov chains include birth-death processes (analog to random walk), the Poisson process, and many queueing systems with exponentially distributed interarrival and service times, e.g., Jackson queueing networks.

See

- ▶ Birth-Death Process
- ▶ Markov Processes
- ▶ Matrix-Analytic Stochastic Models
- ▶ Networks of Queues
- ▶ Poisson Process
- ▶ Queueing Theory
- ▶ Stochastic Process

References

Breiman, L. (1986). *Probability and stochastic processes, with a view toward applications* (2nd ed.). Palo Alto, CA: The Scientific Press.

Chung, K. L. (1967). *Markov chains with stationary transition probabilities*. New York: Springer-Verlag.

Çinlar, E. (1975). *Introduction to stochastic processes*. Englewood Cliffs, NJ: Prentice-Hall.

Feller, W. (1968). *An introduction to probability theory and its applications* (3rd ed., Vol. 1). New York: Wiley.

Heyman, D. P., & Sobel, M. J. (2004). *Stochastic models in operations research, volume I: Stochastic processes and operating characteristics*. New York: Dover.

Iosifescu, M. (1980). *Finite Markov processes and their application*. New York: Wiley.

Isaacson, D. L., & Madsen, R. W. (1976). *Markov chains: Theory and applications*. New York: Wiley.

Karlin, S., & Taylor, H. M. (1975). *A first course in stochastic processes* (2nd ed.). New York: Academic.

Kemeny, J. G., & Snell, J. L. (1976). *Finite Markov chains*. New York: Springer-Verlag.

Kemeny, J. G., Snell, J. L., & Knapp, A. W. (1976). *Denumerable Markov chains* (2nd ed.). New York: Springer-Verlag.

Meyn, S., & Tweedie, R. L. (2009). *Markov chains and stochastic stability* (2nd ed.). New York: Cambridge University Press.

Markov Decision Processes

Chelsea C. White III
Georgia Institute of Technology, Atlanta, GA, USA

Introduction

The finite-state, finite-action Markov decision process (MDP) is a model of sequential decision making under uncertainty. MDPs have been applied in such diverse fields as health care, highway maintenance, inventory, machine maintenance, cash-flow management, and regulation of water reservoir capacity (Derman 1970; Hernandez-Lermer 1989; Ross 1995; White 1969). After defining an MDP and providing a simple illustrative example, various solution procedures for several different types of MDPs are presented, all of which are based on dynamic programming (Bertsekas 2007; Howard 1971; Puterman 2005; Sennott 1999).

Problem Formulation

Let $k \in \{0,\ 1,\ \ldots,\ K-1\}$ represent the kth stage or decision epoch, i.e., when the kth decision must be selected; $K < \infty$ represents the planning horizon of the Markov decision process. Let s_k be the state of the system to be controlled at stage k. This state must be a member of a finite set S, called the state space, where $s_k \in S$, $k = 0, 1, \ldots, K$. The state process $\{s_k, k = 0, 1, \ldots, K\}$ makes transitions according to the conditional probabilities

$$p_{ij}(a) = \Pr\{s_{k+1} = j | s_k = i,\ a_k = a\},$$

where a_k is the action selected at stage k. The action selected must be a member of the finite action space A, which is allowed to depend on the current state value, i.e., $a_k \in A(i)$ when $s_k = i$, thus allowing a_k to be selected on the basis of the current state s_k for all k. Let δ_k be a mapping from the state space into the action space satisfying $\delta_k(s_k) \in A(s_k)$. Then δ_k is called a policy and a sequence of policies $\pi = \{\delta_0, \ldots, \delta_{K-1}\}$ is known as a strategy.

Let $r(i, a)$ be the one-stage reward accrued at stage $k = 0, 1, \ldots, K-1$, if $s_k = i$ and $a_k = a$. Assume $\bar{r}(i)$ is the terminal reward accrued at stage K (assuming $K < \infty$) if

$s_k = i$. The total discounted reward over the planning horizon accrued by strategy $\pi = \{\delta_0, \ldots, \delta_{K-1}\}$ is then given by

$$\sum_{k=0}^{K-1} \beta^k r(s_k, a_k) + \beta^K \bar{r}(s_k)$$

where $a_k = \delta_k(s_k)$, $k = 0, 1, \ldots, K-1$, where β is the nonnegative real-valued discount factor. The problem objective is to select a strategy that maximizes the expected value of the total discounted reward, with respect to the set of all strategies. Any such strategy is called an optimal strategy.

Example — An inspector must decide at each stage, on the basis of a machine's current state of deterioration, whether to replace the machine, repair it, or do nothing. Assume that the machine can be in one of M states, i.e., the state space is $S = \{1, \ldots, M\}$, where 1 represents the perfect machine state, M represents the failed machine state, and $1 < m < M$ represents an imperfect but functioning state of the machine. Each week the machine inspector can choose to let the machine produce (the do-nothing decision $a = 1$), completely replace the machine (the replace decision $a = R$), or perform some sort of maintenance on the machine, $1 < a < R$. Thus, the action space is $A = \{1, \ldots, R\}$. Generally, these problems are expressed in terms of costs rather than rewards, which can be formulated as $r(i, a) = -c(i, a)$, where $c(i, a)$ be the cost accrued over the following week if at the beginning of the week the machine is in state i and the machine inspector selects action a. Let β be the current value of a dollar to be received next week. Assume the transition probabilities $p_{ij}(a)$ are known for all $i, j \in S$, $a \in A$, where generally $p_{i1}(\mathrm{R}) = 1$ and $p_{ij}(1) = 0$ if $j < i$.

Dynamic Programming Formulation (Finite Stage Case)

To formulate the MDP as a dynamic program for the finite planning horizon case, let $f_k(i)$ be the optimal expected total discounted reward accrued from stage k through the terminal stage K, assuming $s_k = i$. Note that $f_k(i)$ should differ from $f_{k+1}(s_{k+1})$ only by the reward accrued at stage k. In fact, it is easily shown that f_k and f_{k+1} are related by the dynamic programming optimality equation

$$f_k(i) = \max_{a \in A(i)} \left\{ r(i, a) + \beta \sum_{j \in S} p_{ij}(a) f_{k+1}(j) \right\},$$

which has boundary condition $f_K(i) = \bar{r}(i)$. Note also that an optimal strategy $\pi^* = \{\delta_0^*, \ldots, \delta_{K-1}^*\}$ necessarily and sufficiently satisfies

$$f_k(i) = r[i, \delta_k^*(i)] + \beta \sum_j p_{ij}[\delta_k^*(i)] f_{k+1}(j)$$

for all $k = 0, 1, \ldots, K-1$. Thus, the action that should be taken at stage k, given $s_k = i$, is any action that achieves the maximum in

$$\max_{a \in A(i)} \left\{ r(i, a) + \beta \sum_j p_{ij}(a) f_{k+1}(j) \right\}.$$

The Infinite Horizon Discounted Reward Case

For the infinite horizon setting where $K = \infty$, there may exist strategies that could generate an infinite reward. However, if the discount factor β is strictly less than 1, no such strategy exists, which can be verified by noting that

$$\sum_{k=0}^{\infty} \beta^k r(s_k, a_k) \le \sum_{k=0}^{\infty} \beta^k \max_{(i,a)} |r(i, a)| = \frac{\max_{(i,a)} |r(i, a)|}{1 - \beta}.$$

Not surprisingly, the dynamic program for the infinite horizon case can be related to the dynamic program for the finite horizon case. Defining m as the number of stages to go until the terminal stage of the finite horizon case, the dynamic program for the finite horizon problem can then be rewritten as

$$g_{m+1}(i) = \max_{a \in A(i)} \left\{ r(i, a) + \beta \sum_j p_{ij}(a) g_m(j) \right\}$$

where $f_k(i) = g_{K-k}(i)$. Now the optimal expected total discounted reward should be $g(i) = \lim_{m \to \infty} g(i)$ for initial state i, which should satisfy

$$g(i) = \max_{a \in A(i)} \left\{ r(i,a) + \beta \sum_j p_{ij}(a) g(i) \right\} \quad (1)$$

if the limit and maximization operators can be interchanged. It so happens that this interchange is possible under the conditions considered here, and hence the optimal expected total discounted reward uniquely satisfies (1). It can also be shown that an optimal strategy exists that is stage invariant and that this strategy, or equivalently, policy, satisfies

$$g(i) = r[i, \delta^*(i)] + \beta \sum_j p_{ij}[\delta^*] g(j) \quad (1a)$$

for all $i \in S$.

Solution Procedures

Three different computational approaches for determining g and δ^* in (1) are presented.

Linear Programming — The following linear program can solve the infinite-horizon discounted MDP:

$$\text{minimize} \sum_{i \in S} g(i)$$
$$\text{subject to} \quad g(i) - \beta \sum_j p_{ij}(a) g(j) \geq r(i,a)$$

where the constraint inequality must be satisfied for all $i \in S$ and $a \in A(i)$, $i \in S$.

Successive Approximations — This procedure, in its simplest form, involves determining $g_m(i)$ for large m, using the iteration equation

$$g_m(i) = \max_{a \in A(i)} \left\{ r(i,a) + \beta \sum_j p_{ij}(a) g_{m-1}(j) \right\},$$

where $g_0(i)$ can be arbitrarily selected; however, it is generally beneficial to select g_0 as close to g as possible if there is some way of estimating g *a priori*.

Policy Iteration — This computational procedure involves the following iterative approach:

Step 0: Select δ

Step 1: Determine g_δ where g_δ, satisfy

$$g_\delta(i) = r[i, \delta(i)] + \beta \sum_j p_{ij}[\delta(i)] g_\delta(j).$$

Note that

$$g_\delta = (I - \beta P_\delta)^{-1} r_\delta$$

where $P_\delta = \{p_{ij}[\delta(i)]\}$, $g_\delta = \{g_\delta(i)\}$, $r_\delta = \{r[i, \delta(i)]\}$, I is the identity matrix, and the inverse is guaranteed to exist since $\beta < 1$.

Step 2: Determine δ' that satisfies

$$r[i, \delta'(i)] + \beta \sum_j p_{ij}[\delta'(i)] g_\delta(j) = \max_{a \in A(i)} \left\{ r(i,a) + \beta \sum_j p_{ij}(a) g_\delta(j) \right\}.$$

Step 3: Set $\delta = \delta'$ and return to Step 1 until g_δ and $g_{\delta'}$ are sufficiently close.

Note that each of the above solution procedures is far more efficient than exhaustive enumeration. Combining policy iteration and successive approximations can lead to efficient computational procedures for large-scale infinite-horizon discounted MDPs.

Markov Decision Processes without Discounting (The Average Reward Case)

Assume that the criterion is

$$\lim_{K \to \infty} \left(\frac{1}{K+1} \right) E \left\{ \sum_{k=0}^{K} r(S_k, a_k) \right\}$$

which is the expected average reward criterion. When the system operates under stationary policy δ, it can be shown that there exist values $v_\delta(i)$, $i \in S$, and a state independent gain γ_δ, which satisfy

$$\gamma_\delta + v_\delta(i) = r[i, \delta(i)] + \sum_j p_{ij}[\delta(i)] v_\delta(i) \quad (2)$$

if P_δ is ergodic. Let γ^*, δ^* and v be such that

$$\gamma^* + v(i) = \max_{a \in A(i)} \left\{ r(i,a) + \beta \sum_j p_{ij} v(j) \right\} = r[i, \delta^*(i)] + \sum_j p_{ij}[\delta^*(i)] v(j)$$

where P_δ is assumed ergodic for all δ. Then, γ^* is the value of the criterion generated by an optimal strategy and δ^* is an optimal strategy. The following is a policy iteration procedure for determining γ^*, δ^* and v, where it is necessary only to know v up to a positive constant due to the sum-to-one characteristic of the probabilities.

Algorithm. Step 0: Choose δ.

Step 1: Solve equation (2) for v_δ and γ_δ, where for some i, $v_\delta(i) = 0$.

Step 2: Determine a policy δ' that achieves the maximum in

$$\max_{a \in A(i)} \left\{ r(i,a) + \sum_j p_{ij} v_\delta(i) \right\}.$$

Step 3: Set $\delta = \delta'$ and go to Step 1 until γ_δ and $\gamma_{\delta'}$ are sufficiently close.

Concluding Remarks

The discussion has focused on the MDP setting where the state and action spaces are finite; the reward is separable with respect to stage; all rewards, the discount factor, and all transition probabilities are known precisely and the current state can be accurately made available to the decision maker before selection of the current alternative. The references treat more general settings. Much research effort is devoted to improving the computational tractability of large-scale MDPs so as to improve both the validity and tractability of this modeling tool. One such approach is approximate dynamic programming, which is treated in detail in Volume II of Bertsekas (2007).

See

- ▶ Approximate Dynamic Programming
- ▶ Dynamic Programming
- ▶ Markov Chains
- ▶ Markov Processes

References

Bertsekas, D.P. (2007). *Dynamic programming and optimal control* (Vols. I & II, 3rd edn.). (Vol. II, 4th edn., 2012). Nashua, NH: Athena Scientific.

Derman, C. (1970). *Finite state Markovian decision processes*. New York: Academic.

Hernandez-Lermer, O. (1989). *Adaptive Markov control processes*. New York: Springer.

Howard, R. (1971). *Dynamic programming and Markov processes*. Cambridge, MA: MIT Press.

Puterman, M. L. (2005). *Markov decision processes: Discrete stochastic dynamic programming*. New York: John Wiley & Sons.

Ross, S. M. (1995). *Introduction to stochastic dynamic programming*. New York: Academic.

Sennott, L. I. (1999). *Stochastic dynamic programming and the control of queueing systems*. New York: John Wiley & Sons.

White, D. J. (1969). *Markov decision processes*. Chichester, UK: John Wiley.

Markov Processes

Douglas R. Miller
George Mason University, Fairfax, VA, USA

Introduction

A Markov process is a stochastic process $\{X(t), t \in T\}$ with state space S and time domain T that satisfies the Markov property, which is also known as lack of memory. In general, probabilities of behavior of a stochastic process at future times usually depend on the behavior of the process at times in the past. The Markov property means that probabilities of future events are completely determined by the present state of the process: if the current state of the process is known, then the past behavior of the process provides no additional information in determining the probabilities of future events. Mathematically, the process $\{X(t), t \in T\}$ is Markov if, for any $n > 0$, any $t_1 < t_2 < \ldots, < t_n < t_{n+1}$ in the time domain T, and any states $x_1, x_2, \ldots, x_n$ and any set A in the state space S,

$$\Pr\{X(t_{n+1}) \in A | X(t_1) = x_1, \ldots, X(t_n) = x_n\} = \Pr\{X(t_{n+1}) \in A | X(t_n) = x_n\}.$$

The conditional probabilities on the right-hand side of this equation are the transition probabilities of the Markov process; they play a key role in the study of Markov processes. The transition probabilities of the process are presented as a transition function $p(s, x; t, A) = \Pr\{X(t) \in A \,|\, X(s) = x\}$, $s < t$, for $s, t \in T$, $x \in S$, and $A \subset S$. The initial distribution of the

process is $q(A) = \Pr\{X(0) \in A\}$, for $A \subset S$. The distribution of a Markov process is uniquely determined by an initial distribution $q(\cdot)$ and a transition function $p(\cdot,\ldots,\cdot)$: for $0 = t_0 < t_1 < \ldots < t_n$ in the time domain, and subsets A_1, A_2,..., A_n of the state space S,

$$\Pr\{X(t_1) \in A_1, \ldots, X(t_n) \in A_n\}$$
$$= \int_{x_0 \in S} q(dx_0) \int_{x_1 \in A_1} p(t_0, x_0; t_1, dx_1) \cdots$$
$$\int_{x_{n-1} \in A_{n-1}} p(t_{n-2}, x_{n-2}; t_{n-1}, dx_{n-1}) p(t_{n-1}, x_{n-1}; t_n, A_n).$$

An equivalent interpretation of the Markov property is that the past behavior and the future behavior of the process are conditionally independent given the present state of the process: for any $m > 0$, any $n > 0$, any $t_{-m} < \ldots, < t_{-1} < t_0 < t_1 < \ldots, < t_n$ in the time domain, and any state x_0 and any sets $A_1, A_2, \ldots, A_m$ and $B_1, B_2, \ldots, B_n$ in the state space S,

$$\Pr\{X(t_{-m}) \in A_m, \ldots, X(t_{-1}) \in A_1, X(t_1) \in B_1, \ldots, X(t_n) \in B_n | X(t_0) = x_0\}$$
$$= \Pr\{X(t_{-m}) \in A_m, \ldots, X(t_{-1}) \in A_1 | X(t_0) = x_0\} \cdot \Pr\{X(t_1) \in B_1, \ldots, X(t_n) \in B_n | X(t_0) = x_0\}.$$

A Markov process has stationary transition probabilities if the transition probabilities are time invariant, i.e., for s, $t > 0$, $\Pr\{X(s + t) \in A \mid X(s) = x\} = \Pr\{X(t) \in A | X(0) = x\}$. In this case the transition function takes the simplified form $p_t(x, A) = \Pr\{X(t) \in A \mid X(0) = x\}$. Most Markov process models assume stationary transition probabilities.

Classification of Markov Processes

There is a natural classification of Markov processes according to whether the time domain T and the state space S are denumerable or non-denumerable. This yields four general classes. Denumerable time domains are usually modeled as the integers or non-negative integers. Non-denumerable time domains are usually modeled as the continuum ($\mathcal{R}$ or $[0, \infty]$). Denumerable state spaces can be modeled as the integers, but it is often useful to retain other descriptions of the states rather than simply enumerating them. Non-denumerable state spaces are usually modeled as a one or higher dimensional continuum. Roughly speaking, discrete is equivalent to denumerable and continuous is equivalent to non-denumerable. In 1907, Markov considered a discrete time domain and a finite state space; he used the word "chain" to denote the dependence over time, hence the term Markov chain for Markov processes with discrete time and denumerable states. See Maistrov (1974) for some historical discussion and see Appendix B of Howard (1971) for a reprint of one of Markov's 1907 papers. There is no universal convention for the scope of definition of Markov chain. Chung (1967) and most elementary operations research/management science textbooks (e.g., Hillier and Lieberman 2009) define Markov processes with denumerable state spaces to be Markov chains. Iosifescu (1980) and the Romanian school use the convention that Markov chain applies to discrete time and any state space, while Markov process applies to continuous time and any state space. The terminology varies in popular texts: Karlin and Taylor (1975, 1981) and Ross (1995) agree with Chung; Breiman (1968) and Çinlar (1975) agree with the Romanians. The terms discrete-time Markov chain (DTMC) and continuous-time Markov chain (CTMC) are sometimes used to clarify the situation.

Here are four examples of Markov processes representing the four classes with respect to discrete or continuous time and denumerable or continuous state space.

(a) *Gambler's Ruin* (*discrete time/denumerable states*). A gambler makes repeated bets. On each bet he wins \$1 with probability p or loses \$1 with probability $1 - p$. Outcomes of successive bets are independent events. He starts with a certain initial stake and will play repeatedly until he loses all his money or until he increases his fortune to \$$M$. Let X_n equal the gambler's wealth after n plays. The stochastic process $\{X_n, n = 0, 1, 2, \ldots\}$ is a discrete time Markov chain (DTMC) with state space $\{0, 1, 2, \ldots, M\}$. The Markov property follows from the assumption that outcomes of successive bets are independent events. The Markov model can be used to derive performance measures of interest for this situation: for example, the probability he loses all his money, the probability he reaches his goal of \$$M$, and the expected number of times he makes a bet. All these performance measures are

functions of his initial stake x_0, probability p and goal M. (The gambler's fortune is a random walk with absorbing boundaries 0 and M.) The gambler's ruin is a simplification of more complex systems that experience random rewards, risk, and possible ruin; for example, insurance companies.

(b) *A Maintenance System* (*continuous time/denumerable states*). A system consists of two machines and one repairman. Each machine operates until it breaks down. The machine is then repaired and put back into operation. If the repairman is busy with the other machine, the just broken machine waits its turn for repair. So, each machine cycles through the states: operating (O), waiting (W), and repairing (R). Labeling the machines as "1" and "2" and using the corresponding subscripts, the states of the system are (O_1,O_2), (O_1,R_2), (R_1,O_2), (W_1,R_2) and (R_1,W_2). Assume that all breakdown instances and repairs are independent of each other and that the operating times until breakdown and the repair times are random with exponential distributions. The mean operating times for the machines are $1/\alpha_1$ and $1/\alpha_2$, respectively (so the machines break down at rates α_1 and α_2). The mean repair times for the machines are $1/\beta_1$ and $1/\beta_2$, respectively (so the machines are repaired at rates β_1 and β_2). Letting $X_i(t)$ equal the state of machine i at time t, the stochastic process $\{(X_1(t), X_2(t)), 0 \leq t\}$ is a continuous time Markov chain (CTMC) on a state space consisting of five states. The Markov property follows from the assumption about independent exponential operating times and repair times. (The exponential distribution is the only continuous distribution with lack-of-memory.) For this type of system there are several performance measures of interest: for example, the long-run proportion of time both machines are broken or the long-run average number of working machines. This maintained system is a simplified example of more complex maintained systems.

(c) *Quality Control System (discrete time/continuous states)*. A manufacturing system produces a physical part that has a particularly critical length along one dimension. The specified value for the length is α. However, the manufacturing equipment is imprecise. Successive parts produced by this equipment vary randomly from the desired value, α. Let X_n equal the size of the nth part produced. The noise added to the system at each step is modeled as $D_n \sim \text{Normal}(0, \delta^2)$. The system can be controlled by attempting to correct the size of the $(n+1)$st part by adding $c_n = -\beta\,(x_n - \alpha)$ to the current manufacturing setting after observing the size x_n of the nth part; however, there is also noise in the control so that, in fact, $C_n \sim \text{Normal}\,(c_n,(\gamma c_n)^2)$ is added to the current setting. This gives $X_{n+1} = X_n + C_n + D_n$. The process $\{X_n,\ n = 0, 1, 2,\ldots\}$ is a discrete-time Markov process on a continuous state space. The Markov property will follow if all the noise random variables $\{D_n\}$ are independent and the control random variables $\{C_n\}$ depend only on the current setting (X_n) of the system. Performance measures of interest for this system include the long-run distribution of lengths produced (if the system is stable over the long-run). There is also a question of determining the values of β for which the system is stable and then finding the optimal value of β.

(d) *Brownian Motion* (*continuous time/continuous states*). In 1828, English botanist Robert Brown observed random movement of pollen grains on the surface of water. The motion is caused by collisions with water molecules. The displacement of a pollen grain as a function of time is a two-dimensional Brownian motion. A one-dimensional Brownian motion can be obtained by scaling a random walk: Consider a sequence of independent, identically-distributed random variables, Z_i, with $\Pr\{Z_i = +1\} = \Pr\{Z_i = -1\} = 1/2$, $i = 1, 2,\ldots$. Let $S_n = \sum_{i=1}^{n} Z_i$, $n = 0, 1, 2,\ldots$. Then, let $X_n(t) = n^{-1/2} S_{[nt]}, 0 \leq t \leq 1$, $n = 1, 2,\ldots$, where $[nt]$ is the greatest integer $\leq nt$. As $n \to \infty$, the sequence of processes $\{X_n(t), 0 \leq t \leq 1\}$ converges to $\{W(t), 0 \leq t \leq 1\}$, standard Brownian motion or the Wiener process; see Billingsley (1968). The Wiener process is a continuous-time, continuous-state Markov process. The sample paths of the Wiener process are continuous. Diffusions are the general class of continuous-time, continuous-state Markov processes with continuous sample paths. Diffusion models are useful approximations to discrete processes analogous to how the Wiener process is an approximation to the above random

walk process $\{S_n,\ n = 0,\ 1,\ 2,\ldots\}$; see Glynn (1990). Geometric Brownian motion $\{Y(t), 0 \le t\}$ is defined as $Y(t) = \exp(\sigma W(t))$, $0 \le t$; it is a diffusion. Geometric Brownian motion has been suggested as a model for stock price fluctuations; see Karlin and Taylor (1975). A performance measure of interest is the distribution of the maximum value of the process over a finite time interval.

There are various performance measures that can be derived for Markov process models. Some specific performance measures were mentioned for the above examples. Some general behavioral properties and performance measures are now described. The descriptions are for a discrete-time Markov chain $\{X_n,\ n = 0,\ 1,\ 2,\ldots\}$ but similar concepts apply to other classes of Markov processes. A Markov chain is strongly ergodic if X_n converges in distribution as $n \to \infty$, independent of the initial state x_0. A Markov chain is weakly ergodic if $n^{-1}\sum_{i=1}^{n} X_i$ converges to a constant as $n \to \infty$, independent of the initial state x_0. Also as $n \to \infty$, under certain conditions and for real-valued functions f: $S \to \mathbb{R}$, $f(X_n)$ converges in distribution, $n^{-1}\sum_{i=1}^{n} f(X_i)$ converges to a constant, and $n^{-1/2}\sum_{i=1}^{n}[f(X_i) - Ef(X_i)]$ is asymptotically normal. Markov process theory identifies conditions for ergodicity, conditions for the existence of limits, and provides methods for evaluation of limits when they exist. For example, in the above maintained system example, $f(\cdot)$ might be a cost function and the performance measure of interest is long-run average cost. The above performance is long-run (or infinite-horizon, or steady-state, or asymptotic) behavior. Short-run (or finite-horizon, or transient) behavior and performance is also of interest. For a subset A of the state space S, the first passage time T_A is the time of the first visit of the process to A: $T_A = \min\{n\colon X_n \in A\}$. The hitting probability $\Pr\{T_A < \infty\}$, the distribution of T_A, and $E(T_A)$ are of interest. In the gambler's ruin example, the gambler wants to know the hitting probabilities for sets $\{0\}$ and $\{M\}$. Transient analysis of Markov processes investigates these and other transient performance measures. The analysis of performance measures takes on different forms for the four different classes of Markov processes.

Evaluation of performance measures for Markov process models of complex systems may be difficult. Standard numerical analysis algorithms are sometimes useful, and specialized algorithms have been developed for Markov models; for example, see Grassmann (1990). Workers in the field of computational probability have developed and evaluated numerical solution techniques for Markov models by exploiting special structure and probabilistic behavior of the system or by using insights gained from theoretical probability analysis. In this spirit, Neuts (1981) has developed algorithms for a general class of Markov chains. A structural property of Markov chains called reversibility leads to efficient numerical methods of performance evaluation; see Keilson (1979), Kelly (1979), and Whittle (1986). There is a relationship between discrete-time and continuous-time Markov chains called uniformization or randomization that can be used to calculate performance measures of continuous-time Markov chains; see Keilson (1979) and Gross and Miller (1984). For Markov chains with huge state spaces, Monte Carlo simulation can be used as an efficient numerical method for performance evaluation; see, for example, Hordijk, Iglehart and Schassberger (1976) and Fox (1990).

There are classes of stochastic processes related to Markov processes. There are stochastic processes that exhibit some lack of memory but are not Markovian. Regenerative processes have lack of memory at special points (regeneration points) but at other times the process has a memory; see Çinlar (1975). A semi-Markov process is a discrete-state continuous-time process that makes transitions according to a DTMC but may have general distributions of holding times between transitions; see Çinlar (1975). It is sometimes possible to convert a non-Markovian stochastic process into a Markov process by expanding the state description with supplementary variables; that is, $\{X(t), 0 \le t\}$ may be non-Markovian but $\{(X(t), Y(t)),\ 0 \le t\}$ is Markovian. Supplementary variables are often elapsed times for phenomena with memory; in this way very general discrete state stochastic systems can be modeled as Markov processes with huge state spaces. The general model for discrete-event dynamic systems is the generalized semi-Markov process (GSMP); see Whitt (1980) and Cassandras and Lafortune (2008).

The index set T of a stochastic process $\{X(t), t \in T\}$ may represent "time" or "space" or both, leading to temporal processes, spatial processes, or spatial-temporal processes when the index set is time, space, or space-time, respectively. Stochastic processes with multi-dimensional index sets are called random

fields. The Markov property can be generalized to the context of multi-dimensional index sets resulting in Markov random fields; see Kelly (1979), Kindermann and Snell (1980) and Whittle (1986). Markov random fields have many applications. They are models for statistical mechanical systems (interacting particle systems). They are useful in texture analysis and image analysis; see Chellappa and Jain (1993).

See

- Hidden Markov Models
- Markov Chain Monte Carlo
- Markov Chains
- Markov Decision Processes
- Markov Random Field
- Monte Carlo Simulation
- Regenerative Process
- Regenerative Simulation
- Reversible Markov Chain/Process

References

Billingsley, P. (1968). *Convergence of probability measures*. New York: Wiley.

Breiman, L. (1968). *Probability*. Reading, MA: Addison-Wesley.

Breiman, L. (1986). *Probability and stochastic processes, with a view toward applications* (2nd ed.). Palo Alto, CA: The Scientific Press.

Cassandras, C. G., & Lafortune, S. (2008). *Discrete event systems: Modeling and performance analysis* (2nd ed.). New York: Springer.

Chellappa, R., & Jain, A. (Eds.). (1993). *Markov random fields: Theory and application*. San Diego: Academic Press.

Chung, K. L. (1967). *Markov chains with stationary transition probabilities*. New York: Springer-Verlag.

Çinlar, E. (1975). *Introduction to stochastic processes*. Englewood Cliffs, NJ: Prentice-Hall.

Feller, W. (1968). *An introduction to probability theory and its applications, volume I* (3rd ed.). New York: Wiley.

Feller, W. (1971). *An introduction to probability theory and its applications, volume II* (2nd ed.). New York: Wiley.

Fox, B. L. (1990). Generating Markov-chain transitions quickly. *ORSA Journal on Computing, 2*, 126–135.

Glynn, P. W. (1989). A GSMP formalism for discrete event systems. *Proceedings of the IEEE, 77*, 14–23.

Glynn, P. W. (1990). Diffusion approximations. In D. P. Heyman & M. J. Sobel (Eds.), *Handbooks in OR and MS* (Vol. 2, pp. 145–198). Amsterdam: Elsevier Science.

Grassmann, W. K. (1990). Computational methods in probability. In D. P. Heyman & M. J. Sobel (Eds.), *Handbooks in OR and MS* (Vol. 2, pp. 199–254). Amsterdam: Elsevier Science.

Gross, D., & Miller, D. R. (1984). The randomization technique as a modelling tool and solution procedure for transient Markov processes. *Operations Research, 32*, 343–361.

Heyman, D. P., & Sobel, M. J. (1982). *Stochastic models in operations research, volume I: Stochastic processes and operating characteristics*. New York: McGraw-Hill.

Hillier, F. S., & Lieberman, G. J. (2009). *Introduction to operations research* (9th ed.). New York: McGraw-Hill.

Hordijk, A., Iglehart, D. L., & Schassberger, R. (1976). Discrete-time methods for simulating continuous-time Markov chains. *Advances in Applied Probability, 8*, 772–778.

Howard, R. A. (1971). *Dynamic probabilistic systems, volume I: Markov models*. New York: Wiley.

Iosifescu, M. (1980). *Finite Markov processes and their application*. New York: Wiley.

Isaacson, D. L., & Madsen, R. W. (1976). *Markov chains: Theory and applications*. New York: Wiley.

Karlin, S., & Taylor, H. M. (1975). *A first course in stochastic processes* (2nd ed.). New York: Academic Press.

Karlin, S., & Taylor, H. M. (1981). *A second course in stochastic processes*. New York: Academic Press.

Keilson, J. (1979). *Markov chain models – Rarity and exponentiality*. New York: Springer-Verlag.

Kelly, F. P. (1979). *Reversibility and stochastic networks*. New York: Wiley.

Kemeny, J. G., & Snell, J. L. (1976). *Finite Markov chains*. New York: Springer-Verlag.

Kemeny, J. G., Snell, J. L., & Knapp, A. W. (1976). *Denumerable Markov chains* (2nd ed.). New York: Springer.

Kindermann, R., & Snell, J. L. (1980). *Markov random fields and their applications*. Providence, RI: American Mathematical Society.

Maistrov, L. E. (1974). *Probability theory: A historical sketch*. New York: Academic Press.

Neuts, M. F. (1981). *Matrix-geometric solutions in stochastic models*. Baltimore: The Johns Hopkins University Press.

Parzen, E. (1962). *Stochastic processes*. San Francisco: Holden-Day.

Ross, S. M. (1995). *Stochastic processes* (2nd ed.). New York: Wiley.

Snell, J. L. (1988). *Introduction to probability*. New York: Random House.

Whitt, W. (1980). Continuity of generalized semi-Markov processes. *Mathematical Methods of Operations Research, 5*, 494–501.

Whittle, P. (1986). *Systems in stochastic equilibrium*. New York: Wiley.

Markov Property

When the behavior of a stochastic process $\{X(t), t \in T\}$ at times in the future depends only on the present state of the process (past behavior of the process affects the future behavior only through the present state of the process); viz., for any $n > 0$, any set of time points $t_1 < t_2 < \ldots < t_n < t_{n+1}$ in the time domain T, and any

states $x_1, x_2, \ldots, x_n$ and any set A in the same space, $\Pr\{X(t_{n+1}) \in A | X(t_1) = x_1, \ldots, X(t_n) = x_n\} = \Pr\{X(t_{n+1}) \in A | X(t_n) = x_n\}$.

See

- ▶ Markov Chains
- ▶ Markov Processes

Markov Random Field

A random field that satisfies a generalization of the Markov property.

See

- ▶ Markov Processes
- ▶ Random Field

Markov Renewal Process

When the times between successive transitions of a Markov chain are independent random variables indexed on the to and from states of the chain.

See

- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Networks of Queues
- ▶ Renewal Process

Markov Routing

The process of assigning customers to nodes in a queueing network according to a Markov chain over the set of nodes, where $p(j, k)$ is the probability that a customer exiting node j proceeds next to node k, with $1 - \sum p(j, k)$ being the probability a customer leaves the network from node j (the sum is over all nodes of the network, including leaving the network altogether).

See

- ▶ Networks of Queues

Markovian Arrival Process (MAP)

- ▶ Matrix-Analytic Stochastic Models

Marriage Problem

Given a group of m men and m women, the marriage problem is to couple the men and women such that the total happiness of the group is maximized when the assigned couples marry. The women and the men determine an $m \times m$ table of happiness coefficients, where the coefficient a_{ij} represents the happiness rating for the couple formed by woman i and man j if they marry. The larger the a_{ij}, the higher the happiness. The problem can be formulated as an assignment problem whose solution matches each woman to one man. This result, which is due to the fact that the assignment problem has a solution in which the variables can take on only the values of 0 or 1, is sometimes used to prove that monogomy is the best form of marriage.

See

- ▶ Assignment Problem

Martingale

A stochastic process (with finite expectation) for which the conditional expectation of future values is equal to the present value. For example, for a discrete-time process $\{X_0, X_1, X_2, \ldots\}$,

$$E[X_{n+1} | X_0, X_1, \ldots X_n] = X_n.$$

Master Problem

The transformed extreme-point problem that results when applying the Dantzig-Wolfe decomposition algorithm.

See

► Dantzig-Wolfe Decomposition Algorithm

Matching

Richard W. Eglese
Lancaster University, Lancaster, UK

Introduction

Matching problems form an important branch of graph theory. They are of particular interest because of their application to problems found in Operations Research. Matching problems also form a class of integer-linear programming problems which can be solved in polynomial time. A good description of the historical development of matching problems and their solutions is contained in the preface of Lovasz and Plummer (2009).

Given a simple non-directed graph $G = [V, E]$ (where V is a set of vertices and E is a set of edges), then a matching is defined as a subset of edges M such that no two edges of M are adjacent. A matching is said to *span* a set of vertices X in G if every vertex in X is incident with an edge of the matching. A perfect matching is a matching which spans V. A maximum matching is a matching of maximum cardinality, i.e. a matching with the maximum number of members in the set.

A graph is called a bipartite graph if the set of vertices V is the disjoint union of sets V_1 and V_2 and every edge in E has the form (v_1, v_2) where v_1 is a member of V_1 and v_2 is a member of V_2.

Matching on Bipartite Graphs

The first type of matching problems consists of those which can be formulated as matching problems on a bipartite graph. For example, suppose V_1 represents a set of workers and V_2 represents a set of tasks to be performed. If each worker is able to perform a subset of the tasks and each task may be performed by some subset of the workers, the situation may be modeled by constructing a bipartite graph G, where there is an edge between v_1 in V_1 and v_2 in V_2 if and only if worker v_1 can perform task v_2. If it is assumed that each worker may only be assigned one task and each task may only be assigned to be carried out by one worker, the problem is an assignment problem. To find the maximum number of tasks which can be performed, the maximum matching on G must be found. If a measure of effectiveness can be associated with assigning a worker to a task, then the question may be asked as to how the workers should be assigned to tasks to maximize the total effectiveness. This is a maximum weighted matching problem. If costs are given in place of measures of effectiveness, the minimum cost assignment problem can be solved as a maximum weighted matching problem after replacing each cost by the difference between it and the maximum individual cost. This assumes all workers or all tasks must be assigned.

Both forms of assignment problem can be solved by a variety of algorithms. For example, a maximum matching on a bipartite graph can be found by modeling the problem as a network flow problem and finding a maximum flow on the model network. A more efficient algorithm is due to Hopcraft and Karp (1973). A well-known algorithm for solving the maximum weighted matching problem (for which the maximum matching problem can be considered a special case) on a bipartite graph is often referred to as the Hungarian method and was introduced by Kuhn (1955, 1956). Kuhn casts the procedure in terms of a primal-dual linear program. The algorithm can be implemented so as to produce an optimal matching in $O(m^2 n)$ steps, where n is the number of vertices and m is the number of edges in the graph. The details are given in Lawler (1976). Although this is an efficient algorithm, it may be necessary to find faster implementations for problems of large size or when the algorithm is used repeatedly as part of a more complex procedure. Various methods have been proposed including those due to Jonker and Volgenant (1986) and Wright (1990).

Job Scheduling

Another example of a problem which can be modeled as a matching problem arises from job scheduling (Coffman and Graham 1972). Suppose n jobs are to be processed and there are two machines available. All jobs require an equal amount of time to complete and can be processed on either machine. However there are precedence constraints which mean that some jobs must be completed before others are started. What is the shortest time required to process all n jobs?

This example can be modeled by constructing a graph G with n vertices representing the n jobs and where an edge joins two vertices if and only if they can be run simultaneously. An optimum schedule corresponds to one where the two machines are used simultaneously as often as possible. Therefore the problem becomes one of finding the maximum matching on G, from which the shortest time can be derived. In this case though, the graph G is no longer bipartite and so an algorithm for solving the maximum matching problem on a general graph is required.

The first efficient algorithm to find a maximum matching in a graph was developed by Edmonds (1965a). Most successful algorithms to find a maximum matching have been based on Edmonds' ideas. Gabow (1976) and Lawler (1976) show how to implement the algorithm in a time of $O(n^3)$. It is possible to modify the algorithm for more efficient performance on large problems. For example, Even and Kariv (1975) present an algorithm running in a time of $O(n^{5/2})$ and Micali and Vazirani (1980) describe an algorithm with running time of $O(mn^{1/2})$.

Arc Routing

There is a close connection between arc routing problems and matching. Suppose a person must deliver mail along all streets of a town. What route will traverse each street and return to the starting point in minimum total distance? This problem is known as the Chinese Postman Problem as it was first raised by the Chinese mathematician Meigu Guan (1962). It may be formulated as finding the minimum length tour on a non-directed graph G whose edges represent the streets in the town and whose vertices represent the junctions, where each edge must be included at least once. Edmonds and Johnson (1973) showed that this problem is equivalent to finding a minimum weighted matching on a graph whose vertices represent the set of odd nodes in G and whose edges represent the shortest distances in G between the odd nodes. Odd nodes are vertices where an odd number of edges meet. This minimum weighted matching problem can be solved efficiently by the algorithm introduced by Edmonds (1965b) for maximum weighted matching problems where the weights on each edge are the distances multiplied by minus one. The Chinese Postman Problem is therefore easier to solve than the Traveling Salesman Problem where a polynomially bounded algorithm has not yet been established.

For large problems, faster versions of the weighted matching algorithm have been developed by Galil, Micali and Gabow (1982) and Ball and Derigs (1983) which require $O(mn \log n)$ steps. A starting procedure which significantly reduces the computing time for the maximum matching problem is described by Derigs and Metz (1986) and involves solving the assignment problem in a related bipartite graph.

b-Matchings

Given an integer b_i for each vertex v_i of V, a b-matching of G is defined as a subset M of edges, such that at each vertex v_i, the number of edges of M incident on v_i is less than or equal to b_i. A matching is therefore a special case of a b-matching where $b_i = 1$ for all i. Efficient algorithms for b-matching problems are described in Gerards (1995), which also provides a good survey of matching in general.

Lower bounds for Vehicle Routing problems can be obtained by relaxing the subtour elimination and vehicle capacity constraints to give a perfect b-matching problem. Miller (1995) shows that this approach can be used in a branch-and-bound frame-work for this application.

See

- ▶ Assignment Problem
- ▶ Branch and Bound
- ▶ Chinese Postman Problem
- ▶ Dual Linear-Programming Problem
- ▶ Graph Theory
- ▶ Hungarian Method

- ▶ Integer and Combinatorial Optimization
- ▶ Maximum-Flow Network Problem
- ▶ Network
- ▶ Transportation Problem
- ▶ Traveling Salesman Problem
- ▶ Vehicle Routing

References

Ball, M. O., & Derigs, U. (1983). An analysis of alternate strategies for implementing matching algorithms. *Networks, 13*, 517–549.

Coffman, E. G., Jr., & Graham, R. L. (1972). Optimal scheduling for two processor systems. *Acta Informatica, 1*, 200–213.

Derigs, U., & Metz, A. (1986). On the use of optimal fractional matchings for solving the (integer) matching problem. *Computing, 36*, 263–270.

Edmonds, J. (1965a). Paths, trees, and flowers. *Canadian Journal of Mathematics, 17*, 449–467.

Edmonds, J. (1965b). Maximum matching and a polyhedron with (0,1) vertices. *Journal of Research National Bureau of Standards, Section B, 69B*, 125–130.

Edmonds, J., & Johnson, E. L. (1973). Matching, Euler tours and the Chinese postman. *Math Programming, 5*, 88–124.

Even, S., & Kariv, O. (1975). An O($n^{5/2}$) algorithm for maximum matching in general graphs. 16th *Annual symposium on foundations of computer science*, IEEE Computer Society Press, New York, pp. 100–112.

Gabow, H. N. (1976). An efficient implementation of Edmond's algorithm for maximum matching on graphs. *Journal of the Association for Computing Machinery, 23*, 221–234.

Galil, Z., Micali, S., & Gabow, H. (1982). Priority queues with variable priority and an O (*EV* log *V*) algorithm for finding a maximal weighted matching in general graphs. *23rd Annual symposium on foundations of computer science*, IEEE Computer Society Press, New York, pp. 255–261.

Gerards, A. M. H. (1995). Matching. In M. O. Ball, T. L. Magnanti, C. L. Monma, & G. L. Nemhauser (Eds.), *Network models, handbooks in operations research and management science* (Vol. 7, pp. 135–224). Amsterdam: Elsevier.

Gondran, M., & Minoux, M. (1984). *Graphs and algorithms*. Chichester: Wiley.

Guan, M. (1962). Graphic programming using odd and even points. *Chinese Mathematics, 1*, 273–277.

Hopcroft, J. E., & Karp, R. M. (1973). An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing, 2*, 225–231.

Jonker, R., & Volgenant, A. (1986). Improving the Hungarian assignment algorithm. *Operations Research Letters, 5*, 171–175.

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly, 2*, 83–97.

Kuhn, H. W. (1956). Variants of the Hungarian method for assignment problems. *Naval Research Logistics Quarterly, 3*, 253–258.

Lawler, E. L. (1976). *Combinatorial optimization, networks and matroids*. New York: Holt, Rinehart and Winston.

Lovasz, L., & Plummer, M. D. (2009). *Matching theory*. Providence, RI: AMS Chelsea.

McHugh, J. A. (1990). *Algorithmic graph theory*. London: Prentice-Hall.

Micali, S., & Vazirani, V. V. (1980). An O($V^{1/2}$E) algorithm for finding maximum matching in general graphs. *21st Annual symposium on foundations of computer science*, IEEE Computer Society Press, New York, pp. 17–27.

Miller, D. L. (1995). A matching based exact algorithm for capacitated vehicle routing problems. *ORSA Journal of Computing, 7*, 1–9.

Wright, M. B. (1990). Speeding up the Hungarian algorithm. *Computers and Operations Research, 17*, 95–96.

Material Handling

Meir J. Rosenblatt
Washington University in St. Louis, St. Louis, MO, USA
Technion – Israel Institute of Technology, Haifa, Israel

Introduction

Material handling is concerned with moving raw materials, work-in-process, and finished goods into the plant, within the plant, and out of the plant to warehouses, distribution networks, or directly to the customers. The basic objective is to move the right combination of tools and materials (raw materials, parts and finished products) at the right time, to the right place, in the right form, and in the right orientation. And to do it with the minimum total cost.

It is estimated that 20% to 50% of the total operating expenses within manufacturing are attributed to material handling (Tompkins et al. 1996). Material handling activities may account for 80% to 95% of total overall time spent between receiving a customer order and shipping the requested items (Rosaler and Rice 1994). This indicates that improved efficiencies in material handling activities can lead to substantial reductions in product cost and production lead-time; better space and equipment utilization, improved working conditions and safety, improvements in customer service; and, eventually to higher profits and larger market share. Material handling adds to the product cost but contributes nothing to the value added of the products.

Design of material handling systems play a critical role in just-in-time (JIT) manufacturing. Under JIT, production is done in small lots so that production lead-times are reduced and inventory holding costs are minimized, requiring the frequent conveyance of material. Thus, successful implementation of JIT needs a fast and reliable material handling system as a prerequisite. A major, related development with great impact on the material handling process has been the extensive implementation of Total Quality Management (TQM) plans.

Production lot-sizing decisions have a direct impact on the assignment of storage space to different items (products) and consequently on the material handling costs. Therefore, lot sizing decisions must take into account not only setup and inventory carrying costs but also warehouse and material handling costs. In other words, production lot sizing, warehouse storage assignment, and material handling equipment decisions must be made simultaneously.

Also, in a flexible manufacturing environment, where batches of products may have several possible alternative routes, the choice of routing-mix can have a significant effect on shop throughput and work-in-process inventory. However, for such a system to be efficient, an appropriate material handling system needs to be designed. This design issue is especially important when expensive machines are being used. Major waste can be caused by a material handling system that is inappropriate and becomes a bottleneck.

Finally, it should be recognized that (computer-aided) facility layout determines the overall pattern of material flow within the plant and, therefore, has a significant impact on the material handling activities and costs. It is estimated that effective facilities planning and layout can reduce material handling costs by at least 10% to 30% (Tompkins et al. 1996). However, an effective layout requires an effective material handling system. Therefore, it is critical that these decisions are made simultaneously.

Material Handling Equipment

There are several ways of classifying material handling equipment: (1) type of control (operator controlled vs. automated); (2) where the equipment works (on the floor vs. suspended overhead); (3) travel path (fixed vs. flexible). The fixed vs. flexible travel path classification is used here as in Barger (1987). Flexible path equipment can be moved along any route and in general is operator-controlled. Trucks are a common mode of operations. There are several types of trucks depending on the type of handling that is needed, and the following are the most common:

Counterbalanced fork trucks — used both for storage at heights of 20 feet or more, as well as for fast transportation);

Narrow-aisle trucks — mainly used for storage applications;

Walkie Pallet trucks — mainly used for transportation over short hauls; and

Manual trucks — mainly used for short hauls and auxiliary services.

There are three important types of fixed-path equipment:

Conveyors — Conveyors are one of the largest families of material handling equipment. They can be classified based on the load-carrying surface involved: roller, belt, wheel, slat, carrier chain; or on the position of the conveyor: on-floor or overhead;

Automatic Guided Vehicles (**AGVs**) — these are electric vehicles with on-board sensors that enable them to automatically track along a guide path which can be an electrified guide wire or a strip of (reflective) paint or tape on the floor. The AGVs follow their designated path using their sensors to detect the electromagnetic field generated by the electric wire or to optically detect the path marked on the floor. AGVs can transport materials between any two points connected by a guide path — without human intervention. Most of today's AGVs are capable of loading and unloading materials automatically. Most applications of AGVs are for load transportation, however, they could also be used in flexible assembly operations to carry the product being assembled through the various stages of assembly. While AGVs have traditionally been fixed path vehicles, advances in technology permit them to make short deviations from their guide path. Such flexibility may considerably increase their usefulness; and

Hoists, Monorails, and Cranes — Hoists are a basic type of overhead lifting equipment and can be suspended from a rail, track, crane bridge or beam.

A hoist consists of a hook, a rope or chain used for lifting, and a container for the rope/chain. Monorails consist of individual wheeled trolleys that can move along an overhead track. The trolleys may be either powered or non-powered. Cranes have traditionally found wide application in overhead handling of materials, especially where the loads are heavy. Besides the overhead type, there are types of cranes that are wall or floor mounted, portable ones and so on. Types such as stacker cranes are useful in warehouse operations.

Interaction with Automated Storage and Retrieval Systems (AS/RS)

AS/RS consist of high-density storage spaces, computer-controlled handling and storage equipment (operated with minimal human assistance) and may be connected to the rest of the material handling system via some conveying devices such as conveyors and AGVs. Several types of AS/RS are available including: Unit Load, Miniload, Man-On-Board, Deep Lane and Carousels. The AS/RS systems help achieve very efficient placement and retrieval of materials, better inventory control, improved floor space utilization, and production scheduling efficiency. They also provide greater inventory accountability and reduce supervision requirements. Normally, stacker cranes that can move both horizontally and vertically at the same time are used for material handling. Typically, a crane operates in a single aisle, but can be moved between aisles (Rosenblatt et al. 1993). Items to be stored or retrieved are brought to/picked from the AS/RS by a conveyor or an AGV. Such integration can be used to automate material handling throughout the plant and warehouse. A great deal of research has been done on scheduling jobs and assigning storage space in the AS/RS (Hausman et al. 1976).

Issues in Material Handling System Design

Unit load concept — Traditional wisdom is that materials should be handled in the most efficient, maximum size using mechanical means to reduce the number of moves needed for a given amount of material. While reducing the number of trips required is a good objective, the drawback of this approach is that it tends to encourage the acceptance of large production lots, large material handling equipment, and large space requirements. Small unit loads allow for more responsive, less expensive, and less consuming material handling systems. Also, the trend toward continuous manufacturing flow processes and the strong drive for automation necessitate the use of smaller unit loads (Apple and Rickles 1987).

Container size and standardization — This is an issue related to the unit load concept. Container size has an obvious correlation with the size of unit load. Hence, not surprisingly, the current trend is to employ smaller containers. The benefits of smaller containers include compact and more efficient workstations, improved scheduling flexibility due to smaller transfer batch size, smaller staging areas, and lighter duty handling systems. Another consideration that strongly influences the optimal container size is the range of items served by one container. In warehouse operations, unless items vary widely in their physical characteristics, the cost of employing two or more container sizes is almost always higher than in the one-size case (Roll et al. 1989). Use of standard containers eliminates the need for container exchanges between operation sites.

Capacity of the system or number of pieces of equipment — The margins in the design of material handling system require a careful examination of the relative costs of acquiring and maintaining of work centers and handling equipment. In the design of the material handling system for an expensive job shop, enough excess capacity should be provided so that the handling system never becomes the bottleneck.

OR Models in Material Handling

Operations Research (OR) tools have been applied to model and study a variety of problems in the area of material handling. One example, dealing with the initial design phase of material handling, used a graph-theoretic modeling framework (Kouvelis and Lee 1990). Other examples include conveyor systems problems using queueing theory, and transfer lines where dynamic programming techniques were applied. Most of the theoretical work has focused on AGVs and AS/RS. The design and control of AGVs are

extremely complex tasks. The design decisions include determining the optimal number of AGVs (Maxwell and Muckstadt 1982), as well as determining the optimal flow paths (Kim and Tanchoco 1993). Factors to be considered in the design decisions include hardware considerations, impacts on facilities layout, material procurement policy, and production policy. Resulting problems tend to be intractable for any realistic scenario, and hence, heuristics and simulation are the most used techniques in addressing design issues. Control problems including dispatching and routing tasks require real time decisions, making it difficult to obtain optimal solutions. Researchers have attempted to solve simplified problems, for example, by examining static versions instead of dynamic systems (Han and McGinnis 1989), and using simple single-loop layouts (Egbelu 1993).

In the study of warehousing in general, and AS/RS in particular, many different measures of effectiveness of warehouse designs have been considered. The most common ones are throughput as measured by the number of orders handled per day, average travel time of a crane per single/dual command, and average waiting time per customer/order (Hausman et al. 1976). Researchers have considered either simulation or optimization models, usually of the nonlinear integer form, to solve these problems. Yet others have combined optimization and simulation techniques to obtain solutions that are both cost effective and operationally feasible (reasonable service time) (Rosenblatt et al. 1993).

Since factories are increasingly automated, numerical control of machine tools and flexible manufacturing systems is common. Material handling systems frequently involve the use of robots. In the absence of an effective material handling system, an automated factory would be reduced to a set of islands of automation. In the integrated and fiercely competitive global economy, material handling systems play a crucial role in the battle to cut costs and improve productivity and service levels.

See

- ▶ Facilities Layout
- ▶ Flexible Manufacturing Systems
- ▶ Integer and Combinatorial Optimization
- ▶ Inventory Modeling
- ▶ Job Shop Scheduling
- ▶ Just-in-Time (JIT) Manufacturing
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Total Quality Management

References

Apple, J. M., & Rickles, H. M. (1987). Material handling and storage. In J. A. White (Ed.), *Production handbook*. New York: Wiley.

Barger, B. F. (1987). Materials handling equipment. In J. A. White (Ed.), *Production handbook*. New York: Wiley.

Egbelu, P. J. (1993). Positioning of automated guided vehicles in a loop layout to improve response time. *European Journal of Operational Research, 71*, 32–44.

Han, M. H., & McGinnis, L. F. (1989). Control of material handling transporter in automated manufacturing. *IIE Transactions, 21*, 184–190.

Hausman, W. H., Schwarz, L. B., & Graves, S. C. (1976). Optimal assignment in automatic warehousing systems. *Management Science, 22*, 629–638.

Kim, K. H., & Tanchoco, J. M. A. (1993). Economical design of material flow paths. *International Journal of Production Research, 31*, 1387–1407.

Kouvelis, P., & Lee, H. L. (1990). The material handling systems design of integrated manufacturing system. *Annals of Operations Research, 26*, 379–396.

Maxwell, W. L., & Muckstadt, J. A. (1982). Design of automated guided vehicle systems. *IIE Transactions, 14*, 114–124.

Mulcahy, D. (1999). *Materials handling handbook*. New York: McGraw-Hill.

Roll, Y., Rosenblatt, M. J., & Kadosh, D. (1989). Determining the size of a warehouse container. *International Journal of Production Research, 27*, 1693–1704.

Rosaler, R. C., & Rice, J. O. (Eds.). (1994). *Standard handbook of plant engineering* (2nd ed.). New York: McGraw-Hill.

Rosenblatt, M. J., Roll, Y., & Zyser, V. (1993). A combined optimization and simulation approach to designing automated storage/retrieval systems. *IIE Transactions, 25*, 40–50.

Tompkins, J. A., White, J. A., Bozer, Y. A., Frazelle, E. H., Tanchoco, J. M. A., & Trevino, J. (1996). *Facilities planning* (2nd ed.). New York: Wiley.

Material Requirements Planning

A material requirements planning (MRP) system is a collection of logical procedures for managing, at the most detailed level, inventories of component assemblies, subassemblies, parts and raw materials in a manufacturing environment. It is an information system and simulation tool that generates proposals for production schedules that managers can evaluate in terms of their feasibility and cost effectiveness.

See

- ▶ Hierarchical Production Planning
- ▶ Production Management

Mathematical Model

A mathematical description of (usually) a real-world problem. In operations research/management science, mathematical models take on varied forms (e.g., linear programming, queueing, Markovian systems), many of which can be applied across application areas. The basic OR/MS mathematical model can be described as the decision problem of finding the maximum (or minimum) of a measure of effectiveness (objective function) $E = F(X, Y)$, where X represents the set of possible solutions (alternative decisions) and Y the given conditions of the problem. Although a rather simple model in its concept, especially since it involves the optimization of a single objective, this mathematical decision model underlies most of the problems that have been successfully formulated and solved by OR/MS methodologies.

See

- ▶ Decision Problem
- ▶ Deterministic Model
- ▶ Stochastic Model

Mathematical Optimization Society

The Mathematical Optimization Society (MOS) is an international organization dedicated to the support and development of the application, computational methods, and theory of mathematical optimization. The society sponsors the triennial International Symposium on Mathematical Optimization and other meetings throughout the world. Until 2010, its name was the Mathematical Programming Society (MPS), which was founded in 1973.

Mathematical Programming

Mathematical programming is a major discipline in operations research/management science and, in general, is the study of how one optimizes the use and allocation of limited resources. Here the programming refers to the development of a plan or procedure for dealing with the problem. It is considered a branch of applied mathematics as it deals with the theoretical and computational aspects of finding the maximum (minimum) of a function $f(\boldsymbol{x})$ subject to a set of constraints of the form $g_i(\boldsymbol{x}) \leq b_i$. The linear-programming model is the prime example of such a problem.

Mathematical-Programming Problem

A constrained optimization problem usually stated as Minimize (Maximize) $f(\boldsymbol{x})$ subject to $g_i(\boldsymbol{x}) \leq 0$, $i = 1,\ldots, m$. Depending on the form of the objective function $f(\boldsymbol{x})$ and the constraints $g_i(\boldsymbol{x})$ the problem will have special properties and associated algorithms.

See

- ▶ Convex-Programming Problem
- ▶ Fractional Programming
- ▶ Geometric Programming
- ▶ Integer and Combinatorial Optimization
- ▶ Integer-Programming Problem
- ▶ Linear Programming
- ▶ Nonlinear Programming
- ▶ Quadratic Programming
- ▶ Separable-Programming Problem

Mathematical-Programming System (MPS)

An integrated set of computer programs that are designed to solve a range of mathematical-programming problems is often referred to as a mathematical-programming system (MPS). Such systems solve linear programs, usually by some form of the simplex method, and often

have the capability to handle integer-variable problems and other nonlinear problems such as quadratic-programming problems. To be effective, an MPS must have procedures for input data handling, matrix generation of the constraints, reliable optimization, user and automated control of the computation, sensitivity analysis of the solution, solution restart, and output reports.

Matrices and Matrix Algebra

Alan Tucker
The State University of New York at Stony Brook, Stony Brook, NY, USA

Introduction

A matrix is an $m \times n$ array of numbers, typically displayed as

$$\boldsymbol{A} = \begin{bmatrix} 4 & 3 & 8 \\ 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix},$$

where the entry in row i and column j is denoted as a_{ij}. Symbolically, $\boldsymbol{A} = (a_{ij})$, for $i = 1,\ldots, m$ and $j = 1,\ldots, n$. A vector is a one-dimensional array, either a row or a column. A column vector is an $m \times 1$ matrix, while a row vector is a $1 \times n$ matrix. For a matrix $\boldsymbol{A}$, its ith row vector is usually denoted by $\boldsymbol{a}_i'$ and its jth column by $\boldsymbol{a}_j$. Thus an $m \times n$ matrix can be decomposed into a set of m row n-vectors or a set of n column m-vectors. Matrices are a natural generalization of single numbers, or scalars. They arise directly or indirectly in most problems in operations research and management science.

The word matrix in Latin means womb. The term was introduced by J.J. Sylvester in 1848 to describe an array of numbers that could be used to generate (give birth to) a variety of determinants. A few years later, Cayley introduced matrix multiplication and the basic theory of matrix algebra quickly followed. A more general theory of linear algebra and linear transformations pushed matrices into the background until the 1940s and the advent of digital computers. During the 1940s, Alan Turing, father of computer science, introduced the LU decomposition and John von Neumann, father of the digital computer, working with Herman Goldstine, started the development of numerical matrix algebra and introduced the condition number of a matrix. Curiously, at the same time Cayley and Sylvester were developing matrix algebra, another Englishman, Charles Babbage, was building his analytical engine, the forerunner of digital computers, which are critical to the use of modern matrix models.

Basic Operations and Laws of Matrix Algebra

The language for manipulating matrices is matrix algebra. Matrix algebra is a multivariable extension of single-variable algebra. The basic building block for matrix algebra is the scalar product. The scalar product $\boldsymbol{a} \cdot \boldsymbol{b}$ of $\boldsymbol{a}$ and $\boldsymbol{b}$ is a single number (a scalar) equal to the sum of the products $a_i\ b_i$, i.e., $\boldsymbol{a} \cdot \boldsymbol{b} = \sum_{i=1}^{n} a_i b_i$, where both vectors have the same dimension n. Observe that the scalar product is a linear combination of the entries in vector $\boldsymbol{a}$ and also a linear combination of the entries of vector $\boldsymbol{b}$.

The product of an $m \times n$ matrix $\boldsymbol{A}$ and a column n-vector $\boldsymbol{b}$ is a column vector of scalar products $\boldsymbol{a}_i' \cdot \boldsymbol{b}$, of the rows $\boldsymbol{a}_i'$ of $\boldsymbol{A}$ with $\boldsymbol{b}$. For example, if

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$$

is a 2×3 matrix and

$$\boldsymbol{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

is a column 3-vector, then

$$\boldsymbol{Ab} = \begin{bmatrix} \boldsymbol{a_1'} \cdot \boldsymbol{b} \\ \boldsymbol{a_2'} \cdot \boldsymbol{b} \end{bmatrix} = \begin{bmatrix} a_{11}b_1 + a_{12}b_2 + a_{13}b_3 \\ a_{21}b_1 + a_{22}b_2 + a_{23}b_3 \end{bmatrix},$$

so that $\boldsymbol{Ab}$ is a linear combination of $\boldsymbol{A}$. Moreover, for any scalar numbers r, q, any $m \times n$ matrix $\boldsymbol{A}$, and any column n-vectors $\boldsymbol{b}$, $\boldsymbol{c}$:

$$\boldsymbol{A}(r\boldsymbol{b} + q\boldsymbol{c}) = r\boldsymbol{A}\boldsymbol{b} + q\boldsymbol{A}\boldsymbol{c}.$$

The product of a row m-vector $\boldsymbol{c}$ and an $m \times n$ matrix $\boldsymbol{A}$ is a row vector of scalar products $\boldsymbol{c} \cdot \boldsymbol{a}_j$, of $\boldsymbol{c}$ with the columns $\boldsymbol{a}_j$ of $\boldsymbol{A}$. For example, if

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix},$$

is a 2×3 matrix and $\boldsymbol{c} = [c_1, c_2]$ is a row 2-vector, then

$$\begin{aligned} \mathbf{cA} &= [\mathbf{c} \cdot \mathbf{a_1}, \mathbf{c} \cdot \mathbf{a_2}, \mathbf{c} \cdot \mathbf{a_3}] \\ &= [a_{11}c_1 + a_{21}c_2, a_{12}c_1 + a_{22}c_2, a_{13}c_1 + a_{23}c_2]. \end{aligned}$$

If $\boldsymbol{A}$ is an $m \times r$ matrix and $\boldsymbol{B}$ is an $r \times n$ matrix, then the matrix product $\boldsymbol{AB}$ is an $m \times n$ matrix obtained by forming the scalar product of each row $\boldsymbol{a}_i'$ in $\boldsymbol{A}$ with each column $\boldsymbol{b}_j$ in $\boldsymbol{B}$. That is, the (i, j)th entry in $\boldsymbol{AB}$ is $\boldsymbol{a}_i' \cdot \boldsymbol{b}_j$. Column j of $\boldsymbol{AB}$ is the matrix–vector product $\boldsymbol{A}b_j$ and each column of $\boldsymbol{AB}$ is a linear combination of the columns of $\boldsymbol{A}$. Row i of $\boldsymbol{AB}$ is vector–matrix product $\boldsymbol{a}_i'\boldsymbol{B}$ and each row of $\boldsymbol{AB}$ is a linear combination of the rows of $\boldsymbol{B}$. The matrix–vector product $\boldsymbol{Ab}$ is a special case of the matrix-matrix product in which the second matrix has just one column; the analogous statement holds for the vector–matrix product $\boldsymbol{bA}$.

Matrix multiplication is not normally commutative. Otherwise it obeys all the standard laws of scalar multiplication.

Associative Law. *Matrix addition and multiplication are associative*:$(\boldsymbol{A} + \boldsymbol{B}) + \boldsymbol{C} = \boldsymbol{A} + (\boldsymbol{B} + \boldsymbol{C})$ *and* $(\boldsymbol{AB})\boldsymbol{C} = \boldsymbol{A}(\boldsymbol{BC})$.

Commutative Law. *Matrix addition is commutative:* $\boldsymbol{A} + \boldsymbol{B} = \boldsymbol{B} + \boldsymbol{A}$. *Matrix multiplication is not commutative* (except in special cases): $\boldsymbol{AB} \neq \boldsymbol{BA}$.

Distributive Law. $\boldsymbol{A}(\boldsymbol{B} + \boldsymbol{C}) = \boldsymbol{AB} + \boldsymbol{AC}$ *and* $(\boldsymbol{B} + \boldsymbol{C})\boldsymbol{A} = \boldsymbol{BA} + \boldsymbol{CA}$.

Law of Scalar Factoring. $r(\boldsymbol{AB}) = (r\boldsymbol{A})\boldsymbol{B} = \boldsymbol{A}(r\boldsymbol{B})$.

For $n \times n$ matrices $\boldsymbol{A}$, there is an identity matrix $\boldsymbol{I}$ with ones on the main diagonal and zeros elsewhere, with the property that $\boldsymbol{AI} = \boldsymbol{IA} = \boldsymbol{A}$. Furthermore, the transpose of an $m \times n$ matrix $\boldsymbol{A}$, denoted by $\boldsymbol{A}^T$, is an $n \times m$ matrix such that the rows of $\boldsymbol{A}$ are the columns of $\boldsymbol{A}^T$.

If matrices are partitioned into submatrices in a regular fashion, say, a 4×4 matrix $\boldsymbol{A}$ is partitioned into four 2×2 submatrices,

$$\boldsymbol{A} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix},$$

and a 4×4 matrix $\boldsymbol{B}$ is similarly partitioned, then the matrix product $\boldsymbol{AB}$ can be computed in terms of the partitioned submatrices:

$$\mathbf{AB} = \begin{bmatrix} \mathbf{A_{11}B_{11}} + \mathbf{A_{12}B_{21}} & \mathbf{A_{11}B_{12}} + \mathbf{A_{12}B_{22}} \\ \mathbf{A_{21}B_{11}} + \mathbf{A_{22}B_{21}} & \mathbf{A_{21}B_{12}} + \mathbf{A_{22}B_{22}} \end{bmatrix}.$$

Solving Systems of Linear Equations

Matrices are intimately tied to linear systems of equations. For example, the system of linear equations

$$\begin{aligned} 4x_1 + 2x_2 + 2x_3 &= 100 \\ 2x_1 + 5x_2 + 2x_3 &= 200 \\ 1x_1 + 3x_2 + 5x_3 &= 300 \end{aligned} \tag{1}$$

can be written as

$$\boldsymbol{Ax} = \boldsymbol{b}, \text{ where } \boldsymbol{A} = \begin{bmatrix} 4 & 2 & 2 \\ 2 & 5 & 2 \\ 1 & 3 & 5 \end{bmatrix}, \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} 100 \\ 200 \\ 300 \end{bmatrix}. \tag{2}$$

Essentially, the only way to solve an algebraic system with more than one variable is by solving a system of linear equations. For example, nonlinear systems must be recast as linear systems to be numerically solved. Since operations research and management science is concerned with complex problems involving large numbers of variables, matrix systems are pervasive in OR/MS.

Observe that the system of equations given by (1) can be approached from the row point of view as a set of simultaneous linear equations and solved by row operations using Gaussian elimination or Gauss-Jordan elimination. The result of elimination will be either no solution, a unique solution or an infinite number of solutions. In linear programming, one typically wants to find a vector $\boldsymbol{x}$ maximizing or

minimizing a linear objective function $\boldsymbol{c} \cdot \boldsymbol{x}$ subject to a system $\boldsymbol{Ax} = \boldsymbol{b}$ of linear constraints. The simplex method finds an optimal solution by a sequence of pivots on the augmented matrix $[\boldsymbol{Ab}]$. A pivot on non-zero entry (i, j) consists of a collection of row operations (multiplying a row by a scalar or subtracting a multiple of one row from another row) producing a transformed augmented matrix $[\boldsymbol{A}' \ \boldsymbol{b}']$ in which entry (i, j) equals 1 and all other entries in the jth column are 0. The pivot step can be accomplished by premultiplying $\boldsymbol{A}$ by a pivot matrix $\boldsymbol{P}$, which is an identity matrix with a modified ith column.

The system of equations given by (1) can also be approached from the column point of view as the following vector equation:

$$x_1 \begin{bmatrix} 4 \\ 2 \\ 1 \end{bmatrix} + x_2 \begin{bmatrix} 2 \\ 5 \\ 3 \end{bmatrix} + x_3 \begin{bmatrix} 2 \\ 2 \\ 5 \end{bmatrix} = \begin{bmatrix} 100 \\ 200 \\ 300 \end{bmatrix}. \qquad (3)$$

Writing the system as (3) raises questions such as which right-hand side vectors $\boldsymbol{b}$ are expressible as linear combinations of the columns of $\boldsymbol{A}$? The set of such $\boldsymbol{b}$ vectors is called the range of the matrix $\boldsymbol{A}$. For a square matrix, the system $\boldsymbol{Ax} = \boldsymbol{b}$ will have a unique solution if and only if no column vector of $\boldsymbol{A}$ can be written as a linear combination of other columns of $\boldsymbol{A}$, or equivalently, if and only if $\boldsymbol{x} = \boldsymbol{0}$ is the only solution to $\boldsymbol{Ax} = \boldsymbol{0}$, where $\boldsymbol{0}$ denotes a vector of all zeroes. When this condition holds, the columns are said to be linearly independent. When $\boldsymbol{Ax} = \boldsymbol{0}$ has non-zero solutions (whether $\boldsymbol{A}$ is square or not), the set of such nonzero solutions is called the kernel of $\boldsymbol{A}$. Kernels, ranges and linear independence are the building blocks of the theory of linear algebra. This theory plays an important role in the uses of matrices in OR/MS. For example, if $\boldsymbol{x}^*$ is a solution to $\boldsymbol{Ax} = \boldsymbol{b}$ and $\boldsymbol{x}^{\mathrm{o}}$ is in the kernel of $\boldsymbol{A}$ (i.e., $\boldsymbol{Ax}^{\mathrm{o}} = \boldsymbol{0}$), then $\boldsymbol{x}^* + \boldsymbol{x}^{\mathrm{o}}$ is also a solution of $\boldsymbol{Ax} + \boldsymbol{b}$, since $\boldsymbol{A}(\boldsymbol{x}^* + \boldsymbol{x}^{\mathrm{o}}) = \boldsymbol{Ax}^* + \boldsymbol{Ax}^{\mathrm{o}} = \boldsymbol{b} + \boldsymbol{0} = \boldsymbol{b}$, and one can show that all solutions to $\boldsymbol{Ax} = \boldsymbol{b}$ can be written in the form of a particular solution $\boldsymbol{x}^*$ plus some kernel vector $\boldsymbol{x}^{\mathrm{o}}$. In a linear program to maximize or minimize $\boldsymbol{c} \cdot \boldsymbol{x}$ subject to $\boldsymbol{Ax} = \boldsymbol{b}$, once one finds one solution $\boldsymbol{x}^*$ to $\boldsymbol{Ax} = \boldsymbol{b}$, improved solutions will be obtained by adding appropriate kernel vectors to $\boldsymbol{x}^*$.

Matrix Inverse

The inverse $\boldsymbol{A}^{-1}$ of a square matrix $\boldsymbol{A}$ has the property that $\boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{AA}^{-1} = \boldsymbol{I}$. The inverse can be used to solve $\boldsymbol{Ax} = \boldsymbol{b}$ as follows: $\boldsymbol{Ax} = \boldsymbol{b} \Rightarrow \boldsymbol{A}^{-1}(\boldsymbol{Ax}) = \boldsymbol{A}^{-1}\boldsymbol{b}$, but $\boldsymbol{A}^{-1}(\boldsymbol{Ax}) = (\boldsymbol{A}^{-1}\boldsymbol{A})\boldsymbol{x} = (\boldsymbol{I})\boldsymbol{x} = \boldsymbol{x}$. Thus $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$.

The square matrix $\boldsymbol{A}$ has an inverse if any of the following equivalent statements hold:

1. For all $\boldsymbol{b}$, $\boldsymbol{Ax} = \boldsymbol{b}$ has a unique solution;
2. The columns of $\boldsymbol{A}$ are linearly independent;
3. The rows of $\boldsymbol{A}$ are linearly independent.

The matrix $\boldsymbol{A}^{-1}$ is found by solving a system of equations as follows. The product $\boldsymbol{AA}^{-1} = \boldsymbol{I}$ implies that if $\boldsymbol{x}_j$ is the jth column of $\boldsymbol{A}^{-1}$ and $\boldsymbol{i}_j$ is the jth column of $\boldsymbol{I}$ ($\boldsymbol{i}_j$ has 1 in the jth entry and zeroes elsewhere), then $\boldsymbol{x}_j$ is the solution to the matrix system $\boldsymbol{Ax}_j = \boldsymbol{i}_j$. An impressive aspect of matrix algebra is that even when a matrix system $\boldsymbol{Ax} = \boldsymbol{b}$ has no solution, i.e., in (3) no linear combination of the columns of $\boldsymbol{A}$ equals $\boldsymbol{b}$, there is still a "solution" $\boldsymbol{y}$ in the sense of a linear combination $\boldsymbol{Ay}$ of the columns of $\boldsymbol{A}$ that is as close as possible to $\boldsymbol{b}$, i.e., the Euclidean distance in n-dimensional space between the vectors $\boldsymbol{Ay}$ and $\boldsymbol{b}$ is minimized. There is even an inverse-like matrix $\boldsymbol{A}^*$, called the pseudoinverse or generalized inverse, such that $\boldsymbol{y} = \boldsymbol{A}^*\boldsymbol{b}$. The matrix $\boldsymbol{A}^*$ is given by the matrix formula $\boldsymbol{A}^* = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$, where $\boldsymbol{A}^T$ is the transpose of $\boldsymbol{A}$, obtained by interchanging rows and columns.

Eigenvalues and Eigenvectors

A standard form of a dynamic linear model is $\boldsymbol{p}' = \boldsymbol{Ap}$, where $\boldsymbol{A}$ is an $n \times n$ matrix and $\boldsymbol{p}$ is a n-column vector of populations or probabilities (in the case of probabilities, it is the convention to use row vectors: $\boldsymbol{p}' = \boldsymbol{pA}$). For some special vectors $\boldsymbol{e}$, called eigenvectors, $\boldsymbol{Ae} = \lambda\boldsymbol{e}$, where λ is a scalar called an eigenvalue. That is, premultiplying $\boldsymbol{e}$ by $\boldsymbol{A}$ has the effect of multiplying $\boldsymbol{e}$ by a scalar. It follows that $\boldsymbol{A}^n\boldsymbol{e} = \lambda^n\boldsymbol{e}$. This special situation is very valuable because it is obviously much easier to compute $\lambda^n\boldsymbol{e}$ than $\boldsymbol{A}^n\boldsymbol{e}$.

Most $n \times n$ matrices have n different (linearly independent) eigenvectors. If the vector $\boldsymbol{p}$ as a linear combination $\boldsymbol{p} = a\,\boldsymbol{e}_1 + b\,\boldsymbol{e}_2$ of, say, two eigenvectors $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$, with associated eigenvalues λ_1, λ_2, then by the linearity of matrix–vector products, $\boldsymbol{Ap}$ and $\boldsymbol{A}^2\boldsymbol{p}$ can be calculated as

$$\boldsymbol{Ap} = \boldsymbol{A}(a\boldsymbol{e}_1 + b\boldsymbol{e}_2) = a\boldsymbol{Ae}_1 + b\boldsymbol{Ae}_2 = a\lambda_1\boldsymbol{e}_1 + b\lambda_2\boldsymbol{e}_2$$

and

$$\begin{aligned}\boldsymbol{A}^2\boldsymbol{p} &= \boldsymbol{A}^2(a\boldsymbol{e}_1 + b\boldsymbol{e}_2) = a\boldsymbol{A}^2\boldsymbol{e}_1 + b\boldsymbol{A}^2\boldsymbol{e}_2 \\ &= a\lambda_1^2\boldsymbol{e}_1 + b\lambda_2^2\boldsymbol{e}_2.\end{aligned}$$

More generally,

$$\begin{aligned}\boldsymbol{A}^k\boldsymbol{p} &= \boldsymbol{A}^k(a\boldsymbol{e}_1 + b\boldsymbol{e}_2) = a\boldsymbol{A}^\kappa\boldsymbol{e}_1 + b\boldsymbol{A}^k\boldsymbol{e}_2 \\ &= a\lambda_1^k\boldsymbol{e}_1 + b\lambda_2^k\boldsymbol{e}_2.\end{aligned}$$

If $|\lambda_1| > |\lambda_i|$, for $i \geq 2$, then for large k, λ^{k_1} will become much larger in absolute value than the other λ^{k_j}, and so $\boldsymbol{A^k p}$ approaches a multiple of the eigenvector associated with the eigenvalue of largest absolute value. For ergodic Markov chains, this largest eigenvalue is 1 and the Markov chain converges to a steady-state probability $\boldsymbol{p}^*$ such that $\boldsymbol{p}^* = \boldsymbol{p}^*\boldsymbol{A}$.

Matrix Norms

The norm $|\boldsymbol{v}|$ of a vector $\boldsymbol{v}$ is a scalar value that is nonnegative, satisfies scalar factoring, i.e., $|r\ \boldsymbol{v}| = r\ |\boldsymbol{v}|$, and the triangle inequality, i.e., $|\boldsymbol{u} + \boldsymbol{v}| \leq |\boldsymbol{u}| + |\boldsymbol{v}|$. There are three common norms used for vectors:

1. The Euclidean, or l_2, norm of $\boldsymbol{v} = [v_1, v_2, \ldots, v_n]$ is defined as $|\boldsymbol{v}|_e = \sqrt{\boldsymbol{v}_1^2 + \boldsymbol{v}_2^2 + \cdots + \boldsymbol{v}_n^2}$.
2. The sum, or l_1, norm of $\boldsymbol{v} = [v_1, v_2, \ldots, v_n]$ is defined $|\boldsymbol{v}|_s = |\boldsymbol{v}_1| + |\boldsymbol{v}_2| + \cdots + |\boldsymbol{v}_n|$.
3. The max, or l_∞, norm of $\boldsymbol{v} = [v_1, v_2, \ldots, v_n]$ is $|\boldsymbol{v}|_m = \max\{|\boldsymbol{v}_1|, |\boldsymbol{v}_2|, \ldots, |\boldsymbol{v}_n|\}$.

The matrix norm $||\boldsymbol{A}||$ is the (smallest) bound such that $|\boldsymbol{Ax}| \leq ||\boldsymbol{A}||\ |\boldsymbol{x}|$, for all $\boldsymbol{x}$. Thus

$$||\boldsymbol{A}|| = \max_{\boldsymbol{x}\neq\boldsymbol{0}} \frac{|\boldsymbol{Ax}|}{|\boldsymbol{x}|}. \tag{4}$$

It follows that $|\boldsymbol{A}^k\boldsymbol{x}| \leq ||\boldsymbol{A}||^k\ |\boldsymbol{x}|$.

The Euclidean, sum, and max norms of the matrix are defined by using the Euclidean, sum, and max vector norms, respectively, in (4). When $\boldsymbol{A}$ is a square, symmetric matrix ($a_{ij} = a_{ji}$), the Euclidean norm $||\boldsymbol{A}||_e$ equals the absolute value of the largest eigenvalue of $\boldsymbol{A}$. When $\boldsymbol{A}$ is not symmetric, $||\boldsymbol{A}||_e$ equals the positive square root of the largest eigenvalue of $\boldsymbol{A}^T\boldsymbol{A}$. The sum and max norms of $\boldsymbol{A}$ are very simple to find and for this reason are often preferred over the Euclidean norm: $||\boldsymbol{A}||_s = \max_j\{|\boldsymbol{A}_j|_s\}$ and $||\boldsymbol{A}||_m = \max_i\{|\boldsymbol{A}_i'|_s\}$, where $\boldsymbol{A}_j$ denotes the jth column of $\boldsymbol{A}$ and $\boldsymbol{A}_i'$ denotes the ith row of $\boldsymbol{A}$. In words, the sum norm of $\boldsymbol{A}$ is the largest column sum (summing absolute values), and the max norm of $\boldsymbol{A}$ is the largest row sum.

Norms have many uses. For example, in a linear growth model $\boldsymbol{p}' = \boldsymbol{Ap}$, the kth iterate $\boldsymbol{p}^{(k)} = \boldsymbol{A}^k\boldsymbol{p}$ is bounded in norm by $|\ \boldsymbol{p}^{(k)}\ | \leq ||\boldsymbol{A}||^k\ |\boldsymbol{p}|$. One can show that if the system of linear equations $\boldsymbol{Ax} = \boldsymbol{b}$ is perturbed by adding a matrix $\boldsymbol{E}$ of errors to $\boldsymbol{A}$, and if $\boldsymbol{x}^*$ is the solution to the original system $\boldsymbol{Ax} = \boldsymbol{b}$ while $\boldsymbol{x}^* + \boldsymbol{e}$ is the solution to $(\boldsymbol{A} + \boldsymbol{E})\boldsymbol{x} = \boldsymbol{b}$, then the relative error $|\boldsymbol{e}|/|\ \boldsymbol{x}^* + \boldsymbol{e}|$ is bounded by a constant $c(\boldsymbol{A})$ times the relative error $||\boldsymbol{E}||/||\boldsymbol{A}||$, i.e., $|\boldsymbol{e}|/|\boldsymbol{x}^* + \boldsymbol{e}| \leq c(\boldsymbol{A})\ ||\boldsymbol{E}||/||\boldsymbol{A}||$. The constant $c(\boldsymbol{A}) = ||\boldsymbol{A}||\ ||\boldsymbol{A}^{-1}||$ and is called the condition number of $\boldsymbol{A}$.

A famous linear input–output model due to Leontief has the form $\boldsymbol{x} = \boldsymbol{Ax} + \boldsymbol{b}$. Here $\boldsymbol{x}$ is a vector of production of various industrial activities, $\boldsymbol{b}$ is a vector of consumer demands for these activities, and $\boldsymbol{A}$ is an inter-industry demand matrix in which entry a_{ij} tells how much of activity i is needed to produce one unit of activity j. Here, $\boldsymbol{Ax}$ is a vector of the input for the different activities needed to produce the output vector $\boldsymbol{x}$. The model $\boldsymbol{x} = \boldsymbol{Ax} + \boldsymbol{b}$ can be shown to have a solution if $||\boldsymbol{A}||_s < 1$, i.e., if the columns sums are all less than one. This condition has the natural economic interpretation that all activities must be profitable, i.e., the value of the inputs to produce a dollar's worth of any activity must be less than one dollar.

Algebraically, $\boldsymbol{x} = \boldsymbol{Ax} + \boldsymbol{b}$ is solved as follows:

$$\begin{aligned}\boldsymbol{x} &= \boldsymbol{Ax} + \boldsymbol{b} \rightarrow \boldsymbol{x} - \boldsymbol{AX} = \boldsymbol{b} \rightarrow (\boldsymbol{I} - \boldsymbol{A})\boldsymbol{x} \\ &= b \rightarrow \boldsymbol{x} = (\boldsymbol{I} - \boldsymbol{A})^{-\boldsymbol{I}}\boldsymbol{b}.\end{aligned}$$

When $||\boldsymbol{A}|| \leq 1$, the geometric series $\boldsymbol{I} + \boldsymbol{A} + \boldsymbol{A}^2 + \boldsymbol{A}^3 + \ldots$, converges to $(\boldsymbol{I} - \boldsymbol{A})^{-1}$, guaranteeing not only the existence of a solution to $\boldsymbol{x} = \boldsymbol{Ax} + \boldsymbol{b}$ but also a solution with nonnegative entries, since when $\boldsymbol{A}$ has nonnegative entries, then all the powers of $\boldsymbol{A}$ will have nonnegative entries implying that $(\boldsymbol{I} - \boldsymbol{A})^{-1}$ has nonnegative entries and hence so does $\boldsymbol{x} = (\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{b}$.

See

- ▶ Analytic Hierarchy Process
- ▶ Gaussian Elimination
- ▶ Gauss-Jordan Elimination Method
- ▶ Linear Programming
- ▶ LU Matrix Decomposition
- ▶ Markov Chains
- ▶ Simplex Method (Algorithm)

References

Lay, D. C. (1993). *Linear algebra and its applications*. Reading, MA: Addison Wesley.
Strang, G. (2009). *Introduction to linear algebra* (4th ed.). Wellesley, MA: Wellesley-Cambridge Press.

Matrix Game

- ▶ Game Theory

Matrix Geometric

When the solution to a stochastic model is (vector) proportional to a geometric distribution whose parameter is a matrix instead of the usual scalar.

See

- ▶ Matrix-Analytic Stochastic Models

Matrix-Analytic Stochastic Models

Marcel F. Neuts
The University of Arizona, Tucson, AZ, USA

Introduction

A rich class of models for queues, dams, inventories, and other stochastic processes has arisen out of matrix/vector generalizations of classical approaches. Three specific examples are presented: matrix-analytic solutions for M/G/1-type queueing problems, matrix-geometric solutions to GI/M/1-type queueing problems, and the Markov arrival process (MAP) generalization of the renewal point process.

Matrix-Analytic M/G/1-Type Queues

The unifying structure that underlies these models is an imbedded Markov renewal process whose transition probability matrix is of the form:

$$\tilde{\boldsymbol{Q}}(x) = \begin{bmatrix} \boldsymbol{B}_0(x) & \boldsymbol{B}_1(x) & \boldsymbol{B}_2(x) & \boldsymbol{B}_3(x) & \boldsymbol{B}_4(x) & \cdots \\ \boldsymbol{C}_0(x) & \boldsymbol{A}_1(x) & \boldsymbol{A}_2(x) & \boldsymbol{A}_3(x) & \boldsymbol{A}_4(x) & \cdots \\ \boldsymbol{0} & \boldsymbol{A}_0(x) & \boldsymbol{A}_1(x) & \boldsymbol{A}_2(x) & \boldsymbol{A}_3(x) & \cdots \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{A}_0(x) & \boldsymbol{A}_1(x) & \boldsymbol{A}_2(x) & \cdots \\ . & . & . & . & . & \cdots \end{bmatrix}$$

where the elements are themselves matrices of probability mass functions. If the matrix

$$\boldsymbol{A} = \sum_{k=0}^{\infty} \boldsymbol{A}_k(\infty)$$

is irreducible and has the invariant probability vector $\boldsymbol{\pi}$, then the Markov renewal process is positive recurrent if and only if some natural moment conditions hold for the coefficient matrices and if

$$\rho = \boldsymbol{\pi} \sum_{k=1}^{\infty} k\boldsymbol{A}_k \boldsymbol{e} < 1 \quad \text{for } \boldsymbol{e} = (1, \ldots, 1)^T.$$

The quantity ρ is the generalized form of the traffic intensity for the elementary queueing models.

The state space is partitioned in levels $\boldsymbol{i}$, which are the sets of m states (i, j), $1 \leq j \leq m$. The crucial object in studying the behavior of the Markov renewal process away from the boundary states in the level $\boldsymbol{0}$ is the fundamental period, the first passage time from a state in $\boldsymbol{i}$ + 1 to a state in $\boldsymbol{i}$. The joint transform matrix $\tilde{\boldsymbol{G}}(z; s)$ of that first passage time, measured in the number of transitions to lower levels (completed services in queueing applications) and in real time, satisfies a nonlinear matrix equation of the form

$$\tilde{\boldsymbol{G}}(z;s) = z\sum_{k=0}^{\infty}\tilde{\boldsymbol{A}}(s)\left[\tilde{\boldsymbol{G}}(z;s)\right]^k.$$

This equation can be analyzed by methods of functional analysis, which leads to many explicit matrix formulas for moments. In terms of the matrix $\tilde{\boldsymbol{G}}(z; s)$, the boundary behavior of the Markov renewal process can be studied in an elementary manner. In queueing applications, the analysis leads to equations for the busy period and the busy cycle. Waiting-time distributions under the first-come, first-served discipline are obtained as first passage time distributions. Extensive generalizations of the Pollaczek-Khinchin integral equation for the classical M/G/1 queue have been obtained (see Neuts 1986b).

Applications of Markov renewal theory lead to a matrix formula for the steady-state probability vector $\boldsymbol{x}_0$ for the states in level $\mathbf{0}$ in the imbedded Markov chain. Next, a stable numerical recurrence due to Ramaswami (1988) permits computation of the steady-state probability vector $\boldsymbol{x}_i$ of the other levels i, $i \geq 1$.

There is an interesting duality between the random walks on the infinite strip of states (i, j), $-\infty < i < \infty$, $1 \leq j \leq m$, that underlie the Markov renewal processes of M/G/1 type and those of GI/M/1-type (which lead to matrix-geometric solutions). That duality is investigated in Asmussen and Ramaswami (1990) and Ramaswami (1990a).

The class of models with an imbedded Markov renewal process of M/G/1-type is very rich. It is useful in the analysis of many queueing models in continuous or discrete time that arise in communications engineering and other applications. In queueing theory, results for a variety of classical models have been extended to versatile input processes and to semi-Markovian services. These generalizations often lead to natural matrix generalizations of familiar formulas. For a discussion of what happens to the M/G/1 model when the input is changed to a Markovian arrival process (MAP — as more precisely presented in a subsequent section), see Lucantoni (1993). A treatment of cycle maxima for the MAP/G/1 queue is found in Asmussen and Perry (1992). A mathematically rigorous discussion of the complex analysis aspects of the models of M/G/1-type is found in Gail, Hantler, and Taylor (1994). Asymptotic results on the tail probabilities of queue length and waiting time distributions are discussed in Abate, Choudhury and Whitt (1994), and Falkenberg (1994).

Matrix-Geometric Solutions

Under ergodicity conditions, discrete-time Markov chains with transition probability matrix $\boldsymbol{P}$ of the form

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{B_0} & \boldsymbol{A_0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ \boldsymbol{B_1} & \boldsymbol{A_1} & \boldsymbol{A_0} & \mathbf{0} & \mathbf{0} & \cdots \\ \boldsymbol{B_2} & \boldsymbol{A_2} & \boldsymbol{A_1} & \boldsymbol{A_0} & \mathbf{0} & \cdots \\ \boldsymbol{B_3} & \boldsymbol{A_3} & \boldsymbol{A_2} & \boldsymbol{A_1} & \boldsymbol{A_0} & \cdots \\ . & . & . & . & . & \cdots \end{bmatrix},$$

where the $\boldsymbol{A}_k$ are $m \times m$ nonnegative matrices summing to a stochastic matrix $\boldsymbol{A}$, and the $\mathbf{B}_k$ are nonnegative matrices such that the row sums of $\boldsymbol{P}$ are one, have an invariant probability vector $\boldsymbol{x}$ of a matrix-geometric form. That is, the unique probability vector $\boldsymbol{x}$ which satisfies $\boldsymbol{xP} = \boldsymbol{x}$, can be partitioned into row vectors $\boldsymbol{x}_i$, $i \geq 0$, which satisfy $\boldsymbol{x}_i = \boldsymbol{x}_0 \boldsymbol{R}^i$. The matrix $\boldsymbol{R}$ is the unique minimal solution to the equation

$$\boldsymbol{R} = \sum_{k=0}^{\infty}\boldsymbol{R}^k\boldsymbol{A}_k,$$

in the set of nonnegative matrices. All eigenvalues of $\boldsymbol{R}$ lie inside the unit disk. The matrix,

$$\boldsymbol{B}[\boldsymbol{R}] = \sum_{k=0}^{\infty}\boldsymbol{R}^k\boldsymbol{B}_k,$$

is an irreducible stochastic matrix. The vector $\boldsymbol{x}_0$ is determined as the unique solution to the equations

$$\begin{cases} \boldsymbol{x_0} = \boldsymbol{x_0}\boldsymbol{B}[\boldsymbol{R}] \\ 1 = \boldsymbol{x_0}(\mathbf{1} - \boldsymbol{R})^{-1}\boldsymbol{e} \end{cases}$$

where $\boldsymbol{e}$ is the column m-vector with all components equal to one. If the matrix $\boldsymbol{A}$ is irreducible and has the invariant probability vector $\boldsymbol{\pi}$, the Markov chain is positive recurrent if and only if

$$\boldsymbol{\pi} \sum_{k=1}^{\infty} k\boldsymbol{A}_k \boldsymbol{e} > 1.$$

Analogous forms of the matrix-geometric theorem hold for Markov chains with a more complicated behavior at the boundary states and for continuous Markov chains with a generator $\boldsymbol{Q}$ of the same structural form. A comprehensive treatment of the basic properties of such Markov chains and a variety of applications is given in Neuts (1981).

This result has found many applications in queueing theory. The subclass where the matrix $\boldsymbol{P}$ or the generator $\boldsymbol{Q}$ are block-tridiagonal are called quasi-birth and death (QBD) processes. These arise naturally as models for many problems in communications engineering and computer performance. The matrix-geometric form of the steady-state probability vector of a suitable imbedded Markov chain leads to explicit matrix formulas for other descriptors of queues, such as the steady-state distributions of waiting times, the distribution of the busy period and others.

In addition to its immediate applications, this construct has also generated much theoretical interest. Its generalization to the operator case was established in Tweedie (1982).

The largest eigenvalue η of the matrix $\boldsymbol{R}$ is important in various asymptotic results. Graphs of η as a function of a parameter of the queue are caudal characteristic curves. Some interesting behavioral features of the queues can be inferred from them (Neuts and Takahashi 1981; Neuts 1986a; Asmussen and Perry 1992). A matrix-exponential form for waiting-time distributions in queueing models was obtained in Sengupta (1989). Its relation to the matrix-geometric theorem was discussed in Ramaswami (1990b). A matrix-analytic treatment, covering all cases of reducibility, of the equation for $\boldsymbol{R}$, is given in Gail, Hantler and Taylor (1994).

The matrix $\boldsymbol{R}$, which is crucial to all applications of the theorem, must be computed by an iterative numerical solution of the nonlinear matrix equation

$$\boldsymbol{R} = \sum_{k=0}^{\infty} \boldsymbol{R}^k \boldsymbol{A}_k$$

A major survey and comparisons of various computational methods is found in Latouche (1993). For the block tri-diagonal case (*QBD*-processes), a particularly efficient algorithm was developed by Latouche and Ramaswami (1993).

Markovian Arrival Processes

The analytic tractability of models with Poisson or Bernoulli input is due to the lack-of-memory property, an extreme case of Markovian simplification. At the expense of performing matrix calculations, more versatile arrival processes can be used in a variety of models. The Markovian arrival process (MAP) is a point process model in which only one of a finite number of phases must be remembered to preserve many of the simplifying Markovian properties. It can be incorporated in many models which remain highly tractable by matrix-analytic methods. The MAP has found many applications in queueing and tele-traffic models to represent bursty arrival streams. Many queueing models for which traditionally Poisson arrivals were assumed are also amenable to analysis with MAP input.

It was first introduced in Neuts (1979), but a more appropriate notation was proposed by David Lucantoni in conjunction with the queueing model discussed in Lucantoni, Meier-Hellstern, and Neuts (1990). Although discrete-time versions of the MAP, as well as processes with group arrivals have been defined, their discussion requires only more elaborate notation than the single-arrival MAP in continuous time described here. Expositions of the basic properties and many examples of the MAP are found in Neuts (1989, 1992) and Lucantoni (1991).

Consider an irreducible infinitesimal generator $\boldsymbol{D}$ of dimension m with stationary probability vector θ. Write $\boldsymbol{D}$ as the sum of matrices $\boldsymbol{D}_0$ and $\boldsymbol{D}_1$, where $\boldsymbol{D}_1$ is nonnegative and $\boldsymbol{D}_0$ has nonnegative off-diagonal elements. The diagonal elements of $\boldsymbol{D}_0$ are strictly negative and $\boldsymbol{D}_0$ is nonsingular. Consider an m-state Markov renewal process $\{(J_n, X_n), n \geq 0\}$ in which each transition epoch has an associated arrival. Its transition probability matrix $\boldsymbol{F}(\cdot)$ is given by

$$\boldsymbol{F}(x) = \int_0^x \exp(\boldsymbol{D}_0 u) du \boldsymbol{D}_1, \quad \text{for } x \geq 0.$$

The most familiar MAPs are the *PH*-renewal process and the Markov-modulated Poisson Process (MMPP). These, respectively, have the pairs of parameter matrices $\boldsymbol{D}_0 = \boldsymbol{T}$, $\boldsymbol{D}_1 = \boldsymbol{T}^{o}\alpha$, where $(\alpha, \boldsymbol{T})$ is the (irreducible) representation of a phase-type distribution and the column vector $\boldsymbol{T}^{o} = -\boldsymbol{T}\boldsymbol{e}$, and $\boldsymbol{D}_0 = \boldsymbol{D} - \boldsymbol{\Lambda}$, $\boldsymbol{D}_1 = \boldsymbol{\Lambda}$, where $\boldsymbol{\Lambda}$ is a diagonal matrix and $\boldsymbol{e}$ is the column m-vector with all components equal to one.

The matrix-analytic tractability of the MAP is a consequence of the matrix-exponential form of the transition probability matrix $\boldsymbol{F}(\cdot)$. It, in turn, follows from the Markov property of the underlying chain with generator $\boldsymbol{D}$, in which certain transitions are labeled as arrivals. A detailed description of that construction is found in Lucantoni (1991).

The initial conditions of the MAP are specified by the initial probability vector $\boldsymbol{\gamma}$ of the underlying Markov chain with generator $\boldsymbol{D}$. Taking $\boldsymbol{\gamma} = \boldsymbol{\theta}$, the stationary probability vector of $\boldsymbol{D}$, leads to the stationary version of the MAP. The rate γ^* of the stationary process is given by $\gamma^* = \boldsymbol{\theta}\boldsymbol{D}_1\boldsymbol{e}$. By choosing $\boldsymbol{\gamma} = (\gamma^*)^{-1}\boldsymbol{\theta}\boldsymbol{D}_1 = \theta_{\text{arr}}$, the time origin is an arbitrary arrival epoch.

Computationally tractable matrix expressions are available for various moments of the MAP. These require little more than the computation of the matrix $\exp(\boldsymbol{D}t)$. A comprehensive discussion of these formulas is found in Neuts and Narayana (1992). For example, the Palm measure, $H(t) = E[N(t) \mid \text{arrival at } t = 0]$, the expected number of arrivals in an interval $(0, t]$ starting from an arbitrary arrival epoch, is given by

$$\boldsymbol{H}(t) = \boldsymbol{\lambda} * t + \theta_{\text{arr}}[\boldsymbol{I} - \exp(\boldsymbol{D}t)](\boldsymbol{e}\boldsymbol{\theta} - \boldsymbol{D})^{-1}\boldsymbol{D}_1\boldsymbol{e}.$$

Other MAPs are constructed by considering selected transitions in Markov chains, by certain random time transformations or random thinning of a given MAP, and by superposition of independent MAPs. Statements and examples of these constructions are found in Neuts (1989, 1992). Specifically, the superposition of two (or more) independent MAPs is again an MAP. If two continuous-time MAPs have the parameter matrices $\{\boldsymbol{D}_k(i)\}$ for $i = 1, 2$, the parameter matrices for their superposition are given by $\boldsymbol{D}_k = \boldsymbol{D}_k(1) \bigotimes \boldsymbol{I} + \boldsymbol{I} \bigotimes \boldsymbol{D}_k(2) = \boldsymbol{D}_k(1) \bigotimes \boldsymbol{D}_k(2)$, for $k \geq 1, 2$, where $\bigotimes$ is the Kronecker pairwise matrix product.

See

- Markov Chains
- Markov Processes
- Matrices and Matrix Algebra
- Phase-Type Probability Distributions
- Queueing Theory

References

Abate, J., Choudhury, G. L., & Whitt, W. (1994). Asymptotics for steady-state tail probabilities in structured Markov queueing models. *Stochastic Models, 10*, 99–143.

Asmussen, S., & Perry, D. (1992). On cycle maxima, first passage problems and extreme value theory for queues. *Stochastic Models, 8*, 421–458.

Asmussen, S., & Ramaswami, V. (1990). Probabilistic interpretation of some duality results for the matrix paradigms in queueing theory. *Stochastic Models, 6*, 715–733.

Falkenberg, E. (1994). On the asymptotic behavior of the stationary distribution of Markov chains of M/G/1-type. *Stochastic Models, 10*, 75–97.

Gail, H. R., Hantler, S. L., & Taylor, B. A. (1994). Solutions of the basic matrix equations for the M/G/1 and G/M/1 Markov chains. *Stochastic Models, 10*, 1–43.

Latouche, G. (1985). An exponential semi-Markov process, with applications to queueing theory. *Stochastic Models, 1*, 137–169.

Latouche, G. (1993). Algorithms for infinite Markov chains with repeating columns. In C. D. Meyer & R. J. Plemmons (Eds.), *Linear algebra, Markov chains and queueing models* (pp. 231–265). New York: Springer-Verlag.

Latouche, G., & Ramaswami, V. (1993). A logarithmic reduction algorithm for quasi-birth-and-death processes. *Journal of Applied Probability, 30*, 650–674.

Lucantoni, D. M. (1991). New results on the single server queue with a batch Markovian arrival process. *Stochastic Models, 7*, 1–46.

Lucantoni, D. M. (1993). The BMAP/G/1 queue: A tutorial. In L. Donatiello & R. Nelson (Eds.), *Models and techniques for performance evaluation of computer and communications systems*. New York: Springer-Verlag.

Lucantoni, D. M., Meier-Hellstern, K. S., & Neuts, M. F. (1990). A single server queue with server vacations and a class of non-renewal arrival processes. *Advances in Applied Probability, 22*, 676–705.

Neuts, M. F. (1979). A versatile Markovian point process. *Journal of Applied Probability, 16*, 764–779.

Neuts, M. F. (1981). *Matrix-geometric solutions in stochastic models: An algorithmic approach*. Baltimore: The Johns Hopkins University Press. Reprinted by Dover Publications, 1994.

Neuts, M. F. (1986a). The caudal characteristic curve of queues. *Advances in Applied Probability, 18*, 221–254.

Neuts, M. F. (1986b). Generalizations of the Pollaczek-Khinchin integral equation in the theory of queues. *Advances in Applied Probability, 18*, 952–990.

Neuts, M. F. (1989). *Structured stochastic matrices of M/G/1 type and their applications*. New York: Marcel Dekker.
Neuts, M. F. (1992). Models based on the Markovian arrival process. *IEEE Transactions on Communications, Special Issue on Teletraffic, E75-B*, 1255–1265.
Neuts, M. F., & Narayana, S. (1992). The first two moment matrices of the counts for the Markovian arrival process. *Stochastic Models, 8*, 459–477.
Neuts, M. F., & Takahashi, Y. (1981). Asymptotic behavior of the stationary distributions in the GI/PH/c queue with heterogeneous servers. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte, 57*, 441–452.
Ramaswami, V. (1988). A stable recursion for the steady state vector in Markov chains of M/G/1 type. *Stochastic Models, 4*, 183–188.
Ramaswami, V. (1990a). A duality theorem for the matrix paradigms in queueing theory. *Stochastic Models, 6*, 151–161.
Ramaswami, V. (1990b). From the matrix-geometric to the matrix-exponential. *Queueing Systems, 6*, 229–260.
Schellhaas, H. (1990). On Ramaswami's algorithm for the computation of the steady state vector in Markov chains of M/G/1-type. *Stochastic Models, 6*, 541–550.
Sengupta, B. (1989). Markov processes whose steady state distribution is matrix-exponential with an application to the GI/PH/1 queue. *Advances in Applied Probability, 21*, 159–180.
Tweedie, R. L. (1982). Operator-geometric stationary distributions for Markov chains with application to queueing models. *Advances in Applied Probability, 14*, 368–391.

MAUT

► Multi-Attribute Utility Theory

Max-Flow Min-Cut Theorem

For a maximum-flow network problem, it can be shown that the maximum flow through the network is equal to the minimum capacity of all the cuts that separate the source (origin) and the sink (destination) nodes, where the capacity of a cut is the sum of the capacities of the arcs in the cut.

See

► Maximum-Flow Network Problem

References

Ford, L. R., & Fulkerson, D. R. (1962). *Flows in networks*. Princeton, NJ: Princeton University Press.

Maximum

A function $f(x)$ is said to have a maximum on a set S when the least upper bound of $f(x)$ on S is assumed by $f(x)$ for some x^0 in S. Thus, $f(x^0) \geq f(x)$ for all x in S.

See

► Global Maximum (Minimum)

Maximum Feasible Solution

► Minimum (Maximum) Feasible Solution

Maximum Matching Problem

Involves finding in a graph a maximal set of links which meet each node at most once.

See

► Matching

Maximum-Flow Network Problem

For a directed, capacitated network with source and sink nodes, the problem is to find the maximum amount of goods (flow) that can be sent from the source to the sink.

See

► Network Optimization

References

Ford, L. R., & Fulkerson, D. R. (1962). *Flows in networks*. Princeton, NJ: Princeton University Press.

MCDM

► Multiple Criteria Decision Making

Measure of Effectiveness (MOE)

In a decision problem, the single objective that is to be optimized is called the measure of effectiveness (MOE). In a linear-programming problem, the MOE is the objective function. In a queueing-theory problem, frequently used MOEs include the expected steady-state queue length and the mean delay in queue.

See

► Mathematical Model

Measure-Valued Differentiation

► Weak Derivatives

Memetic Algorithms

Hybrid metaheuristic evolutionary algorithms (EAs) that combine population-based approaches such as genetic algorithms with local search improvement procedures or individual learning. Also known as Baldwinian EAs, Lamarckian EAs, cultural algorithms or genetic local search. Derived from the word "meme" that was coined by the British scientist Richard Dawkins in his book, *The Selfish Gene* (1976), to represent an evolutionary unit for cultural transmission analagous to a gene in biological evolution.

See

► Evolutionary Algorithms
► Genetic Algorithms
► Metaheuristics

References

Lim, M. H., Gustafson, S., Krasnogor, N., & Ong, Y. S. (2009). Editorial to the first issue. *Memetic Computing, 1*, 1–2.

Memoryless Property

For stochastic processes, lack-of-memory is synonymous with the Markov property. For a positive random variable T that models the duration of some phenomenon, lack-of-memory means that the time remaining is independent of the time already passed, i.e., $\Pr\{T > t + s \mid T > s\} = \Pr\{T > t\}$ for $s, t > 0$. The exponential distribution is the only continuous distribution with lack-of-memory, while the geometric distribution is the only discrete distribution with lack-of-memory.

See

► Exponential Arrivals
► Markov Processes
► Markov Property
► Poisson Arrivals
► Poisson Process
► Queueing Theory

Menu Planning

A diet problem in which the variables represent complete menu items such as appetizers and entrees, instead of individual foods. The problem is formulated as an integer-programming problem in which the integer binary variables represent the decision of selecting or not selecting a complete menu item.

Metagame Analysis

A problem structuring method that addresses situations of conflict and cooperation between independent actors. Based on game-theoretic concepts, it identifies explicit and implicit threats and promises between the actors to analyze the stability of alternative scenarios.

Metaheuristics

Kenneth Sörensen[1] and Fred W. Glover[2,3]
[1]University of Antwerp, Antwerp, Belgium
[2]OptTek Systems, Inc., Boulder, CO, USA
[3]University of Colorado Boulder, Boulder, CO, USA

Introduction

A metaheuristic is a high-level problem-independent algorithmic framework that provides a set of guidelines or strategies to develop heuristic optimization algorithms. The term is also used to refer to a problem-specific implementation of a heuristic optimization algorithm according to the guidelines expressed in such a framework. It combines the Greek prefix meta- (μετά, beyond in the sense of high-level) with heuristic (from the Greek heuriskein or ευρισχειν, to search) and was coined by Fred Glover in 1986.

Most metaheuristic frameworks have their origin in the 1980s (although in some cases roots can be traced to the mid 1960s and 1970s) and were proposed as an alternative to classic methods of optimization such as branch-and-bound and dynamic programming. As a means for solving difficult optimization problems, metaheuristics have enjoyed a steady rise in both use and popularity since the early 1980s. EU/ME – the metaheuristics community – is the EURO-sponsored working group on metaheuristics and the largest platform for communication among metaheuristics researchers worldwide. Conferences and journals devoted to metaheuristics, along with some software, are described at the end of this article.

Different metaheuristics can vary significantly in their underlying foundations. Some metaheuristics mimick a process seemingly unrelated to optimization, such as natural evolution, the cooling of a crystalline solid, or the behavior of animal swarms. Attending such variation is also a striking similarity among some methods that rely on a common foundation. For example, many methods have been proposed (and given different names) that differ in not much more than the metaphor underlying them, which is often a close variant of an original method's metaphor. In this manner, the metaheuristic framework of ant colony optimization, for instance, has spawned a steady stream of different social insect-based methods (using bees, flies, termites, etc.). Most metaheuristic frameworks advise the use of randomness, although some propose completely deterministic strategies. In optimization, metaheuristics are most often used to solve combinatorial optimization problems, although metaheuristics for other problems exist (see below).

One of the defining characteristics of a metaheuristic framework is that the resulting methods are — as the name suggests — always heuristic in nature. Exact methods for combinatorial optimization, such as branch-and-bound or dynamic programming, are subject to combinatorial explosion, i.e., for NP-hard problems the computing time required by such methods increases as an exponential function of the problem size. By relaxing the demand that the optimal solution should be found in a finite (but often prohibitively large) amount of time, optimization methods can be built that attempt to find a solution that is good enough in a computing time that is small enough. However, there are important aspects of metaheuristics that link them more closely with exact methods and that give rise to a number of hybrids that unite these two types of methods. These aspects will be discussed later.

The required quality of a solution and the maximum allowable computing time can, of course, vary greatly across optimization problems and situations. Metaheuristic frameworks, being defined in very general terms, can be adapted to fit the needs of most real-life optimization problems, from the smallest and simplest to the largest and most complex. Additionally, metaheuristics do not put any demands on the formulation of the optimization problem (like requiring constraints or objective functions to be expressed as linear functions of the decision variables), in contrast, for example, to methods for mixed-integer programming. As a result, several

commercial software vendors have implemented metaheuristics as their primary optimization engines, both in specialized software packages for production scheduling, vehicle routing (Sörensen et al. 2008) and nurse rostering (Burke et al. 2004), as well as in general-purpose simulation packages (April et al. 2003; Fu 2002; Glover et al. 1999).

However, the research field of metaheuristics is not without its critics, most of whom attack the perceived lack of a universally applicable design methodology for metaheuristics and the lack of scientific rigor in testing and comparing different implementations. The no free lunch theorems (Wolpert and Macready 1997) state that, when averaged over all problems, all optimization methods perform equally well. This suggests that no single metaheuristic can be considered as a panacea for combinatorial optimization problems, but rather that a lot of problem-specific tuning is necessary to achieve acceptable performance. Moreover, metaheuristics often have a large number of parameters and tuning them is a notoriously difficult process. Consequently, computational testing to compare different metaheuristics is very difficult and often done in an ad-hoc way, rather than by established scientific standards (Barr et al. 1995; Hooker 1995; Rardin and Uzsoy 2001) This has motivated work on self-adaptive metaheuristics that automatically tune their parameters (Cotta et al. 2008; Kramer 2008; Nonobe and Ibaraki 2001, 2002) From an alternative perspective, if a research study identifies parameter values that work well for a selected class of applications — as most studies attempt to do — then for practical purposes other researchers can consider these parameters as being constants (Of course, this doesn't prevent future experimentation from seeking better parameter values.)

Another criticism sometimes levied at metaheuristics concerns the occasional tendency to create overly intricate methods (Michalewicz and Fogel 2004) with many different operators, where the contribution of these operators to the final quality of the solutions found may be poorly understood (Watson et al. 2006). Despite some theoretical results, such as proofs for the convergence of some metaheuristics under special assumptions – usually infinite running time (Eiben et al. 1991; Mitra et al. 1985) – or attempts to explain why genetic algorithms work (such as the heavily criticized Wright et al. (2003) building block hypothesis (Holland 1975)), research papers that attempt to capture the fundamental reasons why metaheuristics work are still few and far between.

Despite these criticisms, the ability to obtain good solutions where other methods fail has made metaheuristics the method of choice for solving a majority of large real-life optimization problems, both in academic research and in practical applications.

Metaheuristic Concepts

Like all optimization methods, metaheuristics attempt to find the best (feasible) solution out of all possible solutions of an optimization problem. In order to do this, they examine various solutions and perform a series of operations on them in order to find different, better solutions.

Metaheuristics operate on a representation or encoding of a solution, an object that can be stored in computer memory and can be conveniently manipulated by the different operators employed by the metaheuristic. Since metaheuristics are most often used to solve combinatorial optimization problems, representations too are generally combinatorial in nature (i.e., they are able to represent only a finite number of solutions). Representations used in the metaheuristics literature are quite diverse (see, e.g., Talbi (2009) for an overview) and range from vector-representations (binary, integer) over permutations to more complex representations such as trees and other graphs. Many metaheuristic algorithms use a combination of different representation types, such as a vector of permutations. Contrary to exact algorithms, metaheuristics do not require the encoding of solutions to be a bijection, i.e., several solutions may share the same encoding and a single solution may be encoded in different ways. Often, an encoding is chosen on the grounds of being convenient to manipulate, although sometimes a time-consuming decoding procedure may be required to obtain the actual solution (such as the encoding used in Prins (2004)).

Although many different metaheuristics have been proposed, their mechanisms for obtaining good solutions primarily operate by manipulating solutions in three ways: by iteratively making small changes to a current solution (local search metaheuristics), by

constructing solutions from their constituting parts (constructive metaheuristics), and by iteratively combining solutions into new ones (population-based metaheuristics). Each of these manipulation mechanisms gives rise to a class of metaheuristic frameworks that are discussed separately below. It is important to note that these classes are not mutually exclusive, and many metaheuristic algorithms combine ideas from each of them. Also, in some instances the transitions from one solution to another are achieved by solving specially generated subproblems.

Local Search Metaheuristics

Local search metaheuristics find good solutions by iteratively making small changes, called moves, to a single solution, called the current solution. The set of solutions that can be obtained by applying a single move to a given solution is called the neighborhood of that solution. In each iteration, a solution from the neighborhood of the current solution is selected to become the new current solution. The sequence of moves defines a trajectory through the search space. Hence, local search metaheuristics are also known under the names of neighborhood search methods or trajectory methods.

For almost all problem representations, different move types can be defined, resulting in different neighborhood structures. The rule used to select the new current solution is called the move strategy or search strategy and determines the aggressiveness of the search. Metaheuristics that use the steepest descent or steepest ascent strategy select the best move from the neighborhood and are often called hill-climbers. Other move strategies include selecting the first move that improves upon the current solution (called the mildest ascent/descent or first-improving strategy), as well as selecting a random improving solution.

In general, the set of allowable moves is restricted to those that result in solutions that are both feasible and improve upon the current solution. Some metaheuristics allow infeasible moves in a strategy that is called strategic oscillation. In this strategy, the search is usually only allowed to temporarily remain in the infeasible region of the search space. A striking example of the utility of this strategy is shown in Glover and Hao (2010).

A solution whose neighborhood does not contain any better solutions is called a local optimum (as opposed to a global optimum, i.e., a best possible solution to the optimization problem). When the current solution is a local optimum, the metaheuristic utilizes a strategy to escape to other regions of the search space. It is this strategy that distinguishes metaheuristics from simple heuristics and from each other. The metaheuristic's name therefore usually refers to the strategy to prevent the search from becoming ensnared within regions whose local optima may be substantially inferior to a global optimum.

The simplest strategy to escape to potentially more fertile regions is to either start the search again from a new, usually random, solution or to make a relatively large change (called a perturbation) to the current solution. These strategies are respectively called multi-start local search (MLS) and iterated local search (ILS) (Lourenco et al. 2003).

A number of metaheuristics define different move types and change the move type used once a local optimum has been reached. The rationale for this strategy is that a local optimum relative to a specific move type can often be improved by performing local search with a different move type. The global optimum on the other hand is a local optimum with respect to every possible move type. Metaheuristics that use this strategy are commonly called variable neighborhood search (VNS) (Mladenović and Hansen 1997) algorithms, but using more than one neighborhood is far more common in the metaheuristics literature and not restricted to algorithms labeled VNS (Sörensen et al. 2008).

Using memory structures is a third commonly encountered way for metaheuristics to avoid remaining trapped in a local optimum and to guide the search in general so as to find good solutions more quickly. Algorithms that use memory structures are commonly grouped under the umbrella term tabu search (Glover 1989, 1990, 1996) algorithms (sometimes also called adaptive memory programming algorithms). Different memory structures may be used to explicitly remember different aspects about the trajectory through the search space that the algorithm has previously undertaken and different strategies may be devised to use this information to direct the search (Glover and Laguna 1993) to promising areas of the search space. Often-used memory structures include the tabu list (from which the name of the metaheuristic

framework derives) that records the last encountered solutions (or some attributes of them) and forbids these solutions (or attributes) from being visited again as long as they are on the list. Some variants record move attributes rather than solution attributes on the tabu list, for the purpose of preventing moves from being reversed. The tabu list is usually organized in a first-in, first-out (FIFO) fashion, i.e., the current solution replaces the oldest one on the list. The length of the tabu list is called the tabu tenure. Frequency memory records how often certain attributes have been encountered in solutions on the search trajectory, which allows the search to avoid visiting solutions that display the most often encountered attributes or to visit solutions with attributes seldom encountered. Such memory can also include an evaluative component that allows moves to be influenced by the quality of solutions previously encountered that contain various attributes or attribute combinations. Other memory structures such as an elite set of the best solutions encountered so far are also common. Another example of the use of memory can be found in a metaheuristic called guided local search (GLS) (Voudouris and Tsang 1999). GLS introduces an augmented objective function that includes a penalty factor for each potential element. When trapped in a local optimum, GLS increases the penalty factor for all elements of the current solution, making other elements (and therefore other moves) more attractive and allowing the search to escape from the local optimum. Similarly, some variants of tabu search use penalties to determine the tabu status of moves, though drawing more strongly on memory.

Contrary to most other local search metaheuristics, simulated annealing uses a random move strategy, emulating the annealing process of a crystalline solid. At each iteration, this strategy selects a random solution x' from the neighborhood of the current solution x and accepts x' as the new current solution with probability $e^{-[f(x')-f(x)]/T}$, where $f(\cdot)$ is the objective function value (to be maximized) of the solution and T is an endogenous parameter called the temperature. The acceptance probability increases as the increase in solution quality is higher (or the decrease is lower). The temperature is initially set to a high value, which leads to higher acceptance probabilities, and then gradually lowered as the search progresses (although it may be increases again at certain moments during the search). The function that describes the evolution of T throughout the different iterations is called the cooling schedule. Simulated annealing was first described in Kirkpatrick et al. (1983), based upon an algorithm by Metropolis et al. (1953).

Relaxation induced local search (RINS) (Danna et al. 2005) is a metaheuristic that constructs a promising neighborhood using information contained in the continuous relaxation of the mixed integer programming (MIP) model of the optimization problem. Because it does not need problem-specific information to construct its neighborhood, RINS can be more easily built into general-purpose MIP solvers [11] and is currently available in the latest versions of LINDO/LINGO and CPLEX. Contrary to other metaheuristics, RINS requires the problem to be formulated as a MIP which makes it less general than other metaheuristics.

Constructive Metaheuristics

Constructive metaheuristics constitute a separate class from local search metaheuristics in that they do not operate on complete solutions, but rather construct solutions from their constituent elements, starting from an empty set and adding one element during each iteration, an operation that is also called a move. After each iteration except the last, the algorithm therefore operates on a partial solution (e.g., a traveling salesperson tour that does not visit all cities), of which it may not be possible to determine the objective function value or the feasibility status. Constructive metaheuristics are often adaptations of greedy algorithms, i.e., algorithms that add the best possible element at each iteration, a myopic strategy that may result in suboptimal solutions.

GRASP, the acronym for greedy randomized adaptive search procedure (Feo and Resende 1995), uses randomization to overcome this drawback of purely greedy algorithms by adding some randomness to the selection process. Several variants of GRASP have been proposed, founded on the following basic idea. At each iteration, a restricted candidate list, which contains the α best elements that can be added, is updated. From the restricted candidate list, a random element is selected for addition to the partial solution, after which the list is updated to reflect the new situation. The parameter α determines the greediness of the search: if α equals 1, the search is completely greedy, whereas if α is equal to the

number of elements that can be added, the search is completely random. A particularly useful advance in GRASP algorithms has occurred by blending them with the path relinking strategy of tabu search. Notable examples of this approach include Commander et al. (2008); Nascimento et al. (2010); Resende et al. (2010).

Rather that using randomness to outperform a greedy heuristic, more strategic ways of performing constructive (or destructive) moves, once again making use of memory, are examined in Fleurent and Glover (1999); Glover et al. (2000). Another approach is embodied in a look-ahead strategy (Pearl 1984), which evaluates the elements that can be added by considering not only the next move, but several moves into the future. The pilot method (Duin and Voß 1999), for example, uses a (usually greedy) constructive heuristic to determine a pilot solution for each potential move, i.e., the value of a potential element is evaluated by determining the objective function value of the solution that results from applying the heuristic to generate a complete solution from the current partial solution with this element added. The idea of looking ahead has a long history, having been proposed in probing strategies for integer programming in (Lemke and Spielberg 1967).

Ant colony optimization (ACO) (Dorigo et al. 1996, 2006) is an umbrella term for a set of related constructive metaheuristics that build solutions by imitating the foraging behavior of ants. Perhaps because of the appeal of its imagery, this class of approaches has received and continues to receive widespread attention in the popular press (e.g., Anonymous 2010). Ant colony optimization introduces an external parameter for each potential element called the pheromone level (a pheromone is a chemical factor that triggers a social response in the same species), initially set to zero for all elements. The metaheuristic uses multiple parallel artificial agents (called ants) that each construct a solution by an iterative constructive process in which elements are selected based on a combination of the value of that element and its pheromone level. Once all ants have constructed a solution, the pheromone level of all elements is updated in a way that reflects the quality of the solution found by that ant (the elements of better solutions receive more pheromone). Each ant then constructs a new solution, but elements that were present in high-quality solutions will now receive a higher probability of being selected by the ants. Periodically, the pheromone level of all elements is reduced to reflect evaporation. The process of constructing solutions in the way described above is repeated, and the best solution found is reported at the end.

To improve the quality of the final solutions, most constructive metaheuristics include a local search phase after the construction phase.

Population-Based Metaheuristics

The main mechanism that allows population-based metaheuristics to find good solutions is the combination of existing solutions from a set, usually called the population. The fundamental reasoning behind this class of metaheuristics is that good solutions can be found by exchanging solution attributes between two or more (usually high-quality) solutions. The most important members of this class are called evolutionary algorithms because they mimic the principles of natural evolution. Following Michalewicz and Fogel (2004), here the term evolutionary algorithms is used as an umbrella term to encompass the wide range names given to metaheuristics based on evolution. This includes genetic algorithms (Goldberg et al. 1989; Holland 1975), genetic/evolutionary programming (Koza 1992), evolutionary computation (Fogel 2006), evolution strategies (Beyer and Schwefel 2002), and many others. The literature on evolutionary algorithms is larger than that on other metaheuristics, and this field has spawned several dedicated journals and conferences.

Typical of the field of evolutionary algorithms is that its researchers tend to adopt the vocabulary of the metaphor on which the algorithms are based. The descriptions of these algorithms therefore are stated in terms of chromosomes (instead of solutions), fitness (instead of objective function value), genotype (instead of encoding), etc. The driving force behind most evolutionary algorithms is selection and recombination. Selection ensures that predominantly high-quality solutions in the population are selected for recombination, usually by biasing the probability of each solution in the population to be selected towards its objective function value. Recombination utilizes specialized operators to combine the attributes of two or more solutions into new ones. The new solutions are then added to the population by a process called

reinsertion, possibly subject to feasibility or minimum quality demands, to replace (usually low-quality) solutions. In a large majority of cases, all operators (selection, recombination and reinsertion) make heavy use of randomness. A large number of evolutionary algorithms additionally include a mutation operator that (again, randomly) changes a solution after it has been recombined. Most evolutionary algorithms iterate the selection, recombination, mutation, and reinsertion phases a number of times, and report the best solution in the population.

Scatter search and path relinking (Glover et al. 2000, 2003) are both population-based metaheuristics for continuous (or mixed-integer) and combinatorial optimization respectively, proposed as a deterministic alternative for the highly stochastic evolutionary algorithms. Scatter search encodes solutions as real-valued vectors (or rounded real-valued vectors for integer values) and generates new solutions by considering convex or concave linear combinations of these vectors. Path relinking, on the other hand, generalizes this idea, making it applicable to combinatorial optimization problems, by generating paths between high-quality solutions. Paths consist of elementary moves such as the ones used in local search metaheuristics and essentially link one solution (called the initiating solution) to a second solution (called the guiding solution) in the solution space. Contrary to local search metaheuristics, path relinking uses a move strategy that chooses the move to execute based on the fact that this move will bring the solution closer to the guiding solution. In both scatter search and path relinking, the selection of both initiating and guiding solution from a population (called the reference set) is done in a deterministic way, as are the mechanisms for updating the reference set once new solutions have been generated.

Hybrid Metaheuristics

Metaheuristics that combine aspects or operators from different metaheuristics paradigms are called hybrid metaheurstics. The term has lost much of its discriminatory power, however, since such combinations of operators from different metaheuristic frameworks have become the norm rather than the exception. Indeed, there is a tendency in the metaheuristics research field to look at metaheuristics frameworks as providing general ideas or components to build optimization algorithms, rather than to consider them as recipes that should be closely followed (Michalewicz and Fogel 2004). In this spirit, many metaheuristics use specialized heuristics to efficiently solve subproblems produced by the metaheuristic method (e.g., Gendreau et al. 1994). Also, a large number of local search metaheuristics use a construction phase to find an initial solution (or a set of initial solutions) from which to start the neighborhood search. In fact the original description of the GRASP metaheuristic (Feo and Resende 1995) prescribes a local search phase to follow the greedy randomized construction phase.

Memetic algorithms (Moscato 1989) are the only class of hybrid metaheuristics that has been given a specific name. Metaheuristics belonging to this class combine recombination operators from the class of evolutionary algorithms with local search (meta) heuristics. Although the name is commonly used, many evolutionary algorithms either replace or complement their mutation operator with a local search phase and can also be considered memetic.

Metaheuristics and Exact Methods

A more recent development has been a special focus on combining ideas from different metaheuristics, usually local search, with exact methods such as branch-and-bound or branch-and-cut. Sometimes called matheuristics, the resulting method usually integrates existing exact procedures to solve subproblems and guide the higher-level heuristic (Dumitrescu and Stützle 2009; Raidl and Puchinger 2008). In a similar way, ideas and operators from constraint programming techniques are integrated with metaheuristics (Van Hentenryck and Michel 2009). The links between metaheuristics and exact methods provide examples of additional forms of combinations:

1. There exist exact methods for solving various special classes of optimization problems, such as linear programming and certain graph (or matroid) problems, that can be incorporated to solve subproblems produced by a metaheuristic method. Such subproblems can be generated by a decomposition strategy, a restriction strategy or a relaxation strategy (see Glover and Klingman (1988); Rego (2005)).
2. Exact methods for more complex problems can sometimes solve small instances of these problems effectively. A metaheuristic may operate by constructing collections of such small instances as

a strategy for generating structured moves that transition from a given solution to a new one (see, e.g., Glover (2005)).
3. An exact method can be run for a very long time to obtain optimal solutions (to at least some instances of a problem class), and these optimal solutions can be used in the learning approach called target analysis (Glover 1990; Glover and Laguna 1997) as a way to produce improved decision rules for both metaheuristics and exact methods.
4. Metaheuristics can be integrated with exact methods to improve the performance of the exact methods (Friden et al. 1989; Glover 1990; Puchinger et al. 2009).
5. By not demanding that the optimal solution be found, metaheuristics can, for example, employ a truncated optimization method in place of (or in conjunction with) generating subproblems that are structured to be easier to solve.

Metaheuristics for Different Optimization Problems

Continuous Optimization

Although metaheuristics are predominantly used for combinatorial optimization, many of them have been adapted for continuous optimization. Some metaheuristics are very naturally defined over continuous search spaces. Notable examples include scatter search (Glover et al. 2000), particle swarm optimization (Kennedy et al. 1995) and an evolutionary approach called differential evolution (Storn and Price 1997). Other, especially constructive and local search approaches, require a considerable adaptation from their original formulation. Nonetheless, algorithms for continuous optimization based on tabu search (Chelouah and Siarry 2000; Glover 1994), GRASP (Hirsch et al. 2007), variable neighborhood search (Liberti and Drazič 2005), and others, have been proposed.

Multi-objective Optimization

Many real-life problems have multiple objectives, for which the notion of optimality is generally replaced with the notion of dominance. A solution is said to dominate another solution if its quality is at least as good on every objective and better on atleast one. In multi-objective optimization, the set of non-dominated solutions is called the Pareto set and the projection of this set onto the objective function space is called the Pareto front or Pareto frontier. The aim of multi-objective metaheuristics, i.e., metaheuristics specifically designed to solve multi-objective optimization problems, is to approximate the Pareto front as closely as possible (Zitzler et al. 2004). The outcome of any multi-objective algorithm is therefore generally a set of mutually non-dominated solutions, the Pareto set approximation. To measure the quality of such an approximation, many different measures exist (Jaszkiewicz 2004). Although adaptations to the multi-objective paradigm of both tabu search and simulated annealing exist (Czyżak et al. 1998; Hansen 1997), most multi-objective metaheuristics are of the evolutionary type (Jones et al. 2002), a fact generally attributed to the observation that these algorithms naturally operate on a set of solutions. Evolutionary multi-objective metaheuristics include the vector evaluated genetic algorithm (VEGA) (Schaffer 1985), the non-dominated sorting algorithm (NDSA) (Srinivas and Deb 1994), the multi-objective genetic algorithm (MOGA) (Fonseca and Fleming 1993) and the improved strength pareto evolutionary algorithm (SPEA2) (Zitzler and Thiele 1999).

Stochastic Optimization

Stochastic combinatorial optimization problems include uncertain, stochastic or dynamic information in their parameters. Metaheuristics for such problems therefore need to take into account that the objective function value is a random variable and that the constraints are violated with some probability. Evaluating a solution's objective function value and/or its feasibility can be done either exactly (if a closed-form expression is available), by approximation or by Monte Carlo simulation. Metaheuristicsusing each of these possibilities have been proposed to solve different stochastic problems (Bianchi et al. 2009; Ribeiro and Resende 2010).

Research in Metaheuristics

Conferences

The premier conference on metaheuristics is MIC, the Metaheuristics International Conference.

Other conferences on metaheuristics include the yearly EU/ME meeting on a specific metaheuristics-related topic, organized by EU/ME in collaboration with a local research group, and the Hybrid Metaheuristics conference series that focuses on combinations of different metaheuristics and the integration of AI/OR techniques. The Learning and Intelligent Optimization conferences aim at exploring the boundaries between machine learning, artificial intelligent, mathematical programming and algorithms for optimization.

A large number of conferences focus exclusively on evolutionary algorithms, including Parallel Problem Solving From Nature (PPSN), the Genetic and Evolutionary Computation Conference (GECCO), EvoStar (a multi-conference comprising EuroGP, EvoCOP, EvoBIO,and EvoApplications), Evolutionary Multi-Criterion Optimization (EMO), and the IEEE Congress on Evolutionary Computation (CEC).

The Ants conference series is dedicated to research in swarm intelligence methods.

Journals

The field of metaheuristics has several dedicated journals: the well-established *Journal of Heuristics* and the newer *International Journal of Metaheuristics* and *International Journal of Applied Metaheuristic Computing* (IJAMC). However, a large majority of articles on metaheuristics are published in general OR/MS journals.

Several journals are devoted exclusively to evolutionary algorithms: *Evolutionary Computation*, *IEEE Transactions on Evolutionary Computation*, *Genetic Programming and Evolvable Machines*, and the *Journal of Artificial Evolution and Applications*.

The journal *Swarm Intelligence* is currently the main journal for advances in the swarm intelligence area.

Metaheuristics Software

Several vendors of commercial optimization software have included (albeit to a limited extent) metaheuristics in their packages. Frontline Systems' Risk Solver Platform and its derivatives, an extension of the Microsoft Excel Solver, include a hybrid evolutionary solver. Tomlab/GENO is a package for static or dynamic, single- or multi-objective optimization based on a real-coded genetic algorithm. Both LINDO/LINGO and CPLEX include the relaxation induced neighborhood search (RINS) metaheuristic.

Open source metaheuristics software frameworks have recently appeared in the COIN-OR library. These include METSlib, an object oriented metaheuristics optimization framework, and Open Tabu Search (OTS), a framework for constructing tabu search algorithms.

Besides these solvers for combinatorial optimization, most commercial (stochastic) simulation packages today include an optimization tool (Fu 2002). Autostat, included in AutoMod, and Simrunner, included in ProModel, both use evolutionary algorithms. A variety of companies in the simulation industry, as well as general management service and consulting firms like Rockwell Software, Dassault Systemes, Flextronics, Halliburton, HP, Planview and CACI, employ OptQuest, which uses tabu search and scatter search.

See

- ▶ Artificial Intelligence
- ▶ COIN-OR Computational Infrastructure for Operations Research
- ▶ Heuristics
- ▶ Integer and Combinatorial Optimization
- ▶ Multi-attribute Utility Theory
- ▶ Neural Networks
- ▶ Simulated Annealing
- ▶ Simulation Optimization
- ▶ Tabu Search

References

Anonymous (2010). Riders on a swarm. *The Economist*, 12 August 2010.

April, J., Glover, F., Kelly, J., & Laguna, M. (2003). Practical introduction to simulation optimization. In S. Chick, T. Sanchez, D. Ferrin, & D. Morrice, (Eds.), *Proceedings of the 2003 Winter Simulation Conference* 2003.

Barr, R. S., Golden, B. L., Kelly, J. P., Resende, M. G. C., & Stewart, W. R. (1995). Designing and reporting on computational experiments with heuristic methods. *Journal of Heuristics, 1*(1), 9–32.

Beyer, H. G., & Schwefel, H. P. (2002). Evolution strategies–a comprehensive introduction. *Natural Computing, 1*(1), 3–52.

Bianchi, L., Dorigo, M., Gambardella, L. M., & Gutjahr, W. J. (2009). A survey on metaheuristics for stochastic combinatorial optimization. *Natural Computing, 80*(2), 239–287.

Burke, E., De Causmaecker, P., Petrovic, S., Berghe, G. V., et al. (2004). Variable neighborhood search for nurse rostering problems. In M. G. C. Resende & A. Viana (Eds.), *Metaheuristics: Computer decision-making* (pp. 153–172). Boston: Kluwer Academic.

Chelouah, R., & Siarry, P. (2000). Tabu search applied to global optimization. *European Journal of Operational Research, 123*(2), 256–270.

Commander, C., Festa, P., Oliveira, C. A. S., Pardalos, P. M., Resende, M. G. C., & Tsitselis, M. (2008). Grasp with path-relinking for the cooperative communication problem on ad hoc networks. In D. A. Grundel, R. A. Murphey, P. M. Pardalos, & O. A. Prokopyev (Eds.), *Cooperative networks: Control and optimization* (pp. 187–207). Cheltenham: Edward Elgar Publishing.

Cotta, C., Sevaux, M., & Sörensen, K. (2008). *Adaptive and multilevel metaheuristics*. Berlin: Springer-Verlag.

Czyżak, P., et al. (1998). Pareto simulated annealing-a metaheuristic technique for multiple-objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis, 7*(1), 34–47.

Danna, E. (2004). Integrating local search techniques into mixed integer programming. 4OR. *A Quarterly Journal of Operations Research., 2*(4), 321–324.

Danna, E., Rothberg, E., & Le Pape, C. (2005). Exploring relaxation induced neighborhoods to improve MIP solutions. *Mathematical Programming, 102*(1), 71–90.

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, 26*(1), 29–41.

Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. *IEEE Computational Intelligence Magazine, 1*(4), 28–39.

Duin, C., & Voß, S. (1999). The pilot method: A strategy for heuristic repetition with application to the Steiner problem in graphs. *Networks, 34*(3), 181–191.

Dumitrescu, I., & Stützle, T. (2009). Usage of exact algorithms to enhance stochastic local search algorithms. In V. Maniezzo, T. Stützle, & S. Voß (Eds.), *Matheuristics: Hybridizing metaheuristics and mathematical programming, volume 10 of annals of information systems* (Vol. 10). New York: Springer-Verlag.

Eiben, A., Aarts, E., & Van Hee K. (1991). Global convergence of genetic algorithms: A Markov chain analysis. *Parallel problem solving from nature*, (pp. 3–12).

Feo, T. A., & Resende, M. G. C. (1995). Greedy randomized adaptive search procedures. *Journal of Global Optimization, 6*(2), 109–133.

Fleurent, C., & Glover, F. (1999). Improved constructive multistart strategies for the quadratic assignment problem using adaptive memory. *INFORMS Journal on Computing, 11*(2), 198–204.

Fogel, D. B. (2006). *Evolutionary computation: Toward a new philosophy of machine intelligence*. New York: Wiley-IEEE Press.

Fonseca, C. M., & Fleming, P. J. (1993). Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In *Proceedings of the fifth international conference on genetic algorithms*, (pp. 416–423), Citeseer.

Friden, C., Hertz, A., & de Werra, D. (1989). *TABARIS: An exact algorithms based on tabu search for finding a maximum independent set in a graph*. Working paper, Swiss Federal Institute of Technology, Lausanne.

Fu, M. C. (2002). Optimization for simulation: Theory vs practice. *INFORMS Journal on Computing, 14*(3), 192–215.

Gendreau, M., Hertz, A., & Laporte, G. (1994). A tabu search heuristic for the vehicle routing problem. *Management Science, 40*(10), 1276–1290.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research, 13*, 533–549.

Glover, F. (1989). Tabu search-part I. *ORSA Journal on Computing, 1*(3), 190–206.

Glover, F. (1990). Tabu search-part II. *ORSA Journal on Computing, 2*(1), 4–32.

Glover, F. (1994). Tabu search nonlinear and parametric optimization (with links to genetic algorithms). *Discrete Applied Mathematics, 49*, 231–255.

Glover, F. (1996). Tabu search and adaptive memory programming: Advances, applications and challenges. In R. Barr, R. Helgason, & J. L. Kennington (Eds.), *Interfaces in computer science and operations research*. Boston: Kluwer Academic.

Glover, F. (2005). Adaptive memory projection methods for integer programming. In C. Rego & B. Alidaee (Eds.), *Metaheuristic optimization via memory and evolution* (pp. 425–440). Boston: Kluwer Academic.

Glover, F., & Hao, J. K. (2010). The case for strategic oscillation. *Annals of Operations Research*. DOI:10.1007/s10479-009-0597-1.

Glover, F., & Klingman, D. (1988). Layering strategies for creating exploitable structure in linear and integer programs. *Mathematical Programming, 40*(1), 165–181.

Glover, F., & Laguna, M. (1993). Tabu search. In C. R. Reeves (Ed.), *Modern heuristic techniques for combinatorial problems* (pp. 70–141). New York: John Wiley & Sons.

Glover, F., & Laguna, M. (1997). *Tabu search*. Boston: Kluwer Academic.

Glover, F., Kelly, J., & Laguna, M. (1999). New advances wedding simulation and optimization. In D. Kelton, (ed.), *Proceedings of the 1999 Winter Simulation Conference*.

Glover, F., Laguna, M., & Martí, R. (2000). Fundamentals of scatter search and path relinking. *Control and Cybernetics, 39*(3), 653–684.

Glover, F., Laguna, M., & Marti, R. (2003). Scatter search and path relinking: Advances and applications. In *Handbook of metaheuristics*, (pp. 1–35).

Goldberg, D. E., et al. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading Menlo Park: Addison-Wesley.

Hansen, M. P. (1997). Tabu search for multiobjective optimization: MOTS. In *Proceedings of the 13th International Conference on Multiple Criteria Decision Making (MCDM'97), Cape Town, South Africa*, (pp. 574–586), Citeseer.

Hirsch, M. J., Meneses, C. N., Pardalos, P. M., & Resende, M. G. C. (2007). Global optimization by continuous GRASP. *Optimization Letters, 1*(2), 201–212.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor, MI: University of Michigan Press.

Hooker, J. N. (1995). Testing heuristics: We have it all wrong. *Journal of Heuristics, 1*(1), 33–42.

Jaszkiewicz, A. (2004). Evaluation of multiobjective metaheuristics. In X. Gandibleux, M. Sevaux, K. Sörensen, & V. T'kindt (Eds.), *Metaheuristics for multiobjective optimization* (Lecture notes in economics and mathematical systems, Vol. 535, pp. 65–90). Berlin: Springer-Verlag.

Jones, D. F., Mirrazavi, S. K., & Tamiz, M. (2002). Multi-objective meta-heuristics: An overview of the current state-of-the-art. *European Journal of Operational Research, 137*(1), 1–9.

Kennedy, J., Eberhart, R. C. et al. (1995). Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*, *4*, 1942–1948.

Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science, 220*(4598), 671.

Koza, J. R. (1992). *Genetic programming: On the programming of computers by means of natural selection*. Cambridge: The MIT press.

Kramer, O. (2008). *Self-adaptive heuristics for evolutionary computation*. Berlin: Springer-Verlag.

Lemke, C., & Spielberg, K. (1967). Direct search algorithms for zero–one and mixed integer programming. *Operations Research, 15*, 892–914.

Liberti, L., & Drazič, M. (2005) Variable neighbourhood search for the global optimization of constrained NLPs. In *Proceedings of GO*, (pp. 1–5).

Lourenco, H., Martin, O., & Stützle, T. (2003). Iterated local search.In *Handbook of metaheuristics*, (pp. 320–353).

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., Teller, E., et al. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics, 21*(6), 1087.

Michalewicz, Z., & Fogel, D. B. (2004). *How to solve it: Modern heuristics*. New York: Springer-Verlag.

Mitra, D., Romeo, F., & Sangiovanni-Vincentelli, A. (1985). Convergence and finite-time behavior of simulated annealing. In *1985 24th IEEE Conference on Decision and Control*, Vol. 24.

Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. *Computers and Operations Research, 24*(11), 1097–1100.

Moscato, P. (1989). On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. *Caltech Concurrent Computation Program, C3P Report*, 826.

Nascimento, M. C. V., Resende, M. G. C., & Toledo, F. M. B. (2010). Grasp heuristic with path-relinking for the multi-plant capacitated lot sizing problem. *European Journal of Operational Research, 200*, 747–754.

Nonobe, K., & Ibaraki, T. (2001). An improved tabu search method for the weighted constraint satisfaction problem. *INFOR, 39*(2), 131–151.

Nonobe, K., & Ibaraki, T. (2002). Formulation and tabu search algorithm for the resource constrained project scheduling problem. In C. C. Ribeiro & P. Hansen (Eds.), *Essays and surveys in metaheuristics* (pp. 557–588). Boston: Kluwer Academic.

Pearl, J. (1984). *Heuristics–intelligent search strategies for computer problem solving*. Reading, MA: Addison-Wesley.

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers and Operations Research, 31*(12), 1985–2002.

Puchinger, J., Raidl, G. R., & Pirkwieser, S. (2009). Metaboosting: Enhancing integer programming techniques by metaheuristics. In V. Maniezzo, T. Stützle, & S. Voß (Eds.), *Matheuristics: Hybridizing metaheuristics and mathematical programming* (Annals of information systems, Vol. 10). New York: Springer-Verlag.

Raidl, G. R., & Puchinger, J. (2008). Combining (integer) linear programming techniques and metaheuristics for combinatorial optimization. In C. Blum, M. J. Blesa Aguilera, A. Roli, & M. Sampels (Eds.), *Hybrid metaheuristics: An emerging approach to optimization* (Studies in computational intelligence, Vol. 114). Berlin: Springer-Verlag.

Rardin, R. L., & Uzsoy, R. (2001). Experimental evaluation of heuristic optimization algorithms: A tutorial. *Journal of Heuristics, 7*(3), 261–304.

Rego, C. (2005). RAMP: A new metaheuristic framework for combinatorial optimization. In C. Rego & B. Alidaee (Eds.), *Metaheuristic optimization via memory and evolution: Tabu search and scatter search* (pp. 441–460). Boston: Kluwer Academic.

Resende, M. G. C., Martí, R., Gallego, M., & Duarte, A. (2010). Grasp and path relinking for the max-min diversity problem. *Computers and Operations Research, 37*, 498–508.

Ribeiro, C. C., & Resende, M. G. C. (2010). Path-relinking intensification methods for stochastic local search algorithms. Research technical report, AT&T Labs.

Schaffer, J. D. (1985). Multiple objective optimization with vector evaluated genetic algorithms. In *Proceedings of the 1st International Conference on Genetic Algorithms*, (pp. 93–100). L. Erlbaum Associates.

Sörensen, K., Sevaux, M., & Schittekat, P. (2008). *"Multiple neighbourhood search" in commercial VRP packages: Evolving towards self-adaptive methods, volume 136 of lecture notes in economics and mathematical systems, chapter adaptive, self-adaptive and multi-level metaheuristics* (pp. 239–253). London: Springer-Verlag.

Srinivas, N., & Deb, K. (1994). Muiltiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation, 2*(3), 221–248.

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization, 11*(4), 341–359.

Talbi, E. G. (2009). *Metaheuristics: From design to implementation*. Hoboken, NJ: Wiley.

Van Hentenryck, P., & Michel, L. (2009). *Constraint-based local search*. Cambridge: The MIT Press.

Voudouris, C., & Tsang, E. (1999). Guided local search and its application to the traveling salesman problem. *European Journal of Operational Research, 113*(2), 469–499.

Watson, J. P., Howe, A. E., & Darrell Whitley, L. (2006). Deconstructing nowicki and Smutnicki's i-TSAB tabu search algorithm for the job-shop scheduling problem. *Computers and Operations Research, 33*(9), 2623–2644.

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*(67).
Wright, A., Vose, M., & Rowe, J. (2003). Implicit parallelism. In *Genetic and evolutionary computation–GECCO 2003*, (pp. 211–211). Springer.
Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the srength Pareto approach. *IEEE Transactions on Evolutionary Computation, 3*(4), 257.
Zitzler, E., Laumanns, M., & Bleuler, S. (2004). A tutorial on evolutionary multiobjective optimization. In X. Gandibleux, M. Sevaux, K. Sörensen, & V. T'kindt (Eds.), *Metaheuristics for multiobjective optimization* (Lecture notes in economics and mathematical systems, Vol. 535, pp. 3–38). Berlin: Springer-Verlag.

Metamodeling

For simulation models, the objective is to provide an explicit input-output relationship through a fitted mathematical function, e.g., using statistical regression, splines, neural networks, or kriging. Differs from the use of the term in computer science.

See

► Response Surface Methodology
► Simulation Metamodeling

Method of Stages

An analysis method that extends the birth-and-death-type analysis to queueing systems with Erlangian service or interarrival times. Since an Erlang random variable can be represented as the sum of independent and identically distributed exponential random variables, the method of stages increases the state space to coincide with the underlying exponential random variables and the resulting system of equations is generally solved using generating functions.

See

► Queueing Theory

Military Operations Other Than War

Dean S. Hartley III
Oak Ridge National Laboratory, Oak Ridge, TN, USA

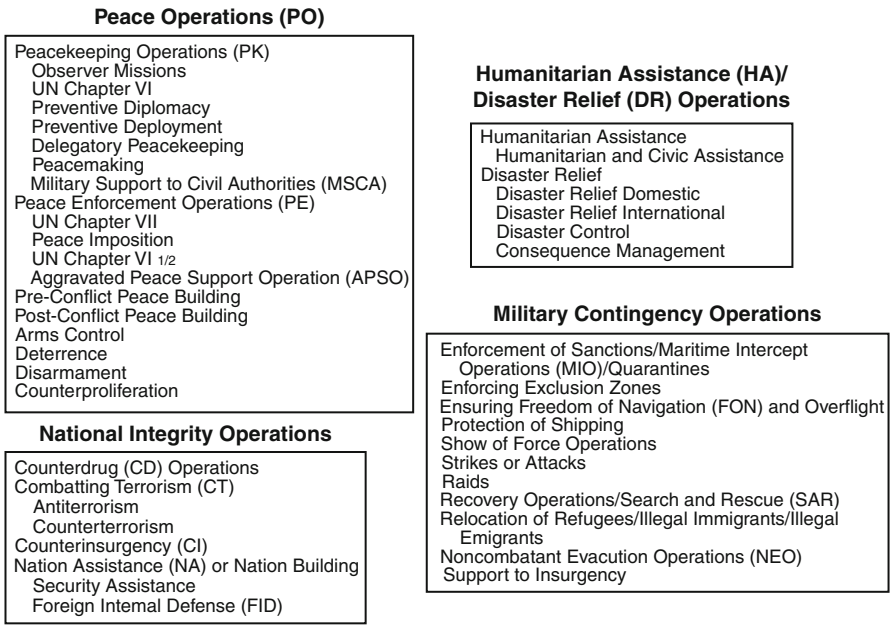
Introduction

Operations Other Than War (OOTW) suffer from an identity crisis. Sometimes called Military Operations Other Than War (MOOTW), sometimes known as Low Intensity Conflict (LIC), sometimes called Stability Support Operations (SSO), and sometimes designated as Small Scale Contingencies (SSC), these operations have caused both theoretical and practical problems for the military.

- These operations range in size from airlifting several fire trucks from Tennessee to Florida to fighting the 1998 Summer fires to the Bosnia Peacekeeping operation involving tens of thousands of U.S. military personnel and tens of thousands of other nations' military personnel, hardly a small-scale contingency.
- They include operations to provide stability to foreign countries, such as Haiti; however, they also include support to insurgencies, a "stability support operation" only in the negative.
- They include Non-combatant Evacuation Operations (NEOs) in which armed force may be needed to support the evacuation; they include operations such as Somalia that result in a number of U.S. military deaths in combat, low intensity conflict providing cold comfort to families of the dead; and they include operations such as fire-fighting that can be defined as conflict only by stretching the definition.

These operations cannot even be distinguished from other operations by time frame or geographic impact:

- Their time span ranges from the one-day cruise missile strike against Iraq to the 17-year peacekeeping operation in the Sinai (or the 45-year peacekeeping operation in Korea).
- Their geographic impact ranges from the purely local issues of disaster relief in Hawaii for Typhoon Iniki to the global geopolitical concerns stirred by peacekeeping in Bosnia.



Military Operations Other Than War, Fig. 1 Types of OOTW

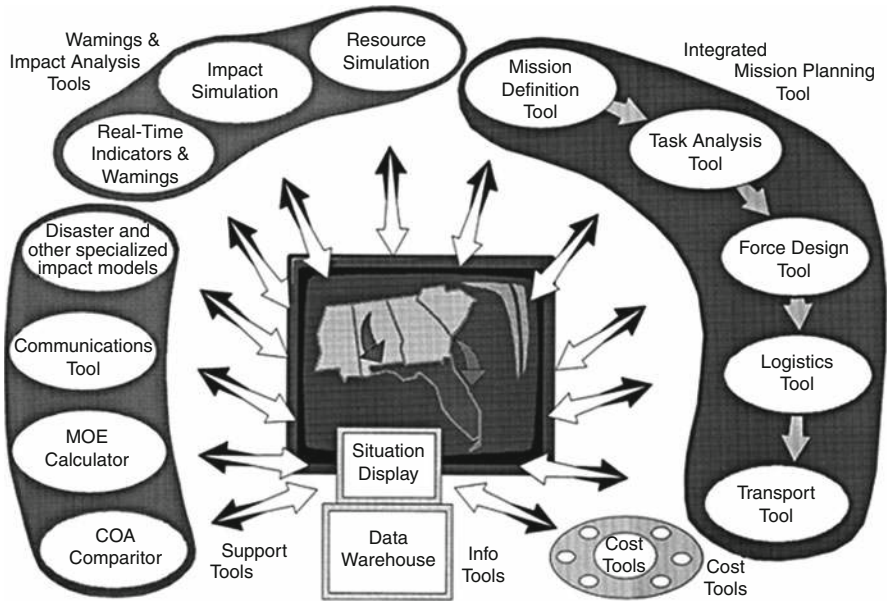
Clumsy as the OOTW designation may be, it is accurate: operations (as opposed to training activities) that are not war are included and operations that are part of a war are not included. Strictly speaking, the people who are using the designation are Department of Defense people and the operations so designated are military operations, leading to a preference for the term MOOTW; however, henceforth the shorter term OOTW will be used, because most of these operations are not led by the military, but by the State Department or some other agency. Figure 1 organizes OOTWs into categories.

The discussion of OOTWs suffering from an identity crisis is more than just a pleasant exercise in rhetoric. The underlying diversity of activities subsumed in the category creates a problem in defining standing operating procedures (SOPs) for dealing with them. The subordinate role of the military creates problems in planning for and executing them. The variability of participation of other federal agencies, other governments, the United Nations, non-governmental organizations (NGOs), and private volunteer organizations (PVOs) exacerbates the problem. Their ad hoc nature means that they are not included in the military's budget; the accounting systems are not designed to capture the costs; and recovering the resulting costs is problematic. These problems would be less troublesome if OOTWs were infrequent; however, since 1990 they have been undertaken at a rate of 20–35 per year! Over the past several years, there has been an increasing recognition of the need for analysis tools to support military planning and execution of OOTWs. Analysis tools to support decision making for large-scale military combat operations (such as major regional contingencies) are relatively mature (Battle Modeling). In contrast, OOTW analysis tools are embryonic or non-existent. The increasing U.S. military involvement in OOTWs during the post-Cold-War era has led to the need to develop OOTW analysis tools.

Questions

The analytical requirements are characterized by the questions that must be answered. The questions fall into five groups:

- Those that are non-mission-related (e.g., what force structure, equipment and plans are needed for the future?).
- Those that support a decision to engage (or not to engage) in a mission (e.g., what impacts will an OOTW have on other operations and how much will it cost?).



Military Operations Other Than War, Fig. 2 OOTW Analysis tool category

- Those needed to plan a mission (e.g., what is the right force structure and what transport support will we provide to reporters, NGO/PVOs, etc.?).
- Those that occur during a mission (e.g., which course of action will most quickly accomplish the mission?).
- Those related to the termination of a mission (e.g., how do we define success and what are its Measures of Effectiveness (MOEs)?).

The question groups are identical to the question groups for combat analysis. Most of the individual questions are also identical. In general, the analysis techniques required to answer the questions are the same. The problem lies in the application: standard applications make assumptions that are valid for combat analysis and invalid for OOTW analysis.

The question of force structure for a mission provides a simple example of the difference between combat analysis and OOTW analysis. For a combat mission, combat troops and equipment are determined first and the balance of the force structure is composed of the troops and materiel required to support them. Analysis procedures and tools are structured to support this situation. For an OOTW, however, the primary forces may be engineers for disaster reconstruction, medical personnel for disease control, some other support function, or combat troops, depending on the particulars of the mission. The implied force structure consists of the troops and materiel to support these forces and may (or may not) include combat troops to protect them. Not only are combat analysis procedures and tools set up backwards for OOTW analysis, but also OOTW analysis involves multiple possible permutations, requiring significantly more flexibility.

Nature of The Analysis Tools

Generally, the desirable tools are decision support tools, are simple (e.g., menu driven, point and click), are deployable, are joint (multi-service), are rigorous, use non-parochial data, have available data, and are capable of rapid turnaround. Analysis tools range from complex simulations of political, economic, sociological, military interactions to database tools, to spreadsheets, to checklists, with the emphasis on small tools. Figure 2 shows the categories of OOTW analysis tools.

Warnings and Impact Analysis Tools

These tools are among the most difficult (scientifically) to create, but are essential to the analysis of OOTWs. Three tools are included in this group.

- The real-time indicators and warnings tool serves to filter and interpret world news in the light of

possible future OOTWs: there are several attempts being made to create such a tool, such as the Protocol for the Assessment of Nonviolent Direct Action (PANDA) (Bond and Vogele 1995).

- The impact simulation models the significant relationships included in and surrounding an OOTW to permit prediction of the results of actions, whether human or environmental: the commercial computer game, Sim City™, is an example of an impact simulation. Unfortunately, the nature of social interactions is a matter for debate and consequently the proper mathematical expressions of these interactions and the best methods for modeling them are undecided. While at least two candidate simulations exist, Spectrum (National Simulation Center 1996) and the Deployable Exercise Support system/Civil Affairs Module (DEXES/CAM) (Woodcock 1996), these are regarded with some misgivings by working analysts, apparently because of lack of transparency or because they are used for training. The Situational Influence Assessment Module (SIAM) uses another technique to address social interactions. It is an influence diagram-based model, not a simulation model, but may be useful in this category.
- The resource simulation models the changes in resource consumption and sequestration over the course of an OOTW: this need may well be satisfied by the Joint Warfare Simulation (JWARS).

Integrated Mission Planning Tool

The five separate tools that comprise this group should ultimately be seamlessly integrated, although the initial integration may be loose. Each tool feeds its successor, while permitting reentry for iterative planning. These tools are relatively simple (scientifically); however, to be useful in an OOTW context, they require careful definition with respect to applicability to joint, coalition (multi-country) and non-military component analysis. The tools are a mission definition tool, a task analysis tool, a force design tool, a logistics tool, and a transportation tool.

- The mission definition tool should provide a reality check to ensure that the complete implications of the mission are fully understood. The Conceptual Model of Peace Operations (CMPO), a peace operations influence diagram-based checklist, is an example (Davis 1996).
- The object of the task analysis tool is to support an accurate and complete analysis of the mission tasks. The tool needed is a decision support tool that connects missions to strategies to tasks, both explicit and implied, in the OOTW domain. It should identify both those tasks that are central to the mission and any contingent tasks that might be implied by reasonable shifts in mission definition. It should also support replanning as the situation changes. Lidy (1998) has produced the data to support such a tool.
- The object of the force design tool is to support the designation of U.S. forces required for an operation in an OOTW context. The tool needed is a decision support tool that connects the tasks to generic resources and connects generic resources to actual available resources, including U.S. military, U.S. non-military, foreign government, NGO/PVO, and contractor resources. Data requirements include task capability for all resources (or the facility for user input of unique resources) and availability data (based on reserve commitments, etc.). It should provide for restrictions on choices based on cultural issues. Processing should include selection of military resources and substitution of other resources. The tool should also support replanning as the situation changes.
- The object of the logistics analysis tool is to support the logistics analysis of the mission in an OOTW context. The tool needed is a decision support tool that derives the logistics requirements from the total force structure. It should allow for supply from outside sources and provide for supply of non-military personnel. It should support replanning as the situation changes. Recent work has investigated the availability and utility of existing tools of this type (Brundage et al. 1998).
- The object of the transport analysis tool is to support the transportation analysis for mission arrival, sustainment, and departure in an OOTW context. The tool needed is a decision support tool that plans the transport requirements, based on all appropriate constraints. It must support replanning when the situation changes after some transport has been accomplished. The Joint

Flow and Analysis System for Transportation (JFAST) and the Model for Intertheater Deployment by Air and Sea (MIDAS) are examples of this type tool.

Support Tools

This group contains three specific tools and a cluster of several tools related by type. The *COA* comparitor permits the development of courses of action (COAs) through several levels of alternatives: an influence diagram/decision tree methodology would support this type analysis. The MOE calculator supports the calculation and tracking of MOE values. The communications tool supports planning the communications system within the complex context of OOTWs. The cluster of disaster impact tools (e.g., hurricanes, volcanoes, earthquakes, fires, and nuclear accident) supports the estimate of the situation in several technical areas, such as engineering and health. The Consequence Assessment Tool Set (CATS) supports some of these functions.

Cost Models

Seven tools make up this group. Their object is to calculate the cost information for various aspects of OOTWs: incremental costs of notional OOTWs, to support long-term analysis; probable incremental costs, to support the decision on engaging in a particular OOTW; relative (full) costs, to support the selection of the mission plan; costs incurred, to support cost recovery from other U.S. agencies and from foreign organizations and governments; incremental costs of a particular OOTW, to support the Congressional Budget process; costs of a particular OOTW, including equipment depreciation, readiness losses, increased reserve recruitment and training costs, and perhaps other costs, to support future acquisition, budgeting and training decisions; and actual costs of a completed OOTW, to support improved estimates of future operations and reports to Congress on actual costs. Work is underway to address analysis tools (Institute for Defense Analyses 1998; Hartley and Packard 1998b).

Information Tools

There are two tools in this category. The situation display presents the information concerning the situation in a manner designed to maximize understanding: the Virtual Information Center (VIC) project represents a first attempt at creating this type tool (Sovereign 1998). The data warehouse either stores or provides links to (as appropriate) all pertinent data. The data and their useability are critical to good analysis in the OOTW domain, as well as in the combat domain. However, the data required for OOTW analysis and the display requirements are in an embryonic state when compared to the state of affairs of combat analysis.

Tool Definition Process

Analysis of OOTWs is a new field and is in a state of flux. The first concerted effort to address the need for analytic tools is documented in Hartley (1996). Follow-on efforts are documented in Staniec (1998), Hartley and Packard (1998a), Brundage et al. (1998), Lidy (1998), Sovereign (1998), Hartley and Packard (1998b), and Hartley and Packard (1999).

See

- ▶ Analytic Hierarchy Process
- ▶ Battle Modeling
- ▶ Cost Analysis
- ▶ Crime and Justice
- ▶ Econometrics
- ▶ Economics and Operations Research
- ▶ Global Models
- ▶ Health Care Management
- ▶ Influence Diagrams
- ▶ Logistics and Supply Chain Management
- ▶ Military Operations Research
- ▶ Operations Management
- ▶ Production Management
- ▶ Public Policy Analysis
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Supply Chain Management
- ▶ System Dynamics

References

Bond, D., & Vogele, W. B. (1995). *Profiles of international "Hotspots"*. Harvard, Cambridge, MA: Center for International Affairs.

Brundage, W., et al. (1998). *Analysis of U.S. involvement in multiple small scale contingencies — Failed state*, OSD (PA&E).

Davis, D. F. (1996). Peace operations analysis with Bayesian belief networks. In *13th International symposium on military operational research (ISMOR)*, England.

Hartley, D. S., III. (1996). *Operations other than war: Requirements for Analysis Tools Research Report, K/DSRD-2098*. Oak Ridge, TN: Lockheed Martin Energy Systems, Inc.

Hartley, D. S., III, & Packard, S. L. (1998a). *OOTW tool requirements in relation to JWARS, K/DSRD-3076*. Oak Ridge, TN: Lockheed Martin Energy Systems, Inc.

Hartley, D. S., III, & Packard, S. L. (1998b). *OOTW cost tools, Y/DSRD-3099*. Oak Ridge, TN: Lockheed Martin Energy Systems, Inc.

Hartley, D. S., III, & Packard, S. L. (1999). *OOTW mission planning, Y/DSRD-3117*. Oak Ridge, TN: Lockheed Martin Energy Systems, Inc.

Institute for Defense Analyses (1998). Contingency operations support tool web site, http://www.ida.org/COST.

Lidy, A. M. (1998). *United States military role in smaller scale contingencies, D2166*. Alexandria, VA: Institute for Defense Analyses.

National Simulation Center (1996). *Information Paper*. http://www.leav.army.mil/nsc/famsim/spectrum/infopapr.htm.

Sovereign, M. (1998). *Humanitarian assistance and disaster relief in the next century* (Workshop Report). Arlington, VA: CCRP, National Defense University.

Staniec, C. (1998). *MORS workshop on OOTW analysis and modeling techniques (OOTWAMT)*. Alexandria, VA: Military Operations Research Society.

Woodcock, A. E. R. (1996). Modeling and analysis of societal dynamics: The deployable exercise support (DEXES) system. In A. Woodcock & D. Davis (Eds.), *Analytical approaches to the study of future conflict*. Clemensport, Nova Scotia: The Lester B. Pearson Canadian International Peacekeeping Training Centre.

Military Operations Research

Brian R. McEnany and Robert S. Sheldon
Military Operations Research Society (MORS), Alexandria, VA, USA

Introduction

To say that Military Operations Research (MOR) is the application to military operations of the methods of operations research (OR) is strictly correct, but gives only one clue to understanding the subject. The MOR accomplishments in World War II, sketched below, pioneered and greatly influenced the early development and institutionalization of operations research generally. Also, they led to the continuation of MOR after the war, in the governments of World War II participants, in academia, in industry, in not-for-profit think tanks, and its adoption in similar institutions of other nations. The emphasis in this article is on practice and trends in the United States, with particular emphasis on the Army.

The general methods of OR apply in particular to many aspects of military applications. Such differences as exist pertain mainly to the needs of military security and classification procedures, the nature of military operations and equipment, and the concerns of strategy, operational art, and tactics that relate to the use of military forces as instruments of national policy.

Current developments in the field are described in the quarterly bulletin *Phalanx* and the journal *Military Operations Research* published by the Military Operations Research Society (MORS) and the Military Applications Society (MAS) of INFORMS. MORS also conducts annual classified symposia, as well as smaller mini-symposia and workshops (some unclassified), from which they publish proceedings and monographs.

World War II MOR Accomplishments

Although there were individual contributions to the scientific study of military operations, ranging from Archimedes to the work of Thomas A. Edison in World War I, it was in World War II that MOR became widespread and institutionalized. Solandt (1955) recalled that MOR began in the services in England as operational research in the early days of the war. The British work centered about different subjects depending on the service: in the Air Force it was the problem of how to use radar, in the Navy it was the problem of anti-submarine warfare, and in the Army it was first limited to anti-aircraft problems and again centered around radar. Professor Blackett is sometimes said to have started the work in all three services, and his account in Blackett (1962) drew on earlier papers to describe both results and methods.

Schrader (2006) describes the organization and use of OR by the U.S. Army from WWII until 1995. His detailed account of how and where OR was used represents a definitive study of the U.S. Army's use of OR in peace and war, and much of what is summarized here is based on his writings.

Cooperation between the United States and Britain over the use of OR did not begin immediately during WW II. British liaison teams visited the U.S., but it was not until late in 1940, just after the fall of France, that President Roosevelt authorized the creation of the National Defense Research Committee (NRDC) and subsequently, the Office of Scientific Research and Development (OSRD) under Professor Vannevar Bush. This office helped recruit, manage and organize the military OR effort in the U.S defense establishment during the war.

While Britain fielded OR teams and detachments with its Army and Navy during the war, only the U.S. Navy and the U.S. Army Air Force (AAF) took full advantage of the new discipline after Pearl Harbor. OR teams of scientists and businessmen, recruited and organized through the OSRD, formed the initial groups. Small detachments were sent to AAF to conduct bombing accuracy studies and assessments of tactics. A useful account of World War II MOR, centering about the AAF, is Brothers (1954). In addition to illuminating examples such as aerial bombing accuracy improvement, it gives valuable guidance on the organization of MOR groups and operating procedures. In World War II, most of the MOR practitioners were civilians (though sometimes in uniform), and they had to earn the trust of military operators over time through useful work. This, of course, is by no means unique to MOR in World War II.

The U.S. Army's Technical Services – the scientific branches (Ordnance Department, the Medical Services, Signal Corps and Chemical Warfare Service) took advantage of the expertise offered by the new multi-disciplinary teams and detachments were deployed in Europe and in the U.S. The Army ground forces, on the other hand, were reluctant to begin using operational analysts (or "Op Annies" as they were called) until 1944. Teams were primarily used to support anti-aircraft weapons development and support to U.S. Army forces in the Pacific area.

The AAF was quick to emulate its British comrades and OR teams were soon supporting the various major Air Force operations in Europe and elsewhere. The Army's technical services were slower, but before the end of the war, studies in support of radar training, development, and organization, signal work load in message centers, transportation scheduling, loading, and handling, as well as some operational studies involving introduction of new equipment and technology to units were undertaken. The ground forces lagged well behind until late in 1944 when OR teams were sent to the Pacific.

At the end of the war, the rapid demobilization of the U.S. Army dissolved its existing teams and organizations as civilian scientists quickly returned to their academic or business careers. The national offices, NRDC and OSRD, were also demobilized, but the newly organized Department of Defense (DoD) created the Weapon Systems Evaluation Group (WSEG) to carry on work begun earlier. The limited use of OR in the Army's decision-making process during the war lagged well behind the other services. In the postwar period, the civilian leadership recognized the benefit provided by the studies and analysis of weapon systems and their development. The ground Army quickly closed the gap in the postwar period.

Early in the post-war period, Morse and Kimball (1946) drew on the work of many early MOR analysts of the Operations Research Group, U.S. Navy, to give results and methods. That work, once it was declassified and slightly modified, was republished in 1951 and was very influential, not only in introducing MOR to future analysts, but also in introducing the potential applications of OR generally to a wider audience. This Morse and Kimball classic was republished by MORS in 1998.

The above work quotes a letter from Admiral King that enumerated helpful MOR applications (suggestive also of the work in other services):

(a) The evaluation of new equipment to meet military requirements.
(b) The evaluation of specific phases of operations (e.g., gun support, anti-aircraft fire) from studies of action reports.
(c) The evaluation and analysis of tactical problems to measure the operational behavior of new material.
(d) The development of new tactical doctrine to meet specific requirements.
(e) The technical aspects of strategic planning.
(f) The liaison for the fleet with the development and research laboratories, naval and extra-naval.

Morse and Kimball also gave some reasons for the emergence in World War II of the practical value of the methods of MOR. As opposed to earlier wars there were the following:

(i) more repetitive operations susceptible to analysis — strategic bombing, submarine attacks on shipping, landing operations, etc.;
(ii) increased mechanization of warfare, in that "... a men-plus-machines operation can be studied statistically, experimented with, analyzed, and predicted by the use of known scientific techniques just as a machine operation can be."
(iii) increasing tempo of obsolescence in military equipment ... When we can no longer have the time to learn by lengthy trial and error on the battlefield, the advantages of quantitative appraisal and planning become more apparent."

Post-War MOR Developments

After World War II ended, a majority of the MOR practitioners returned to non-military pursuits: universities, laboratories, industry, etc. The military services wondered how much MOR would be needed in peacetime. Each decided to institutionalize its use of MOR. An early chapter of Tidman (1984) gives an interesting account of how the Navy chose to continue MOR by establishing the Center for Naval Analysis (CNA) after World War II and by 1948, each service had a different choice or mix of civil service groups, not-for-profit groups, use of industry, etc., and their emphases varied over time. The newly organized U.S. Air Force soon created Project RAND (later RAND Corporation) in 1948 to support its research and development efforts. The newly formed DoD followed suit with establishment of WSEG. Fairly soon, as the Cold War emerged, there was general recognition that it would be necessary to increase the use of MOR. Both Tidman and Schrader appropriately addressed this topic as periods of consolidation and growth in their respective histories.

The Army rapidly demobilized after the war, as stated above, the civilian scientists quickly returned to their jobs and homes. While the Army ground forces quickly inactivated its MOR organizations, the technical services (Ordnance and Signal) retained theirs. By 1948, the Army's leadership created a relationship with John Hopkins University under Dr. Ellis Johnson to form the Operations Research Office (ORO), a relationship that was to last for 13 years. World War II had seen the introduction of radar, atomic weapons, cruise missiles, and ballistic missiles, but each type was still improving rapidly at war's end. Their implications for, and fuller integration into, military forces needed more thought. The Cold War climate also provided a sense of urgency, and MOR offices took on these problems as important foci of effort. The growing Cold War with the Soviet Union forced the Army to address more than just weapons design and tactical doctrine. ORO soon began addressing areas well beyond weapons development – entering international politics, economics, national policy and global strategy while the technical services and newly organized field force boards maintained their focus on weapons development. Several key MOR organizations were created – ORO, Combat Operations Research Group (CORG), the Human Resources Research Organization (HumRRO), and Special Operations Research Office (SORO) dealing with psychological operations. Computer modeling of complex systems met increased need to process large quantities of data. At Headquarters, Department of the Army (HQ DA), the Strategic Tactics and Analysis Group (STAG) was formed to study force structure and future forces capability through gaming and simulation. The increasing use of MOR in the combat development process fostered a need for increased numbers of military officers with MOR training and a formal Operations Research/Systems Analysis (ORSA) specialty program was created in 1967 to satisfy the growing need to form in-house MOR capabilities as the Army moved toward a competitive contractual arrangement with various commercial and academic analytic groups. The Research Analysis Corporation (RAC) took over as the primary research arm of the Army staff in 1963 while primary research efforts were funneled into academia through the Army Research Office (ARO) at Duke University.

Some of the postwar applications of MOR resembled wartime MOR, with combat operations replaced by tests or exercises. With the rise of the Continental Army Command (CONARC), MOR organizations began efforts involving war gaming and field experimentation. As technology increased and problems became more complex, recommendations soon increased the amount of field

experimentation and testing, and by 1956, the first combat development and testing command was created. However, some of the OR (or operations analysis or operations evaluation, as it was often termed) *remained* devoted to operations of supply, logistics, recruiting, and training. Moreover, much of the post-war efforts went into thinking through the implications of new weapons for new types of combat operations. It fostered an atmosphere that led to increased use of digital computing capabilities in war gaming and simulation to help solve increasingly more sophisticated and complex problems.

The Emergence of Systems Analysis

MOR also took on problems at a level higher than that of individual weapon systems or engagements between two opposing weapon systems. Even in a Cold War climate, there were significant limits on national expenditures for armed forces. It was necessary for government to decide "how much is enough" and MOR sought to aid this decision.

Applications of OR at this high level, often termed systems analysis, face difficulties far greater than the difficulties of World War II MOR, significant as the latter were. Wartime combat analysis, sometimes without recognizing it, had already faced criterion problems of sub-optimization, as Hitch (1953) points out. These become still more significant when structuring forces for the future, seeking to be prepared to deal with contingencies still beset with great uncertainty.

Hitch (1955) gives an understanding of the relative difficulty of systems analysis by comparing the World War II problem of improving bomber accuracy with the postwar problems of weapon system development and force composition. In the former problem, difficult as it seemed at the time, known were the types of aircraft involved, how many there were, much about their characteristics, the kind of bombs available, and much about enemy targets and their defenses. These become variables when considering an uncertain future that may sometimes hold a multiplicity of potential opposing forces.

The difficulties are in the problems, as Hitch went on to point out. Despite these difficulties, governments must make decisions and systems analysis, with all of its limitations, has much to offer. MOR analysts developed judgment in cutting problems down to size, and Quade (1954) collected some of the helpful approaches in an influential volume. Quade and Boucher (1968) and Miser and Quade (1988) give refinements and extensions to non-defense analysis.

The Institutionalization and Impact of Systems Analysis

Hitch and McKean (1960) did much to introduce cost-effectiveness studies as instruments of defense systems analysis. In the Kennedy administration in 1961, Secretary of Defense McNamara brought Hitch into the Office of the Secretary of Defense (OSD) as Comptroller to install a system of planning-programming-budgeting (PPB), and Enthoven, as Hitch's assistant, started an office of systems analysis. Although the titles and organizational placement have changed over the years, OSD has continued both PPB and systems analysis.

These new offices had great impact. The government sought to create similar offices in other departments (Bureau of the Budget 1965). Within the DoD, the new OSD offices played an important role in departmental decisions. As its emphasis on, and requests for, quantitative analysis increased, the military services organized and enlarged their MOR offices to meet the demand.

The above developments came at a time when computer capabilities were rapidly increasing. Many MOR offices sought to use the new capabilities in producing cost-effectiveness studies required for systems analysis. Computer simulation models began to proliferate in the effort to understand what new or proposed weapon systems would contribute to the future battlefields. Because this effort contributed to studies with great impact on weapon systems acquisition, it has continued to grow.

Wartime Combat OR in Korea and Vietnam

Although its successes in World War II led service leadership to gradually incorporate MOR into its decision-making process, MOR efforts came to emphasize future weapon system acquisition as described above. For more details of what is summarized here, see Schrader (2008).

KOREA: Right from the beginning, the Army leadership was admonished to deploy MOR teams to Japan and Korea. As in WWII, the analysis of current operations, organizations and tactics were prominent. Increased interest in new organizations, counter-measures, winter operations, clothing, airborne operations, and psychological warfare were undertaken. By the end of 1953, the efforts of the deployed MOR teams had validated WWII experience and demonstrated that MOR could be successfully applied to land warfare. Between Korea and Vietnam, multiple MOR organizations provided analyses and supported the combat development process. Major war gaming and simulation centers were created to study the impact of new weapons, organizations, and future force structure and tactics. Centers grew within The U.S. Army Training and Doctrine Command (TRADOC) at Fort Leavenworth and White Sands to assist in defining new organizations, doctrine and tactics while HQ DA continued to rely upon the successor to STAG, the Concepts Analysis Agency (CAA), to evaluate future force structures. Individual weapons research and evaluation continued to expand at Aberdeen Proving Grounds where the Ballistics Research laboratory (BRL) studied, evaluated, assessed and developed new, improved weapons.

VIETNAM: While a shooting war was underway in Southeast Asia (SEA), the rise of PPB at the Pentagon split Army MOR activities. It developed additional in-house capability to support the centralization of decision-making begun under Secretary McNamara and the Office, Secretary of the Army (OSA). More MOR trained personnel were needed to support the PPB System (PPBS) and a formal specialty program was created in 1967 for military officers. This was coupled with use of civilian contractors and Federally Funded Research and Development Centers (FFRDCs). That is not to say that MOR activities were totally devoted to PPBS. Of particular interest was the lengthy analysis and assessment of the air mobility concept and organization of the air assault division prior to the war in SEA. Multiple organizations, field boards and MOR offices significantly supported the vast testing and experimentation of the air mobility concept.

The war in SEA renewed interest in the study of current operations, battlefield performance of weapons, equipment, organizations and tactics. RAC, the successor to ORO, deployed teams to to collect data along with HumRRO, SORO and Combat Development Command (CDC). HQ Military Assistance Command, Vietnam (MACV) established an in-theater analysis and assessment capability. Quantitative methods were employed extensively at Field Force and Division level. Manually assisted war games were run to help develop alternate strategies and think through potential issues. Efforts were focused upon counter-insurgency operations and suffered from lack of large amounts of quantitative data needed to adequately analyze it. Still, as one division commander noted, the "judicious use of operational analysis and analytic techniques when melded with military judgment were quite effective in improving performance of many activities."

In Chapter I of Hughes (1989), Thomas observed that combat OR both in Korea and later in Vietnam was very similar to that of World War II. Despite the postwar increase in modeling and computer capabilities, it did not make nearly as much contribution in Korea or Vietnam as might have been expected. "Though the menu of available techniques increased with time, much that had been learned in World War II was forgotten and relearned in later conflicts." The 1960s and 1970s were a time of great growth in the analytic community. MOR efforts greatly expanded force planning and management with a commensurate need to expand the number of MOR-trained officer personnel. A whole new set of challenges faced the Army after Vietnam as the MOR community assisted in helping the Army reorganize, revitalize, and reorient itself prior to the First Gulf War.

Contributions After Vietnam and the Gulf War

The period after Vietnam was a time of recovery and reorganization for the U.S. Army (see Schrader 2009 for more details). The multi-year conflict had severely damaged the Army's equipment modernization process and MOR efforts concentrated upon providing the analytic underpinning for major changes in weapon systems, equipment, organizations, doctrine and training. In light of two major studies affecting MOR organizations, competitive contracting was more formalized

(RAC was disestablished) and MOR assets became more concentrated into fewer organizations – CAA, TRADOC Analysis Command (TRAC), the Operational Test and Evaluation Command (OPTEC) and the Army's Material Systems Analysis Agency (AMSAA) ultimately concentrated the efforts of the majority of civilian and military MOR specialists and performed the majority of all studies. MOR became more integrated into the Army's decision-making process as new technology, better weapon systems, and improved organizations were developed. A pyramid of responsibility was formed with CAA at the apex studying force structure and strategy, TRAC focused on battalion to Corps level studies, and AMSAA dealing with individual weapon system analysis.

The end of the Cold War in 1989 presented the Army and the MOR community with entirely new issues – much more complex and demanding that ever before – and MOR support to the material acquisition process became more important. The ever-increasing improvements in technology and computing power brought with it an expanding use of models and simulations to solve the issues facing the Army. This expansion also created issues in validation, verification and accreditation of the analytic tools used to support the decision making process.

During the First Gulf War in 1991, the efforts of 20 years of MOR involvement in conjunction with new organizations, new equipment and weapon systems, new doctrinal, and training improvements, fielded the finest fighting force in the history of the United States. Each of the major organizations actively supported the collection of data. CAA was intimately involved in the evaluation of the forces involved during the planning phase of the operation. War games and separate assessments assisted Army planners and major headquarters in preparing for the deployment and employment of forces. Multiple rapid response assessments – some as short as 12 h – were provided during Operation Desert Shield. Ultimately, a small MOR cell was deployed to support HQ Central Command (CENTCOM), but most MOR efforts were conducted in the continental U.S. (CONUS). The successful military outcome underscored the need for rapid and flexible support to deployed forces with a full range of theater level analysis capabilities.

MOR Lessons from Desert Shield/Desert Storm

The new computer and modeling capabilities seemed to have more impact in MOR for the Gulf War combat of 1991. Vandiver et al. (1992) concluded that while some of its analytic lessons were reminiscent of World War II, and some lessons were probably peculiar to wars like the Gulf War, there were trends indicative of future combat analysis:

- Computer influence on analysis is increasingly varied and pervasive.
- Software analytical tools are increasingly available to all - including non-analysts.
- The demand for good databases is growing more rapidly than the supply.
- There is growing need for coalition and joint service analysis.
- There is increasing analytical interest in operational art and campaign focus.
- There is a need to have MOR teams ready to join, and planning models and simulations in place with deployed forces.
- Teams must be ready to improvise quickly to support ongoing and planned operations in the field.
- There is less danger of central misuse of field analysis and data than formerly. The lessons of better methods of data collection and selection of more accurate measures of effectiveness learned in earlier conflicts have been absorbed by the MOR community.

Concluding Remarks

Although MOR has been a flourishing enterprise with an expanding technological menu, there are still issues to resolve, some long standing. While it is clear that MOR tools and techniques improved the material acquisition process and the PPBS, a significant fraction of the issues relate to modeling and simulation, or are frequently so characterized. Some of the more serious concerns address scientific foundations (including verification and validation); DoD organization and management (including that for MOR); management; filling a perceived need; and taking suitable advantage of technological opportunities.

See

- ▶ Air Force Operations Research
- ▶ Battle Modeling
- ▶ Center for Naval Analyses
- ▶ Cost Analysis
- ▶ Cost-Effectiveness Analysis
- ▶ Exploratory Modeling and Analysis
- ▶ Military Operations Other Than War
- ▶ Operations Research Office and Research Analysis Corporation
- ▶ RAND Corporation
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Systems Analysis
- ▶ War Game

References

Blackett, P. M. S. (1962). *Studies of war*. New York: Hill and Wang.

Brothers, L. A. (1954). Operations analysis in the United States Air Force. *Operations Research, 2*, 1–16.

Bureau of the Budget. (1965). Planning-programming-budgeting. Bulletin 65-5, Washington, DC.

Hitch, C. J. (1953). Sub-optimization in operations problems. *Operations Research, 1*, 87–99.

Hitch, C. J. (1955). An appreciation of systems analysis. *Operations Research, 3*, 466–481.

Hitch, C. J., & McKean, R. N. (1960). *The economics of defense in the nuclear age*. Santa Monica, CA: RAND. R-348.

Hughes, W. P., Jr. (Ed.). (1989). *Military modeling* (2nd ed). Alexandria, VA: MORS.

Miser, H. J., & Quade, E. S. (Eds.). (1988). *Handbook of systems analysis: Craft issues and procedural choices*. New York: North-Holland.

Morse, P. M., & Kimball, G. E. (1946). *Methods of operations research*. OEG Rpt. 54, Office of the Chief of Naval Operations, Navy Dept., Washington, DC.

Quade, E. S. (Ed.). (1954). *Analysis for military decisions*. Santa Monica, CA: RAND. R-387-PR.

Quade, E. S., & Boucher, W. I. (Eds.). (1968). *Systems analysis and policy planning: Applications in defense*. Santa Monica, CA: RAND. R-439-PR.

Schrader, C. R. (2006). *History of operations research in the US Army:1942-1962*. Carlise, England: Center for Military History. Pub70-102-1.

Schrader, C. R. (2008). *History of operations research in the US Army:1961-1973*. Carlise, England: Center for Military History. Pub70-105-1.

Schrader, C. R. (2009). *History of operations research in the US Army:1973-1995*. Carlise, England: Center for Military History. Pub70-110-1.

Solandt, O. (1955). Observation, experiment, and measurement in operations research. *Operations Research, 3*, 1–14.

Tidman, K. R. (1984). *The operations evaluation group, a history of naval operations analysis*. Annapolis, MD: Naval Institute Press.

Vandiver, E. B., et al. (1992). Lessons are learned from desert shield/desert storm. *PHALANX, 25*(1), 6–87.

MIMD

Multiple instruction, multiple data. A class of parallel computer architectures in which each processing element fetches and decodes its own stream of instructions, possibly different from the instruction streams for other processors.

Minimum

A real-valued function $f(x)$ is said to have a minimum on a set S when the greatest lower bound of $f(x)$ on S is assumed by $f(x)$ for some x^0 in S. Thus, $f(x^0) \leq f(x)$ for all x in S.

See

- ▶ Global Maximum (Minimum)

Minimum (Maximum) Feasible Solution

In a mathematical-programming problem, the solution that both satisfies the constraints of the problem and minimizes (maximizes) the objective function is a minimum (maximum) feasible solution. Such solutions may not be unique.

Minimum Spanning Tree Problem

Given a connected network with n nodes and individual costs associated with all edges, the problem is to find the least-cost spanning trees.

See

- Network Optimization
- Spanning Tree

Minimum-Cost Network-Flow Problem

In a directed, capacitated network with supply and demand nodes, the problem is to determine the flows of a single, homogeneous commodity from the supply nodes to the demand nodes that minimize a linear cost function. In its general form, when the network contains transshipment or intermediate nodes – nodes that are neither supply nor demand nodes – the problem is called the transshipment problem. Conservation of flow through each node is assumed. Due to its special mathematical structure, this problem has a solution in integer flows, given that the data that define the network are integers. It is a linear-programming problem whose major constraints form a node-arc incidence matrix.

See

- Conservation of Flow
- Maximum-Flow Network Problem
- Network Optimization

MIP

- Mixed-Integer Programming Problem (MIP)

MIS

Management information systems.

See

- Information Systems and Database Design in OR/MS

Mixed Network

A queueing network in which some customers can enter and leave the network while others neither enter nor leave but cycle through the nodes endlessly. A queueing network in which the routing process contains at least one closed set of states for some types of customers but not others.

See

- Closed Network
- Networks of Queues
- Open Network
- Queueing Theory

Mixed-Integer Programming Problem (MIP)

A mathematical-programming problem in which the constraints and objective function are linear, but some of the variables are constrained to be integer valued. The integer variables can either be binary or take on general integer values.

See

- Binary Variable
- Integer and Combinatorial Optimization
- Linear Programming
- Mathematical Programming

Model

An idealized — abstract and simplified — representation of a real-world situation that is to be studied and/or analyzed. Models can be classified in many ways. A mental model is an individual's conceptual, unstated, view of the situation under review; a verbal or written model is a description of one's mental model; an iconic model looks like what it is supposed to represent (e.g., an architectural model of

a building); an analogue model relates the properties of the entity being studied with other properties that are both descriptive and meaningful (e.g., the concept of time as described by the hands and markings of a clock); a symbolic or mathematical model represents a symbolic representation of the process under investigation, e.g., Einstein's equation $E = mc^2$, a linear-programming model, or a computer simulation model.

See

- ▶ Descriptive Model
- ▶ Deterministic Model
- ▶ Linear Programming
- ▶ Mathematical Model
- ▶ Normative Model
- ▶ Predictive Model
- ▶ Prescriptive Model
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Stochastic Model

Model Accreditation

Saul I. Gass
University of Maryland, College Park, MD, USA

Model accreditation is an official determination that a model is acceptable for a specific purpose (Williams and Sikora 1991; Ritchie 1992). Accreditation certifies that the element being accredited meets given standards. For a model, accreditation must be done with respect to the model's explicit specifications and the demonstration that the computer-based model does or does not meet the specifications. This demonstration is the responsibility of the model developers, who must show that their work passes agreed-to user and developer acceptance tests. If the modeling process was done properly and was accompanied by appropriate documentation, accreditation of the model for its specified uses should follow.

Accreditation of a model must rely on a review and evaluation of its available documentation. Such an evaluation, usually done by an independent third-party, is made against various criteria to determine the levels of accomplishment of the criteria, in particular those of verification and validation. The review is made with a specific user and uses in mind. The review should produce a report that gives guidance to the user on whether or not the model in question can be used with confidence for the designated uses, that is, the model is or is not accredited for specific uses (Gass 1993).

The ideas, if not the general process behind model accreditation, have been accepted by modeling agencies within government and private industry, most notably by the U.S. Department of Defense (2009) in the context of modeling and simulation (see also Sargent 2005).

See

- ▶ Model Evaluation
- ▶ Model Management
- ▶ Practice of Operations Research and Management Science
- ▶ Validation
- ▶ Verification
- ▶ Verification, Validation, and Testing of Models

References

DoDI. (2009). *DoD modeling and simulation (M&S) verification, validation, and accreditation* (*VV&A*). DoDI 5000.61, December 9, 2009.

Gass, S. I. (1993). Model accreditation: A rationale and process for determining a numerical rating. *European Journal of Operational Research, 66*(2), 250–258.

Ritchie, A. E. (Ed.). (1992). *Simulation validation workshop proceedings (SIMVAL II)*. Alexandria, VA: Military Operations Research Society.

Sargent, R. G. (2005). Verification and validation of simulation models. In M. E. Kuhl, N. M. Steiger, F. B. Armstrong, & J. A. Joines, (Eds.), *Proceedings of the 2005 Winter Simulation Conference*, IEEE Press.

Williams, M. K., & Sikora, J. (1991). SIMVAL Minisymposium — A Report," Phalanx, *Bulletin of the Military Operations Research Society*, *24*, 2.

Model Builder's Risk

Probability of rejecting the credibility of a model when in fact the model is sufficiently credible.

See

► Verification, Validation, and Testing of Models

Model Evaluation

Saul I. Gass
University of Maryland, College Park, MD, USA

Model evaluation or assessment is a process by which interested parties, who were not involved in a model's origins, development and implementation, can assess the model's results in terms of its structure and data inputs so as to determine, with some level of confidence, whether or not the results can be used in decision making. Model evaluation encompasses: (1) verification, validation, and quality control of the usability of the model and its readiness for use, and (2) investigations into the assumptions and limitations of the model, its appropriate uses, and why it produces the results it does.

There are three reasons for advocating evaluation of models: (1) for many models, the ultimate decision maker is far removed from the modeling process and a basis for accepting the model's results by such a decision maker needs to be established; (2) for complex models, it is difficult to assess and to comprehend fully the interactions and impact of a model's assumptions, data availability, and other elements on the model structure and results without a formal, independent evaluation; and (3) users of a complex model that was developed for others must be able to obtain a clear statement of the applicability of the model to the new user problem area (Gass 1977a).

All procedures for evaluating a model are basically information gathering activities, with the detail and level of information being a function of the purposes of the assessment and the skills of the assessors. Specific evaluation approaches are given in Gass (1977a, b), Gass (1980), U.S GAO (1979), with an evaluation case study given in Fossett et al. (1991).

A model evaluation procedure and its objectives should be tailored to the scope and purposes of the model and will vary with the model, model developers, assessors, users, and available resources. Model assessment is an expensive and involved undertaking; all models need not be assessed. Model developers and users should recognize that by applying proper modeling management procedures, the burdens that evaluators of models have to contend with are alleviated greatly (Gass 1987).

See

► Model Accreditation
► Model Management
► Practice of Operations Research and Management Science
► Project Management
► Verification, Validation, and Testing of Models

References

Fossett, C., Harrison, D., Weintrob, H., & Gass, S. I. (1991). An assessment procedure for simulations models: A case study. *Operations Research, 39*, 710–723.

Gass, S. I. (1977a). Evaluation of complex models. *Computers and Operations Research, 4*, 27–35.

Gass, S. I. (1977b). A procedure for the evaluation of complex models. *Proceedings of the First International Conference in Mathematical Modeling*, 247–258.

Gass, S. I., (Ed.). (1980). *Validation and Assessment Issues of Energy Models* (National Bureau of Standards Special Publication 569, U.S. GPO Stock No. 033-003-02155-5). Washington, DC: U.S. Government Printing Office.

Gass, S. I., (Ed.). (1980). *Validation and Assessment Issues of Energy Models* (National Bureau of Standards Special Publication 616). Washington, DC: U.S. Government Printing Office.

Gass, S. I. (1983). Decision-aiding models: Validation, assessment, and related issues for policy analysis. *Operations Research, 31*, 603–631.

Gass, S. I. (1987). Managing the modeling process: A personal perspective. *European Journal of Operational Research, 31*, 1–8.

Ritchie, A. E., (Ed.). (1992). *Simulation validation workshop proceedings*, (*SIMVAL II*). Alexandria, VA: Military Operations Research Society.

U.S GAO. (1979). *Guidelines for model evaluation*. Washington, DC: GAO/PAD-79-17.

Willemain, T. R. (1995). Model formulation: What experts think about and when. *Operations Research, 43*, 916–932.

Model Management

Ramayya Krishnan[1] and Kaushal Chari[2]
[1]Carnegie Mellon University, Pittsburgh, PA, USA
[2]University of South Florida, Tampa, FL, USA

Introduction

The term model management was coined in the mid-1970s in the context of work on decision support systems (DSS) (Sprague and Watson 1975; Will 1975). An important objective of the DSS concept was to provide an environment in which decision makers could gain materially useful insights by interactively exercising OR/MS models. However, developing such an environment required principled solutions to problems of specifying, representing and interacting with models. This focus on models, and in turn on modeling, led to the study of model management, defined broadly to encompass the study of model representation, the set of operations facilitated by such representation at various stages of the modeling life cycle, and computer-based environments that facilitate modeling.

What follows is a brief review of work in two areas that have been actively studied in model management. First, work on languages to specify models, and on the development of techniques to facilitate operations that support modelers in both the pre-solution and post-solution phases of the modeling life cycle. Second, work on the representation of a collection of models (e.g., a model library) and the development of techniques to enable model selection and configuration. As with other information technology-based fields, model management has benefitted from the growth of Internet technologies. A detailed review of the implications for model management of the growth in Internet, and in particular the World Wide Web technologies, is in Bhargava and Krishnan (1998), and Bhargava, Power, and Sun (2007).

Model Management-I

Modeling languages — The need to represent a model in a notation that is easy to validate, verify, debug, maintain and communicate motivated the development of modeling languages (Fourer 1983). Prior to their development, the only computer-executable representation of a model was in an arcane format optimized for efficient solution (e.g., the Mathematical Programming System MPS format).

Current modeling languages provide a high-level symbolic notation to specify models. Solution operations can also be declared and all the required details of binding the model instance to the data structures required by solver done transparently. Further, this has greatly increased the productivity of model-based work.

Four principles have been articulated as essential to modeling language design (Bhargava and Kimbrough 1993; Fourer 1983; Geoffrion 1992a; Krishnan and Chari 2000). These are:

- *Model data independence*: requires the mathematical structure of the model to be independent of the data used to instantiate it. This permits model data to be modified in format, dimension, units or values without any modification to the model representation.
- *Model solver independence*: requires the model representation to be independent of the representation required by the solver. This permits more than one solver to be used with a given model. Further, it recognizes the fundamental differences in the requirements placed on model representations and representations required by the solver.
- *Model paradigm independence*: requires that the modeling language allow the representation of models drawn from different paradigms (e.g., mathematical programming and discrete event simulation).
- *Meta level representation and reasoning*: requires that the modeling language represent information *about* model components and models, in addition to their mathematical structure in order to enable semantic consistency checking.

Modeling languages incorporate these principles to varying degrees. Examples of modeling languages include spreadsheet-based languages such as IFPS (Gray 1987), algebraic modeling languages such as GAMS (Bischop and Meeraus 1982), AMPL (Fourer et al. 1990), and MODLER (Greenberg 1992),

relational modeling languages such as SQLMP (Choobineh 1991), graphical modeling languages such as NETWORKS (Jones 1991), and Model Graphs (Chari and Sen 1997), typed modeling languages such as ASCEND (Piela et al. 1992), and XML-based languages such as OptML, SNOML, FML, and MathML. A survey of XML-based representations can be found in Valente and Mitra (2007). New developments in algebraic modeling languages include extensions for constraint programming (Fourer and Gay 2002), and extensions for stochastic programming (Valente et al. 2009). The formal analysis of the semantics of typed modeling languages is in Bhargava, Krishnan, and Piela (1997). There is also an active market in commercial modeling languages and systems. A survey of these systems can be found in Sharda and Rampal (1995).

Two developments have had a significant impact on modeling languages. One is the seminal work on Structured Modeling (SM) (Geoffrion 1987). Developments and research directions are described in a survey of structured modeling (Geoffrion 1999a), and an annotated bibliography is given in Geoffrion (1999b). While previous work on modeling languages had sought to provide a computer executable representation of the notation traditionally used by modelers, SM defines a theory that treats models as hierarchical collections of definitional dependencies. This enables structured modeling languages to satisfy all the four design principles discussed above. While several languages have implemented SM, the most completely developed of these is SML (Geoffrion 1992a, b). The other important development is the embedded languages technique, which can be used to define an architecture of considerable generality for modeling environments. This technique is used to specify modeling languages, as well as information about the terms and expressions stated in these languages. The TEFA modeling environment (Bhargava and Kimbrough 1993) has been implemented using this technique.

Operations — The early work on model management focused on model solution. The objective was to transparently bind solution algorithms to model instances. As noted above, modeling languages have realized this objective. Model management research has since focused on operations required to support both pre-solution and post-solution phases of the modeling life cycle. Next, research related to a pre-solution phase, model formulation, and a post-solution phase, model interpretation, are described.

Model Formulation — Model formulation is the task of converting a precise problem description into a mathematical model (Krishnan and Chari 2000). It is a complex task requiring diverse types of knowledge. The appropriateness of a model depends on a variety of factors such as accuracy, tractability, availability of relevant data, and understandability. Model formulation research has primarily focused on the development of theory, tools and techniques to support the formulation of deterministic mathematical programming models. Work by Gassmann and Ireland (1996) has studied the formulation of stochastic mathematical programming models.

Using protocol analysis, detailed studies of the expert modeling process have been conducted and process models have been developed (Krishnan et al. 1992; Raghunathan et al. 1994). Domain-independent and domain-specific model formulation strategies have been implemented in model formulation support systems (Krishnan 1990; Ma et al. 1989; Raghunathan et al. 1994) and a variety of representation and (deductive) reasoning schemes have been investigated. Liang and Konsynski (1993) have also investigated alternative approaches such as analogical reasoning and case-based reasoning to implement model formulation systems. A principled approach to formulating mathematical programming models is in Murphy, Stohr, and Asthana (1992). A survey of this research is given in Bhargava and Krishnan (1993).

Model Interpretation — Model interpretation consists of a variety of techniques to help a modeler comprehend a model. These include parametric analysis, structural analysis, and structure inspection.

Parametric analysis has long been supported in model management systems. Spreadsheets routinely support *what-if* analysis and goal seeking. Modeling languages for mathematical programming implement the theory of sensitivity analysis.

The pioneering work on structural analysis is due to Greenberg on the ANALYZE system (Greenberg 1987). ANALYZE extracts model structures that cause exceptions such as redundancy and infeasibility in linear programming models. The stream of work begun with ANALYZE has been considerably extended. Guieu and Chinneck (1999) described work and a toolkit called Mprobe that analyzes

infeasibility in mixed integer and integer linear programming models. Sharda and Steiger (1996) presented work on applying inductive learning techniques to facilitate model analysis. Kimbrough and Oliver (1994) examined the issue of post-solution analysis for models other than linear programs and have at-tempted to fashion a solution along the lines of ANALYZE. An important feature of their approach is the analysis of the impact on the solution to a model when changes are made to the parameters of a surrogate model.

Piela et al. (1992) described the use of a browser to inspect the structure of a model. Dhar and Jarke (1993) and Raghunathan et al. (1995) examined the usefulness of recording the rationale underlying a model. The documented rationale is used to aid comprehension as well as to correctly and consistently propagate the changes made to the structure of a model. Work on analyzing assumptions associated with models and visualizing the structure as a graph is reported in Basu and Blanning (1998). More recently, model ontology and model schema developed using OWL, a web ontology language based on XML, has been used for model representation and interpretation (Bhrammanee and Wuwongse 2008).

Model Management-II

Model libraries —In contrast to the work reviewed in the previous section, the focus of this stream of research assumes the existence of a library of debugged and validated models. This has led to the study of issues such as the representation of model libraries and operations such as model selection and configuration.

Model Representation — Predominantly, models are abstractly represented as black boxes, i.e., as a set of named inputs and outputs. This is in contrast to the detailed representation of the structure of the model in the previous section. A variety of representations, including virtual relations (Blanning 1982) and predicate logic (Bonczek et al. 1978) have been used to represent models. Additional structure has been imposed on these representations. Mannino, Greenberg, and Hong (1990) proposed the use of categories such as model type, model template, and model instance to organize the collection of models in a library. A model type is a general description of a model class such as linear programming. A model template is a refinement of a model type such as a production planning LP model, and a model instance is an instance of a model template in which the source of values for each parameter has been declared. Model templates have been represented using key-value pairs and filter lists in (Chari 2002), as Web Services Description Language (WSDL) service descriptors (Madhusudan 2007), and as OWL (XML-based) model profiles (Bhrammanee and Wuwongse 2008). Metagraphs (Basu and Blanning 1994a; Basu et al. 1997), a specialized type of graph structure, has been the significant development in this area.

Model Selection — Model selection leverages the existence of previously developed models to create a model for a new problem. In addition to the set of inputs and outputs associated with a model, additional information such as model assumptions need to be represented. Mannino et al. (1990) described model selection operators that match, either exactly or fuzzily, the assumptions associated with a model and those that are part of a problem statement. Work by Banerjee and Basu (1993) adopted the same framework as Mannino et al. (1990) but differed in its use of structuring technique called the Box Structure method (Mills et al. 1986), borrowed from the domain of systems analysis and design to develop its taxonomy of model types. Later, Guenther, Muller, Schmidt, Bhargava, and Krishnan (1997) studied the problem of selecting models and methods from web-based electronic catalogs. Chari (2002), implemented an approach based on matching filter spaces in selecting models. More recently, the work by Guntzer, Muller, Muller, and Schimkat (2007) have used a graph-matching procedure for selecting structured models represented as graphs. The problem of selecting and composing appropriate data mining models from a model library is now gaining attention (Liu and Tuzhilin 2008).

Model Configuration — Model configuration leverages previously developed models by either linking them together (referred to as model composition) or by integrating them (referred to as model integration). Model composition links together independent models such that the output of one model becomes an input to another. Model composition is often used in conjunction with model selection when no one model meets the requirements of a problem.

An example of model composition is the linking together of a demand forecasting model and a production scheduling model.

While the early work only permitted links between variables with the same name, later work of Muhanna (1992) and Krishnan, Piela, and Westerberg (1993) permitted linkages between objects (variables, arrays, instances of types, etc.) as long as certain semantic constraints are met. Muhanna (1992) also proposed methods that determine the order in which a collection of linked models should be solved. Representation methods and algorithms that can determine the set of models that need to be composed in order to obtain a set of outputs from a given set of inputs have been a major focus of model composition research. While the early work was based on virtual relations (Blanning 1982) and predicate logic (Bonczek et al. 1978), later work based on a construct called metagraphs (Basu and Blanning 1994a) has shown considerable promise. In addition to model composition (Basu and Blanning 1994b), the metagraph construct enables the representation of and reasoning with metadata such as assumptions associated with models (Basu, Blanning and Shtub, 1998). Work within the last ten years has focused on automating model composition and execution process, and combining partial solutions from multiple composite models and databases as in Chari (2002), leveraging XML in model composition (Bhrammanee and Wuwongse 2008) and implementing model composition through a sequence of web service invocations as in the WEBOPT project (Valente and Mitra 2007), and in (Madhusudan 2007).

Model integration differs from model composition in allowing modifications to be made to the models being integrated. Model integration involves both schema integration and solver integration (Dolk and Kotteman 1993). Schema integration is the task of merging the internal structure of two or more models to create a new model, while process integration is the task of interweaving associated solution processes in order to solve the integrated model.

Support for conflict resolution is a major focus of research in schema integration. This has involved the development of a variety of typing schemes that seek to integrate data typing (Muhanna 1992), and concepts such as quiddity and dimensions (Bhargava et al. 1991).

Detailed procedures for integrating models specified in the Structured Modeling Language (SML) (Geoffrion 1992a, b) have been proposed (Geoffrion 1989) and extended (Tsai 1998). The method uses to advantage the ability of structured modeling to trace the effects of changes and the formal definition of what constitutes a structured model. An update to structured modeling research is given in Geoffrion (1999a).

The pioneering work on solver integration is the work of Dolk and Kotteman (1993). They used the theory of communicating sequential processes (Hoare 1985) to address the problem of solver integration. A simplified version of the problem was addressed by Muhanna (1992) in the SYMMS system. As software components have emerged as a viable technology for web-based deployment of solvers on the Web, recent work has studied integration of solvers/methods on the Web (Guenther et al. 1997). Technology has made it possible to wrap a solver with a software layer that exposes standard interfaces thereby enabling multiple solvers to be invoked in a standard manner as in the case of Open Solver Interface (OSI) in the COIN-OR repository (Saltzman 2002).

Concluding Remarks

Research in the general area of model management since 2000 has contributed to (1) the extension of modeling languages to represent a variety of models; (2) the development of distributed model management systems using web technologies to support models as services; (3) the automation of model composition process; and (4) the integration of modeling languages and systems with databases. Among the numerous surveys that have been published on the subject, the model management chapter in the book on information systems and decision processes (Stohr and Konsynski 1992), the special issue of *Decision Support Systems* edited by Blanning (1993), and the special issue of the *Annals of Operations Research* edited by Shetty (1992) deserve special mention for their broad coverage of issues and their quality of exposition. A survey of the model management literature may be found in Krishnan and Chari (2000). A survey of model management issues pertaining to data mining models can be found in Liu and Tuzhilin (2008).

See

- ▶ Algebraic Modeling Languages for Optimization
- ▶ Decision Support Systems (DSS)
- ▶ Structured Modeling
- ▶ Verification, Validation, and Testing of Models

References

Banerjee, S., & Basu, A. (1993). Model type selection in an integrated DSS environment. *Decision Support Systems, 9*, 75–89.

Basu, A., & Blanning, R. (1994a). Metagraphs: A tool for modeling decision support systems. *Management Science, 40*, 1579–1600.

Basu, A., & Blanning, R. (1994b). Model integration using metagraphs. *Information Systems Research, 5*, 195–218.

Basu, A., & Blanning, R. (1998). The analysis of assumptions in model bases using metagraphs. *Management Science, 44*, 982–995.

Basu, A., Blanning, R., & Shtub, A. (1997). Metagraphs in hierarchical modeling. *Management Science, 43*, 623–639.

Bhargava, H. K., & Kimbrough, S. O. (1993). Model management: An embedded languages approach. *Decision Support Systems, 10*, 277–300.

Bhargava, H. K., Kimbrough, S., & Krishnan, R. (1991). Unique names violations: A problem for model integration. *ORSA Journal on Computing, 3*, 107–120.

Bhargava, H. K., & Krishnan, R. (1993). Computer aided model construction. *Decision Support Systems, 9*, 91–111.

Bhargava, H. K., & Krishnan, R. (1998). The World Wide Web and its implications for OR/MS. *INFORMS Journal on Computing, 10*, 359–383.

Bhargava, H. K., Krishnan, R., & Piela, P. (1997). On formal semantics and analysis of typed modeling languages. *INFORMS Journal on Computing, 10*, 189–208.

Bhargava, H. K., Power, D. J., & Sun, D. (2007). Progress in web-based decision support technologies. *Decision Support Systems, 43*, 1083–1095.

Bhrammanee, T., & Wuwongse, V. (2008). ODDM: A framework for modelbases. *Decision Support Systems, 44*, 689–709.

Bischop, J., & Meeraus, A. (1982). On the development of a general algebraic modeling system in a strategic planning environment. *Mathematical Programming Study, 20*, 1–29.

Blanning, R. (1982). A relational framework for model management, *DSS-82 Transaction*, 16–28.

Blanning, R. (1993). Decision support systems: *Special issue on model management*. In R. Blanning., C. Holsapple., & A. Whinston (Eds.). Elsevier.

Bonczek, R., Holsapple, C., & Whinston, A. (1978). Mathematical programming within the context of a generalized data base management system. *R.A.I.R.O. Recherche Operationalle, 12*, 117–139.

Bradley, G., & Clemence, R. (1987). A type calculus for executable modeling languages. *IMA Journal on Mathematics in Management, 1*, 277–291.

Chari, K. (2002). Model composition using filter spaces. *Information Systems Research, 13*(1), 15–35.

Chari, K., & Sen, T. K. (1997). An integrated modeling system for structured modeling using model graphs. *INFORMS Journal on Computing, 9*(4), 397–416.

Choobineh, J. (1991). SQLMP: A data sublanguage for the representation and formulation of linear mathematical models. *ORSA Journal on Computing, 3*, 358–375.

Dhar, V., & Jarke, M. (1993). On modeling processes. *Decision Support Systems, 9*, 39–49.

Dolk, D. K., & Kotteman, J. E. (1993). Model integration and a theory of models. *Decision Support Systems, 9*, 51–63.

Fourer, R. (1983). Modeling languages versus matrix generators for linear programming. *ACM Transactions on Mathematical Software, 2*, 143–183.

Fourer, R., & Gay, D. (2002). Extending an algebraic modeling language to support constraint programming. *INFORMS Journal on Computing, 14*(4), 332–344.

Fourer, R., Gay, D., & Kernighan, B. W. (1990). A mathematical programming language. *Management Science, 36*, 519–554.

Gassmann, H. I., & Ireland, A. M. (1996). On the formulation of stochastic linear programs using algebraic modeling languages. *Annals of Operations Research, 64*, 83–112.

Geoffrion, A. M. (1987). An introduction to structured modeling. *Management Science, 33*, 547–588.

Geoffrion, A. M. (1989). Reusing structured models via model integration. In J. F. Nunamaker (Ed.), *Proceedings of Twenty-Second Annual Hawaii International Conference on the System Sciences*, III, (pp. 601–611). Los Alamitos, California: IEEE Press.

Geoffrion, A. M. (1992a). The SML language for structured modeling: Levels 1 and 2. *Operations Research, 40*, 38–57.

Geoffrion, A. M. (1992b). The SML language for structured modeling: Levels 3 and 4. *Operations Research, 40*, 58–75.

Geoffrion, A. M. (1999a). An informal annotated bibliography on structured modeling. *Interactive Transactions OR/MS, 1(2)*, online at http://catt.bus.okstate.edu/ITORMS/.

Geoffrion, A. M. (1999b). Structured modeling: Survey and future research directions. *Interactive Transactions OR/MS, 1(3)*, online at http://catt.bus.okstate.edu/ITORMS/.

Gray, P. (1987). *Guide to IFPS*. New York: McGraw-Hill.

Greenberg, H. J. (1987). ANALYZE: A computer-assisted analysis system for linear programming models. *Operations Research Letters, 6*, 249–255.

Greenberg, H. J. (1992). MODLER: Modeling by object-driven linear elemental relations. *Annals of Operations Research, 38*, 239–280.

Guenther, O., Muller, R., Schmidt, P., Bhargava, H. K., & Krishnan, R. (1997). MMM: A WWW-based method management system for using software modules remotely. *IEEE Internet Computing, 1*(5), 59–68.

Guieu, O., & Chinneck, J. W. (1999). Analyzing infeasible mixed-integer and integer linear programs. *INFORMS Journal on Computing, 11*, 63–77.

Guntzer, U., Muller, R., Muller, S., & Schimkat, R. (2007). Retrieval for decision support resources by structured models. *Decision Support Systems, 43*, 1117–1132.

Hoare, (1985). Communicating sequential processes. Prentice-Hall, Inc. Upper Saddle River, NJ, USA

Jones, C. V. (1990). An introduction to graph based modeling systems, part I: Overview. *ORSA Journal on Computing, 2*, 136–151.
Jones, C. V. (1991). An introduction to graph based modeling systems, part II: Graph grammars and the implementation. *ORSA Journal on Computing, 3*, 180–206.
Kimbrough, S., & Oliver, J. (1994). On automating candle lighting analysis: Insight from search with genetic algorithms and approximate models. In J. F. Nunamaker (Ed.), *Proceedings of the Twenty Seventh Hawaii International Conference on the System Sciences*, III (pp. 536–544). Los Alamitos, California: IEEE Press.
Krishnan, R. (1990). A logic modeling language for model construction. *Decision Support Systems, 6*, 123–152.
Krishnan, R., & Chari, K. (2000). Model management: Survey, future research directions and a bibliography. *Interactive Transactions of ORMS, 3*(1).
Krishnan, R., Li, X., & Steier, D. (1992). Development of a knowledge-based model formulation system. *Communications of the ACM, 35*, 138–146.
Krishnan, R., Piela, P., & Westerberg, A. (1993). Reusing mathematical models in ASCEND. In C. Holsapple & A. Whinston (Eds.), *Advances in decision support systems* (pp. 275–294). Munich, Germany: Springer-Verlag.
Liang, T. P., & Konsynski, B. R. (1993). Modeling by analogy: Use of analogical reasoning in model management systems. *Decision Support Systems, 9*, 113–125.
Liu, B., & Tuzhilin, A. (2008). Managing large collection of data mining models. *Communications of the ACM, 51*(2), 85–89.
Ma, P.-C., Murphy, F., & Stohr, E. (1989). A graphics interface for linear programming. *Communications of the ACM, 32*, 996–1012.
Madhusudan, T. (2007). Web services framework for distributed model management. *Information System Frontiers, 9*, 9–27.
Mannino, M. V., Greenberg, B. S., & Hong, S. N. (1990). Model libraries: Knowledge representation and reasoning. *ORSA Journal on Computing, 2*, 287–301.
Mills, H., Linger, R., & Hevner, A. (1986). *Principles of information systems analysis and design*. Orlando, FL: Academic Press.
Muhanna, W. (1992). On the organization of large shared of model bases. *Annals of Operations Research, 38*, 359–396.
Murphy, F., & Stohr, E. (1986). An intelligent system for formulating linear programs. *Decision Support Systems, 2*, 39–47.
Murphy, F., Stohr, E. A., & Asthana, A. (1992). Representation schemes for mathematical programming models. *Management Science, 38*, 964–991.
Piela, P., McKelvey, R., & Westerberg, A. (1992). An introduction to ASCEND: Its language and interactive environment. In J. F. Nunamaker Jr (Ed.), *Proceedings of the Twenty-Fifth Annual Hawaii International Conference on System Sciences*, Vol. III (pp. 449–461). Los Alamitos, California: IEEE Press.
Raghunathan, S., Krishnan, R., & May, J. (1994). MODFORM: A knowledge tool to support the modeling process. *Information Systems Research, 4*, 331–358.
Raghunathan, S., Krishnan, R., & May, J. (1995). On using belief maintenance systems to assist mathematical modeling. *IEEE Transactions on Systems, Man, and Cybernetics, 25*, 287–303.
Saltzman, M. J. (2002). COIN-OR: An open-source library for optimization. In S. S. Nielsen (Ed.), *Programming languages and systems in computational economics and finance*. Boston: Kluwer Academic Publishers.
Sharda, R., & Rampal, G. (1995). Algebraic Modeling Languages on PCs. *OR/MS Today, 22*(3), 58–63.
Sharda, R., & Steiger, D. (1996). Inductive model analysis systems: Enhancing model analysis in decision support systems. *Information Systems Research, 7*, 328–341.
Shetty, B. (1992). Annals of operations research: *Special issue on model management in operations research*. In B. Shetty (Ed.). Amsterdam: J.C. Baltzer Scientific Publishing.
Sprague, R. H., & Watson, H. J. (1975). Model management in MIS. *Proceedings of Seventeenth National AIDS Conference*, 213–215.
Stohr, E., & Konsynski, B. (1992). *Information systems and decision processes*. Los Altimos, CA: IEEE Press.
Tsai, Y.-C. (1998). Model integration using SML. *Decision Support Systems, 22*, 355–377.
Valente, P., & Mitra, G. (2007). The evolution of web-based optimization: From ASP to e-Services. *Decision Support Systems, 43*, 1096–1116.
Valente, P., Mitra, G., Sadki, M., & Fourer, R. (2009). Extending algebraic modelling languages for stochastic programming. *INFORMS Journal on Computing, 21*, 1, 107–122.6.
Will, H. J. (1975). Model management systems. In E. Grochia & N. Szyperski (Eds.), *Information systems and organization structure* (pp. 468–482). Berlin, Germany: Walter de Gruyter.

Model Testing

Investigating whether inaccuracies or errors exist in a model.

See

- ▶ Validation
- ▶ Verification
- ▶ Verification, Validation, and Testing of Models

Model User's Risk

Probability of accepting the credibility of a model when in fact the model is not sufficiently credible.

Model Validation

- ▶ Validation
- ▶ Verification
- ▶ Verification, Validation, and Testing of Models

Model Verification

- ▶ Validation
- ▶ Verification
- ▶ Verification, Validation, and Testing of Models

Model-based Search Methods

A class of global optimization methods that uses a probability distribution to generate candidate solutions, where in each iteration of the algorithm, the probability distribution is updated according to the performance of the population of candidate solutions. Examples include estimation of distribution algorithms, the cross-entropy method, and model reference adaptive search.

See

- ▶ Cross-Entropy Method

References

Larrañaga, P., & Lozano, J. A. (2002). *Estimation of distribution algorithms: A new tool for evolutionary computation*. Boston: Kluwer Academic.

MODI

Modified Distribution Method. A procedure for organizing the hand computations when solving a transportation problem using the transportation simplex method.

See

- ▶ Transportation Simplex (Primal-Dual) Method

MOIP

Multi-objective integer programming.

See

- ▶ Multiple Criteria Decision Making

MOLP

Multi-objective linear programming.

See

- ▶ Multiobjective Programming

Moment Generating Function

For a random variable X, the moment generating function is given by $M_X(t) = E[e^{tX}]$, assuming the expectation exists. For non-negative continuous random variables, it is basically identical to the Laplace transform for the corresponding probability density function.

Monte Carlo Methods

General term used to refer to the use of random numbers in a particular methodology, e.g., evaluating a high-dimensional deterministic integral or carrying out a randomized algorithm or simulation of a stochastic system, all based on statistical sampling techniques. The term "Monte Carlo" signifies the random or uncertain component that characterizes the

method and was coined in the 1940s by physicists working on the Manhattan nuclear weapons project, an allusion to gambling in Monte Carlo casinos. One of the strengths of the Monte Carlo method is that in many applications its computational burden grows only linearly in the dimension of problems where other methods suffer from an exponential (geometric) growth in computation.

See

- ▶ Las Vegas Algorithm
- ▶ Monte Carlo Simulation
- ▶ Randomized Algorithm
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Fishman, G. S. (1996). *Monte Carlo: Concepts, algorithms, and applications*. New York: Springer.
Metropolis, N., & Ulam, S. (1949). The Monte Carlo method. *Journal of the American Statistical Association, 44*(247), 335–341.

Monte Carlo Simulation

Simulation of systems modeled using random variables and/or stochastic processes. The underlying inputs are generally random numbers, sequences of independent and identically distributed random variables uniformly distributed on the unit interval. Sometimes called the Monte Carlo method, where the term "Monte Carlo" signifies the random or uncertain component that characterizes the method and was coined in the 1940s by physicists working on the Manhattan nuclear weapons project, an allusion to gambling in Monte Carlo casinos. Monte Carlo simulation is one of the most widely used tools in operations research and management science (OR/MS) and can be used to provide detailed models of complex systems arising in various OR/MS fields from manufacturing to transportation to computer/communications networks to financial engineering. One of the strengths of the Monte Carlo method is that in many applications its computational burden grows only linearly in the dimension of problems where other methods suffer from an exponential (geometric) growth in computation.

See

- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Simulation Optimization
- ▶ Variance Reduction Techniques in Monte Carlo Methods

References

Fishman, G. (2010). *Monte Carlo: Concepts, algorithms, and applications* (4th ed.). New York: Springer.
Metropolis, N., & Ulam, S. (1949). The Monte Carlo method. *Journal of the American Statistical Association, 44*(247), 335–341.
Rubinstein, R. Y., & Kroese, D. P. (2007). *Simulation and the Monte Carlo method* (2nd ed.). New York: Wiley-Interscience.

MOR

Military operations research; also used as an abbreviation for the journal *Mathematics of Operations Research*.

See

- ▶ Military Operations Research

Moral Hazard

A term in economics describing a situation in which a decision maker's actions are taken without bearing full risk, responsibility, or consequences for the potential outcomes. For example, having a valuable item with full insurance coverage against theft might make the owner more lax in safeguarding it.

Economist Paul Krugman described moral hazard as: "...any situation in which one person makes the decision about how much risk to take, while someone else bears the cost if things go badly."

References

Krugman, P. (2009). *The return of depression economics and the crisis of 2008*. New York: W.W. Norton.

MORS

Military Operations Research Society.

See

▶ Military Operations Research

MPS

▶ Mathematical-Programming System (MPS)

MRP

▶ Material Requirements Planning

MS

Management Science

MSE

Mean square error.

Multicommodity Network Flows

Bala Shetty
Texas A&M University, College Station, TX, USA

Introduction

The multicommodity minimal cost network flow problem may be described in terms of a distribution problem over a network $[V, E]$, where V is the node set with order n and E is the arc set with order m. The decision variable x^{jk} denotes the flow of commodity k through arc j, and the vector of all flows of commodity k is denoted by $\boldsymbol{x}^k = [x^{1k},\ldots, x^{mk}]$. The unit cost of flow of commodity k through arc j is denoted by c^{jk} and the corresponding vector of costs by $\boldsymbol{c}^k = [c^{1k},\ldots, c^{mk}]$. The total capacity of arc j is denoted by b^j with corresponding vector $\boldsymbol{b} = [b^1,\ldots, b^m]$. Mathematically, the multicommodity minimal cost network flow problem may be defined as follows:

$$\text{Minimize} \quad \sum_k \boldsymbol{c}^k \boldsymbol{x}^k$$

s.t.

$$\boldsymbol{A}\boldsymbol{x}^k = \boldsymbol{r}^k, \quad k = 1,\ldots, K$$

$$\sum_k \boldsymbol{x}^k \le \boldsymbol{b}$$

$$0 \le \boldsymbol{x}^k \le \boldsymbol{u}^k, \quad \text{for all } k,$$

where K denotes the number of commodities, $\boldsymbol{A}$ is a node-arc incidence matrix for $[V, E]$, $\boldsymbol{r}^k$ is the requirements vector for commodity k, and $\boldsymbol{u}^k$ is the vector of upper bounds for decision variable $\boldsymbol{x}^k$.

Multicommodity network flow problems are extensively studied because of their numerous applications and because of the intriguing network structure exhibited by these problems (Ahuja et al. 1993; Ali et al. 1984; Assad 1978; Castro and Nabona 1996; Kennington 1978; McBride 1998). Multicommodity models have been proposed for planning studies involving urban traffic systems (Chen and Meyer 1988; LeBlanc 1973; Potts and Oliver 1972) and communications systems (LeBlanc 1973; Naniwada 1969). Models for solving scheduling and routing problems have been proposed by Bellmore et al. (1971) and by Swoveland (1971). A multicommodity model for assigning students to achieve a desired ethnic composition was suggested by Clark and Surkis (1968). Multicommodity models have also been used for casualty evacuation of war time casualties, grain transportation, and aircraft routing for the USAF. A discussion of these applications can be found in Ali et al. (1984). Additional applications of multicommodity flows are given in Gautier and Granot (1995), and Popken (1994).

Solution Techniques

There are two basic approaches which have been employed to develop specialized techniques for multicommodity network flow problems: decomposition and partitioning. Decomposition approaches may be further characterized as price-directive or resource directive. A price-directive decomposition procedure directs the coordination between a master program and each of several subprograms by the changing the objective functions (prices) of the subprograms. The objective is to obtain a set of prices (dual variables) such that the combined solution for all subproblems yields an optimum for the original problem. A resource-directive decomposition procedure (Held et al. 1974; Kennington and Shalaby 1977), when applied to a multicommodity problem having K commodities, is to distribute the arc capacity among the individual commodities in such a way that solving K sub-programs yields an optimal flow for the coupled problem. At each iteration, an allocation is made and K single commodity flow problems are solved. The sum of capacities allocated to an arc over all commodities is equal to the arc capacity in the original problem. Hence, the combined flow from the solutions of the subproblems provides a feasible flow for the original problem. Optimality is tested and the procedure either terminates or a new arc capacity allocation is developed. Partitioning approaches are specializations of the simplex method where the current basis is partitioned to exploit its special structure. These techniques are specializations of primal, dual, or primal-dual simplex method. The papers of Hartman and Lasdon (1972), and Graves and McBride (1976) are primal techniques, while the work of Grigoriadis and White (1972) is a dual technique. An extensive discussion of these techniques can be found in Ahuja et al. (1993) and Kennington and Helgason (1980).

Several researchers have suggested algorithms for the multicommodity flow problem: Gersht and Shulman (1987), Barnhart (1993), Farvolden and Powell (1990), Farvolden et al. (1993), Liu (1997), and Schneur and Orlin (1998) all present alternative approaches for the multicommodity model. Parallel optimization has also been applied for the solution of multicommodity networks. Pinar and Zenios (1990) present a parallel decomposition algorithm for the multicommodity model using penalty functions. Shetty and Muthukrishnan (1990) develop a parallel projection which can be applied to resource-directive decomposition. Chen and Meyer (1988) decompose a nonlinear multicommodity problem arising in traffic assignment into single commodity network components that are independent by commodity. The difficulty of solving a multicommodity problem explodes when the decision variables are restricted to be integers. Very little work is available in the literature for the integer problem (Evans 1978; Evans and Jarvis 1978; Gendron and Crainic 1997).

Several computational studies involving multicommodity models have been reported in the literature. Ali et al. (1980) present a computational experience using the price-directive decomposition procedure (PPD), the resource directive-decomposition procedure (RDD), and the primal partitioning procedure (PP). They find the primal partitioning and price directive decomposition methods take approximately the same amount of computing time, while the resource directive decomposition runs in approximately one-half the time of the other two methods. Convergence to the optimal solution is guaranteed for PPD and PP, whereas RDD may experience convergence problems. Ali et al. (1984) present a comparison of the primal partitioning algorithm for solving the multicommodity model with a general purpose LP code. On a set of test problems, they find that the primal partitioning technique runs in approximately one-half the time required by the LP code. Farvolden et al. (1993) report very promising computational results for a class of multicommodity network problems using a primal partitioning code (PPLP). On these problems, they find PPLP to be two orders of magnitude faster than MINOS and about 50 times faster than OB1, a state-of-the-art LP solver.

Linear, nonlinear, and integer multicommodity models have numerous important applications in scheduling, routing, transportation, and communications. Real-world multicommodity models tend to be very large and there is a need for faster and more efficient algorithms for solving these models.

Thus, multicommodity models present unlimited opportunities for future research in large-scale optimization.

See

- ▶ Large-Scale Systems
- ▶ Linear Programming
- ▶ Logistics and Supply Chain Management
- ▶ Minimum-Cost Network-Flow Problem
- ▶ Network Optimization
- ▶ Transportation Problem

References

Ahuja, R. K., Magnanti, T. L., & Orlin, J. B. (1993). *Network flows: Theory, algorithms, and applications*. New Jersey: Prentice Hall.

Ali, A., Barnett, D., Farhangian, K., Kennington, J., McCarl, B., Patty, B., Shetty, B., & Wong, P. (1984). Multicommodity network flow problems: Applications and computations. *IIE Transactions, 16*, 127–134.

Ali, A., Helgason, R., Kennington, J., & Lall, H. (1980). Computational comparison among three multicommodity network flow algorithms. *Operations Research, 28*, 995–1000.

Assad, A. A. (1978). Multicommodity network flows–A survey. *Networks, 8*, 37–91.

Barnhart, C. (1993). Dual ascent methods for large-scale multicommodity flow problems. *Naval Research Logistics, 40*, 305–324.

Bellmore, M., Bennington, G., & Lubore, S. (1971). A multivehicle tanker scheduling problem. *Transportation Science, 5*, 36–47.

Castro, J., & Nabona, N. (1996). An implementation of linear and nonlinear multicommodity network flows. *European Journal of Operational Research, 92*, 37–53.

Chen, R., & Meyer, R. (1988). Parallel optimization for traffic assignment. *Mathematical Programming, 42*, 327–345.

Clark, S., & Surkis, J. (1968). An operations research approach to racial desegregation of school systems. *Socio-Economic Planning Sciences, 1*, 259–272.

Evans, J. (1978). The simplex method for integral multicommodity networks. *Naval Research Logistics, 25*, 31–38.

Evans, J., & Jarvis, J. (1978). Network topology and integral multicommodity flow problems. *Networks, 8*, 107–120.

Farvolden, J. M., & Powell, W. B. (1990). *A primal partitioning solution for multicommodity network flow problems* (Working Paper 90–04). Canada: Department of Industrial Engineering, University of Toronto.

Farvolden, J. M., Powell, W. B., & Lustig, I. J. (1993). A primal partitioning solution for the arc-chain formulation of a multicommodity network flow problem. *Operations Research, 41*, 669–693.

Gautier, A., & Granot, F. (1995). Forest management: A multicommodity flow formulation and sensitivity analysis. *Management Science, 41*, 1654–1668.

Gendron, B., & Crainic, T. G. (1997). A parallel branch-and-bound algorithm for multicommodity location with balancing requirements? *Computers and Operations Research, 24*, 829–847.

Gersht, A., & Shulman, A. (1987). A new algorithm for the solution of the minimum cost multicommodity flow problem. *Proceedings of the IEEE Conference on Decision and Control, 26*, 748–758.

Graves, G. W., & McBride, R. D. (1976). The factorization approach to large scale linear programming. *Mathematical Programming, 10*, 91–110.

Grigoriadis, M. D., & White, W. W. (1972). A partitioning algorithm for the multicommodity network flow problem. *Mathematical Programming, 3*, 157–177.

Hartman, J. K., & Lasdon, L. S. (1972). A generalized upper bounding algorithm for multicommodity network flow problems. *Networks, 1*, 331–354.

Held, M., Wolfe, P., & Crowder, H. (1974). Validation of subgradient optimization. *Mathematical Programming, 6*, 62–88.

Kennington, J. L. (1978). A survey of linear cost multicommodity network flows. *Operations Research, 26*, 209–236.

Kennington, J. L., & Helgason, R. (1980). *Algorithms for network programming*. New York: Wiley.

Kennington, J., & Shalaby, M. (1977). An effective subgradient procedure for minimal cost multicommodity flow problems. *Management Science, 23*, 994–1004.

LeBlanc, L. J. (1973). *Mathematical programming algorithms for large scale network equilibrium and network design problems*. Unpublished Dissertation, Industrial Engineering and Management Sciences Department, Northwestern University.

Liu, C.-M. (1997). Network dual steepest-edge methods for solving capacitated multicommodity network problems. *Computers and Industrial Engineering, 33*, 697–700.

McBride, R. (1998). Advances in solving the multi-commodity-flow problem. *Interfaces, 28*(2), 32–41.

Naniwada, M. (1969). Multicommodity flows in a communications network. *Electronics and Communications in Japan, 52-A*, 34–41.

Pinar, M. C., & Zenios, S. A. (1990). *Parallel decomposition of multicommodity network flows using smooth penalty functions*. Technical Report 90-12-06, Department of Decision Sciences, Wharton School, University of Pennsylvania, Philadelphia.

Popken, D. A. (1994). An algorithm for the multiattribute, multicommodity flow problem with freight consolidation and inventory costs. *Operations Research, 42*, 274–286.

Potts, R. B., & Oliver, R. M. (1972). *Flows in transportation networks*. New York: Academic.

Schneur, R., & Orlin, J. B. (1998). A scaling algorithm for multicommodity flow problems. *Operations Research, 46*, 231–246.
Shetty, B., & Muthukrishnan, R. (1990). A parallel projection for the multicommodity network model. *Journal of Operational Research, 41*, 837–842.
Swoveland, C. (1971). *Decomposition algorithms for the multi-commodity distribution problem* (Working Paper, No. 184). Los Angeles: Western Management Science Institute, University of California.

Multicommodity Network-Flow Problem

A minimum-cost network flow problem in which more than one commodity simultaneously flows from the supply nodes to the demand nodes. Unlike the single commodity problem, an optimal solution is not guaranteed to have integer flows. The problem takes on the block-angular matrix form that is suitable for solution by Dantzig-Wolfe decomposition. Applications areas include communications, traffic and logistics.

See

- ▶ Dantzig-Wolfe Decomposition Algorithm
- ▶ Minimum-Cost Network-Flow Problem
- ▶ Multicommodity Network Flows
- ▶ Network Optimization

Multidimensional Transportation Problem

Usually a transportation problem with a third index that refers to a product type available at the origins and demanded at the destinations. The variables x_{ijk} represent the amount of the kth product type shipped from the ith origin to the jth destination. The constraint set is a set of linear balance equations, with the usual linear cost objective function. It is also a special form of the multicommodity network-flow problem. Unlike the transportation problem, its optimal solution may not be integer-valued even if the network data are given as integers. The problem can also be defined with more than three indices.

See

- ▶ Multicommodity Network Flows
- ▶ Transportation Problem

Multiobjective Linear-Programming Problem

This problem has the usual set of linear-programming constraints ($\boldsymbol{Ax} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$) but requires the simultaneous optimization of more than one linear objective function, say p of them. It can be written as "Maximize" $\boldsymbol{Cx}$ subject to $\boldsymbol{Ax} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$, where $\boldsymbol{C}$ is a $p \times n$ matrix whose rows are the coefficients defined by the p objectives. Here "Maximize" represents the fact that it is usually impossible to find a solution to $\boldsymbol{Ax} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$, that simultaneously optimizes all the objectives. If there is such an (extreme) point, the problem is thus readily solved. Special multiobjective computational procedures are required to select a solution that is in effect a compromise solution between the extreme point solutions that optimize individual objective functions. The possible compromise solutions are taken from the set of efficient (nondominated) solutions. This problem is also called the vector optimization problem.

See

- ▶ Efficient Solution
- ▶ Multiobjective Programming
- ▶ Pareto-Optimal Solution

Multiobjective Programming

Ralph E. Steuer
University of Georgia, Athens, GA, USA

Introduction

Related to linear, integer, and nonlinear programming, multiobjective programming addresses the extensions to theory and practice of mathematical programming

problems with more than one objective function. Single objective programming must settle on a single objective such as to maximize profit or minimize cost. However, many if not most real-world problems are in an environment of multiple conflicting criteria. A sample of problems modeled with multiple objectives:

Oil Refinery Scheduling

min {cost}
min {imported crude}
min {high sulfur crude}
min {deviations from demand slate}

Production Planning

max {total net revenue}
max {minimum net revenue in any period}
min {backorders}
min {overtime}
min {finished goods inventory}

Forest Management

max {timber production}
max {visitor days of recreation}
max {wildlife habitat}
min {overdeviations from budget}

Emerging as a new topic in the 1970s, multiobjective programming has grown to the extent that numerous books have been written on the subject (e.g., Zeleny 1982; Yu 1985; Steuer 1986; Miettinen 1999; Ehrgott 2005) and applications of multiobjective programming can now be found in virtually all areas of operational research.

Terminology

A multiobjective programming problem is the following:

$$\begin{aligned} &\text{maximize} \quad \{f_1(\boldsymbol{x}) = z_1\} \\ &\vdots \\ &\text{maximize} \quad \{f_k(\boldsymbol{x}) = z_k\} \\ &\text{subject to} \quad \boldsymbol{x} \in S \end{aligned}$$

where k is the number of objectives, the z_i are criterion values, and S is the feasible region in decision space. Let $Z \subset R^k$ be the feasible region in criterion space where $\boldsymbol{z} \in Z$ if and only if there exists an $\boldsymbol{x} \in S$ such that $\boldsymbol{z} = (f_1(\boldsymbol{x}),\ldots,f_k(\boldsymbol{x}))$. Let $K = \{1,\ldots,k\}$. Criterion vector $\overline{z} \in Z$ is nondominated if and only if there does not exist another $\boldsymbol{z} \in Z$ such that $z_i \geq \overline{z}_i$ for all $i \in K$ and $z_i > \overline{z}_i$ for at least one $i \in K$. The set of all nondominated criterion vectors is designated N and is called the nondominated set. A point $\overline{\mathbf{x}} \in S$ is efficient if and only if its criterion vector $\overline{z} = (f_1(\overline{\boldsymbol{x}}),\ldots,f_k(\overline{\boldsymbol{x}})$ is nondominated. The set of all efficient points is designated E and is called the efficient set.

Let $U: R^k \rightarrow R$ be the utility function of the decision maker (DM). A $\boldsymbol{z}^\circ \in Z$ that maximizes U over Z is an optimal criterion vector and any $\boldsymbol{x}^\circ \in S$ such that $(f_1(\boldsymbol{x}^\circ),\ldots, f_k(\boldsymbol{x}^\circ)) = \boldsymbol{z}^\circ$ is an optimal solution of the multiobjective program. The interest in the efficient set E and the nondominated set N stems from the fact that if U is coordinatewise increasing (i.e., more is always better than less of each objective), $\boldsymbol{x}^\circ \in E$ and $\boldsymbol{z}^\circ \in N$. In this way, a multiobjective program can be solved by finding the most preferred criterion vector in N.

One might think that the best way to solve a multiobjective program would be to assess the DM's utility function and then solve

$$\begin{aligned} &\text{maximize} \quad \{U(z_1,\ldots,z_k)\} \\ &\text{subject to} \quad f_i(\boldsymbol{x}) = z_i, \quad i \in K,\ x \in S \end{aligned}$$

because any solution that solves this program is an optimal solution of the multiobjective program. However, multiobjective programs are usually not solved in this way because (1) of the difficulty in assessing an accurate enough U, (2) U would almost certainly be nonlinear, and (3) the DM would not likely see other candidate solutions during the solution process from which to gain an appreciation of the tradeoffs inherent in the problem.

Consequently, multiobjective programming employs mostly interactive procedures that only require implicit, as opposed to explicit, knowledge about the DM's utility function. In interactive procedures, the goal is to search the nondominated set for the DM's most preferred criterion vector. Unfortunately, because of the size of N, finding the best criterion vector in N is not a trivial task. As a result, interactive procedures are carefully crafted and can generally only be expected to conclude with what is called a final solution, a solution that is either optimal or close enough to being optimal to satisfactorily terminate the decision process.

Background Concepts

Along with the basics of conventional mathematical programming, multiobjective programming requires additional concepts not widely employed elsewhere in operations research. The key ones are as follows.

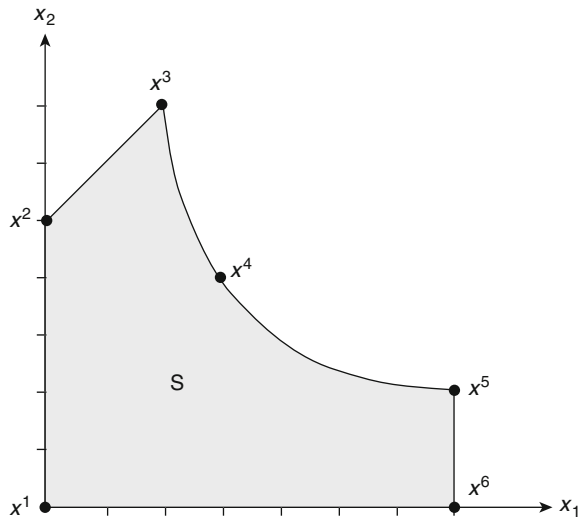
1. *Decision Space vs. Criterion Space*. Whereas single objective programming is typically studied in decision space, multiobjective programming is mostly studied in criterion space. To illustrate, consider

$$\begin{aligned} &\text{maximize} && \{x_1 - 1/2x_2 = z_1\} \\ &\text{maximize} && \{x_2 = z_2\} \\ &\text{subject to} && x \in S \end{aligned}$$

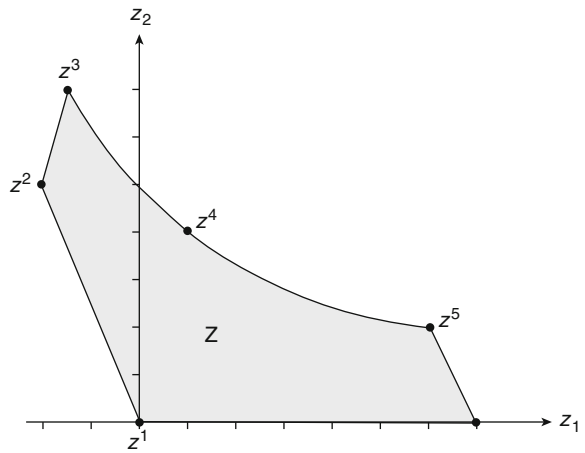
where S in decision space is in Fig. 1, and Z in criterion space is in Fig. 2. For instance z^4, which is the image of $\boldsymbol{x}^4 = (3, 4)$, is obtained by plugging the point (3, 4) into the objective functions to generate $z^4 = (1, 4)$. In Fig. 2, the nondominated set N is the set of boundary criterion vectors z^3 through z^4 to z^5 to z^6, inclusive. In Fig. 1, the efficient set E is the set of inverse images of the criterion vectors in N, namely the set of boundary points $\boldsymbol{x}^3$ through $\boldsymbol{x}^4$ to $\boldsymbol{x}^5$ to $\boldsymbol{x}^6$, inclusive. Note that Z is not necessarily confined to the nonnegative orthant.

2. *Unsupported Nondominated Criterion Vectors*. A $z \in N$ is unsupported if and only if it is possible to dominate it by a convex combination of other nondominated criterion vectors. In Fig. 2, the set of unsupported nondominated criterion vectors is the set of criterion vectors from z^3 through z^4 to z^5, exclusive of z^3 and z^5. The set of supported nondominated criterion vectors is the set that consists of z^3 plus the line segment z^5 to z^6, inclusive. Unsupported nondominated criterion vectors can only occur in problems that possess non-convex feasible regions; hence, they can easily be present in integer and nonlinear multiobjective programs.

3. *Identifying Nondominated Criterion Vectors*. To graphically determine whether a $\overline{z} \in Z$ is non-dominated or not, visualize the nonnegative orthant in R^k translated so that its origin is at $\overline{z}$. Note that, apart from $\overline{z}$, a vector dominates $\overline{z}$ if and only if the vector is in the translated nonnegative orthant. In other words, $\overline{z}$ is nondominated if and only if the translated nonnegative orthant is empty of feasible criterion vectors other than for $\overline{z}$. Visualizing in Fig. 2 the nonnegative orthant translated to z^4, it can be seen that z^4 is nondominated. Visualizing the nonnegative orthant translated to z^2, it can be seen that z^2 is dominated.



Multiobjective Programming, Fig. 1 Representation in decision space



Multiobjective Programming, Fig. 2 Representation in criterion space

4. *Payoff Tables*. Assuming that each objective is bounded over the feasible region, a payoff table is of the form

	z_1	z_2		z_k
z^1	z_1^*	z_{12}		z_{1k}
z^2	z_{21}	z_2^*		z_{2k}
z^k	z_{k1}	z_{k2}		z_k^*

where the rows are criterion vectors resulting from individually maximizing the objectives. For instance, z_{12} is the value of the second objective function at the point that maximizes the first objective. The z_i^* entries along the main diagonal of the payoff table are the maximum criterion values of the different objectives over the nondominated set. The minimum value in the ith column of the payoff table is often used as an estimate of the minimum criterion value of the ith objective over N because the true minimum criterion values over N (called nadir values) are typically difficult to obtain (Isermann and Steuer 1988; Alves and Costa 2009)

5. z^{**} *Reference Criterion Vectors*. A $z^{**} \in R^k$ *reference* criterion vector is a criterion vector that is suspended above the nondominated set. Its components are given by

$$z_i^{**} = z_i^* + \in_i$$

where the $\in_i$ are small computationally significant positive values.

6. *Weighting Vector Space*. Without loss of generality, let

$$\Lambda = \left\{ \lambda \in R^k | \lambda_i \in (0,1), \sum_{i \in k} \lambda_i = 1 \right\}$$

be weighting vector space. In an interactive environment, subsets of Λ called interval defined subsets are of the form

$$\Lambda^{(h)} = \left\{ \lambda \in R^k | \lambda_i \in \left(\ell_i^{(h)}, \mu_i^{(h)} \right), \sum_{i \in k} \lambda_i = 1 \right\}$$

where h is the iteration number and

$$0 \le l_i^{(h)} \le \mu_i^{(h)} \le 1 \quad i \in K$$
$$\mu_i^{(h)} - l_i^{(h)} = \mu_j^{(h)} - l_j^{(h)} \quad \text{for all } i \ne j$$

Sequences of successively smaller interval subsets can be defined by reducing the $\mu_i^{(h)} - l_i^{(h)}$ interval widths at each iteration.

7. *Sampling Programs*. The weighted-sums program

$$\max \left\{ \sum_{i \in K} \lambda_i f_i(x) | x \in S \right\}$$

can be used to sample the nondominated set because, as long as $\lambda \in \Lambda$, the program returns an efficient point. A disadvantage of the weighted-sums program is that it cannot generate unsupported points.

To make downward probes of the nondominated set from a z^{**} as required in many of the interactive procedures of multiobjective programming, the augmented Tchebycheff program is employed

$$\text{minimize} \left\{ \alpha - \rho \sum_{i \in K} z_i \right\}$$
$$\text{subject to}$$
$$\alpha \ge \lambda_i \left(z_i^{**} - z_i \right) \quad i \in K$$
$$f_i(\boldsymbol{x}) = \boldsymbol{z_i} \quad \boldsymbol{i \in K}$$
$$\boldsymbol{x} \in S$$
$$z \in R^k \text{ unrestricted}$$

where $\alpha \in R$, $\lambda \in \Lambda$, and ρ is a small computationally significant positive number. A disadvantage of the augmented Tchebycheff program is that, regardless of the value of ρ, there may still remain unsupported members of the nondominated set that the program is unable to compute (Steuer 1986).

A program that has better mathematical properties, although somewhat more difficult to implement, is the lexicographic Tchebycheff program

$$\text{lex}\ \min\left\{\alpha, -\sum_{i\in K} z_i\right\}$$
$$\text{subject to}$$
$$\alpha \geq \lambda_i\left(z_i^{**} - z_i\right) \quad i \in K$$
$$f_i(\boldsymbol{x}) = z_i \quad i \in K$$
$$\boldsymbol{x} \in S$$
$$z \in R^k \text{ unrestricted}$$

where $\lambda \in \Lambda$. At the first lexicographic level it is solved to minimize α. At the second lexicographic level, subject to only those solutions that minimize α, $-\sum_{i\in K} z_i$ is minimized. Not only does the lexicographic Tchebycheff program always return a nondominated criterion vector, but if $\boldsymbol{z}$ is nondominated, there then exists a $\overline{\lambda} \in \Lambda$ such that $\boldsymbol{z}$ uniquely solves the program (Steuer 1986).

8. *Aspiration Criterion Vectors*. An aspiration criterion vector $\boldsymbol{q} \in R^k$ is a criterion vector specified by a DM to reflect his or her hopes or expectations from a problem. An aspiration criterion vector, when specified, is typically projected onto N by an augmented or lexicographic Tchebycheff program in order to find the nondominated criterion vector closest to the aspiration criterion vector.
9. *T-vertex λ-vector Defined by* $\boldsymbol{q}$ *and* $\boldsymbol{z}^{**}$. The T-vertex (Tchebycheff-vertex) λ-vector defined by $\boldsymbol{q}$ and z^{**} is the $\lambda \in \Lambda$ whose components are given by

$$\lambda_i = \frac{1}{\left(z_i^{**} - q_i\right)}\left[\sum_{i\in K}\frac{1}{\left(z_j^{**} - q_j\right)}\right]^{-1}$$

The T-vertex λ-vector, when installed in an augmented or lexicographic Tchebycheff program, causes the program to probe the nondominated set along a line that goes through both z^{**} and $\boldsymbol{q}$ in the direction

$$-\left(\frac{1}{\lambda_1}, \ldots, \frac{1}{\lambda_k}\right)$$

Vector-Maximum Algorithms

In the linear case, a multiple objective linear program (MOLP) is sometimes written in vector-maximum form

Multiobjective Programming, Table 1 Average numbers of MOLP efficient extreme points

MOLP size $k \times m \times n$	Efficient extreme points	Approximate times in seconds
$3 \times 50 \times 75$	1,798	2
$3 \times 100 \times 150$	11,897	40
$3 \times 200 \times 300$	128,237	1,600
$4 \times 50 \times 75$	9,921	30
$4 \times 100 \times 150$	682,920	3,500
$5 \times 50 \times 75$	141,444	300

$$\text{“ max ”}, \{\boldsymbol{Cx} = \boldsymbol{z} | \boldsymbol{x} \in \boldsymbol{S}\}$$

where $\boldsymbol{C}$ is the $k \times n$ matrix whose rows are the coefficient vectors of the k objectives. A point is a solution to a vector-maximum problem if and only if it is efficient. Algorithms for characterizing the efficient set E of an MOLP are called vector-maximum algorithms. In the 1970s, considerable effort was spent on the development of vector-maximum codes to compute all efficient extreme points. The thought was that, by reviewing the list of nondominated criterion vectors associated with the efficient extreme points, a DM would be able to identify his or her efficient extreme point of greatest utility in hopes of satisfactorily terminating the decision process.

Unfortunately, MOLPs have many efficient extreme points as indicted in Table 1 (sample size of ten for each problems size). Whereas the number of variables and the number of constraints play a role, the factor most dramatically affecting the number of efficient extreme points is the dimensionality of the criterion cone, the convex cone generated by the gradients of the k objective functions.

With nondominated sets of sizes indicated in Table 1, other approaches have been attempted such as by Klamroth, Tind and Wiecek 2002, but mostly, the figures have led to interactive procedures moving to the forefront of multiobjective programming.

Interactive Procedures

In interactive multiobjective programming, an exploration over the feasible region for the best point in the non-dominated set is conducted. Interactive

procedures are characterized by phases of decision making alternating with phases of computation. A pattern is generally established and kept repeating it until termination. At each iteration, a solution, or a group of solutions, is generated for examination. As a result of the examination, the DM inputs updated preference information to the solution procedure in the form of values of the controlling parameters (preference weights, aspiration criterion vectors, λ-vector interval widths, criterion vector components to be increased/decreased/held fixed, criterion vector lower bounds, etc., depending upon the particular interactive procedure).

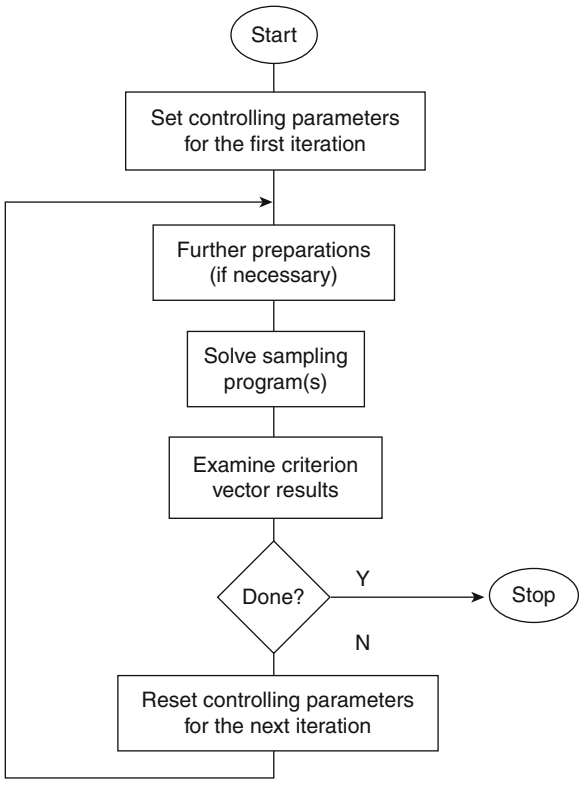
While many interactive procedures have been proposed, virtually all of them more or less follow the same general algorithmic outline. As portrayed in Fig. 3, the general algorithmic outline includes:

- an initial setting of the controlling parameters;
- optimization of one or more mathematical programming problems to probe (i.e., sample) the nondominated set;
- examination of the criterion vector results; and
- a resetting of the controlling parameters for the next iteration in the light of what was learned on the current iteration

With the consensus being that a range of interactive procedures is necessary because the most appropriate one to use is often application or user decision-making style dependent, ten of the most prominent interactive procedures, along with the dates of their original articles, are as follows:

1. ECON: e-Constraint Method, Traditional method
2. STEM: (Benayoun et al. 1971)
3. GDF: Geoffrion-Dyer-Feinberg Procedure (1972)
4. ZW: Zionts-Wallenius Procedure (1976)
5. IGP: Interactive Goal Programming (Spronk 1981)
6. WIERZ: Wierzbicki's Aspiration Criterion Vector Method (1982, 1986)
7. TCH: Tchebycheff Method (Steuer and Choo 1983)
8. RACE: Pareto Race (Korhonen and Laakso 1986; Korhonen and Wallenius 1988)
9. NIMBUS: (Miettinen 1999)
10. MICA: Modified Interactive Chebychev Algorithm (Luque et al. 2010)

Other interactive multiobjective programming procedures include those by Nakayama and Sawaragi (1984), Climaco and Antunes (1987), and Koksalan and Karasakal (2006).



Multiobjective Programming, Fig. 3 General algorithmic outline

Selected Interactive Procedures

The Aspiration Criterion Vector Method (WIERZ) begins by asking the DM to specify an aspiration criterion vector $\boldsymbol{q}^{(1)} < z^{**}$. Using the T-vertex λ-vector defined by $\boldsymbol{q}^{(1)}$ and z^{**}, the augmented Tchebycheff program is solved, thus projecting $\boldsymbol{q}^{(1)}$ onto N in order to produce $z^{(1)}$. In the light of $z^{(1)}$, the DM specifies a new aspiration criterion vector $\boldsymbol{q}^{(2)}$. Using the T-vertex λ-vector defined by $\boldsymbol{q}^{(2)}$ and z^{**}, the augmented Tchebycheff program is solved, thus projecting $\boldsymbol{q}^{(2)}$ onto N in order to produce $z^{(2)}$. In the light of $z^{(2)}$, the DM specifies a third aspiration criterion vector $\boldsymbol{q}^{(3)}$, and so forth. Algorithmically, the steps are as follows:

Step 1. $h = 0$. Construct a payoff table, form a z^{**} reference criterion vector, and specify $\rho > 0$ for use in the augmented Tchebycheff program. The DM specifies aspiration criterion vector $\boldsymbol{q}^{(1)}$.

Step 2. $h = h + 1$. Compute T-vertex λ-vector defined by $\boldsymbol{q}^{(h)}$ and z^{**}.

Step 3. Using the T-vertex λ-vector, solve the augmented Tchebycheff program for $z^{(h)}$.

Step 4. In the light of what the DM has been able to learn about the problem so far, the DM contemplates $z^{(h)}$.

Step 5. If the DM wishes to cease iterating, stop with $(z^{(h)}, x^{(h)})$ as the final solution. Otherwise, continue on to Step 6.

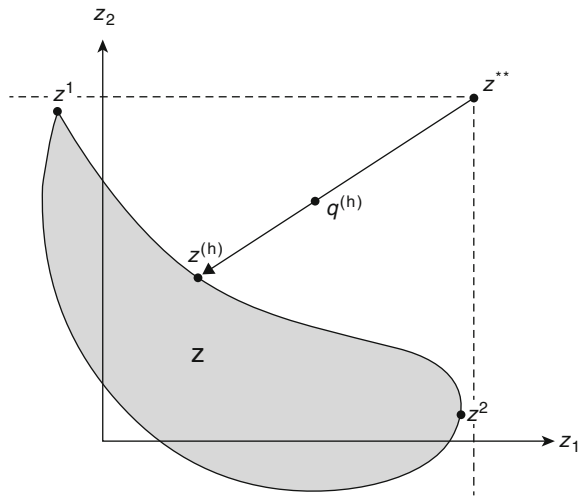
Step 6. The DM specifies another aspiration criterion vector, designated $q^{(h-1)}$. Go to Step 2.

Consider Fig. 4 in which N is the set of boundary criterion vectors z^1 through $z^{(h)}$ to z^2, inclusive. In the figure, it can be seen the way aspiration criterion vector $q^{(h)}$ is projected onto the nondominated set by means of the augmented Tchebycheff program. Note that the direction of the arrow emanating from z^{**} and going through $q^{(h)}$ is given by

$$-\left(\frac{1}{\lambda_1}, \ldots, \frac{1}{\lambda_k}\right)$$

where the λ_i are the components of the T-vertex λ-vector defined by $q^{(h)}$ and z^{**}. Thus changing $q^{(h)}$ changes the $z^{(h)}$ generated by the sampling program.

Instead of generating only one solution at each iteration, the Tchebycheff Method (TCH) generates groups of solutions by making multiple probes of each subset in a sequence of progressively smaller subsets of N. Letting P be the number of solutions to be presented to the DM at each iteration, TCH begins by generating P well-spaced λ-vectors from $\Lambda^{(1)} = \Lambda$. Then the lexicographic Tchebycheff program is solved for each of the λ-vectors. From the P resulting nondominated criterion vectors, the DM selects his or her most preferred, designating it $z^{(1)}$. At this point, the interval widths of $\Lambda^{(1)}$ are reduced and centered about the T-vertex λ-vector defined by $z^{(1)}$ and z^{**} to form an interval defined subset $\Lambda^{(2)}$. Then P well-spaced λ-vectors are generated from $\Lambda^{(2)}$ and the lexicographic Tchebycheff program is solved for each of the λ-vectors. From the P resulting non-dominated criterion vectors, the DM selects the most preferred, designating it $z^{(2)}$. Now the interval widths of $\Lambda^{(2)}$ are reduced and centered about the T-vertex λ-vector defined by $z^{(2)}$ and z^{**} to form an interval defined subset $\Lambda^{(3)}$. Then P well-spaced λ-vectors are generated from $\Lambda^{(3)}$ and the lexicographic Tchebycheff program is solved for each of them, and so forth.



Multiobjective Programming, Fig. 4 Projection of $q^{(h)}$ onto the nondominated set

Another procedure that also generates multiple solutions at each iteration, but employs the weighted-sums program, is the Geoffrion-Dyer-Feinberg (GDF) procedure. GDF begins with the specification of an initial feasible criterion vector $z^{(0)}$. Then the DM specifies a λ-vector that is to be reflective of the local marginal tradeoffs at $z^{(0)}$. Using this λ-vector, the weighted-sums program is solved for criterion vector $y^{(1)}$. Then the line through the feasible region in criterion space Z that starts at $z^{(0)}$ and ends at $y^{(1)}$ is divided into segments so as to create P equally spaced criterion vectors. The most preferred of the equally spaced criterion vectors becomes $z^{(1)}$. Then the DM specifies a new λ-vector that is to be reflective of the local marginal tradeoffs at $z^{(1)}$. Using this λ-vector, the weighted-sums program is solved for criterion vector $y^{(2)}$. Then the line segment through Z that starts at $z^{(1)}$ and ends at $y^{(2)}$ is divided into segments so as to create P new equally spaced criterion vectors. The most preferred of the new equally spaced criterion vectors becomes $z^{(2)}$, and so forth.

Features from different procedures can easily be combined. For instance, drawing from STEM, WIERZ and NIMBUS, one could have the following. After forming a z^{**} reference criterion vector, an initial aspiration criterion vector $q^{(1)}$ specified. Then one of the Tchebycheff programs is solved using the T-vertex λ-vector defined by $q^{(1)}$ and z^{**} to produce $z^{(1)}$. The DM then specifies the components of $z^{(1)}$ that are to be

increased, the amounts of each increase, the components that are to be relaxed, and the amounts of each relaxation in order to form a second aspiration criterion vector $\boldsymbol{q}^{(2)}$. Using the T-vertex λ-vector defined by $\boldsymbol{q}^{(2)}$ and z^{**}, one of the Tchebycheff programs is solved to produce $z^{(2)}$. The DM then specifies which components of $z^{(2)}$ are to be increased, the amounts of each increase, the components that are to be relaxed, and the amounts of each relaxation in order to form $\boldsymbol{q}^{(3)}$. Using the T-vertex λ-vector defined by $\boldsymbol{q}^{(3)}$ and z^{**}, one of the Tchebycheff programs is solved to produce $z^{(3)}$, and so forth.

Concluding Remarks

Because the weighted-sums, augmented Tchebycheff, and other variants of these programs that are used to sample the nondominated set are single criterion optimization problems, conventional mathematical programming software can in most cases be employed (Gardiner and Steuer 1994). In this way, interactive procedures can address multiobjective programming problems with as many constraints and variables as in single objective programming. Unfortunately, in multiobjective programming, there are limitations with regard to the number of objectives. Problems with up to about five objectives can generally be accommodated, but above this number, difficulties can arise because of the rate at which the nondominated set grows as the number of objectives increases.

See

- ▶ Decision Analysis
- ▶ Goal Programming
- ▶ Linear Programming
- ▶ Multiple Criteria Decision Making
- ▶ Utility Theory

References

Alves, M. J., & Costa, J. P. (2009). An exact method for computing the nadir values in multiple objective linear programming. *European Journal of Operational Journal, 198*(2), 637–646.

Benayoun, R., de Montgolfier, J., Tergny, J., & Larichev, O. (1971). Linear programming with multiple objective functions: Step method (STEM). *Mathematical Programming, 1*(3), 366–375.

Benson, H. P., & Sun, E. (2000). Outcome space partition of the weight set in multiobjective linear programming. *Journal of Optimization Theory and Applications, 105*(1), 17–36.

Climaco, J. C. N., & Antunes, C. H. (1987). TRIMAP–An interactive tricriteria linear programming package. *Foundations Control Engineering, 12*(3), 101–120.

Ehrgott, M. (2005). *Multicriteria optimization*. Berlin: Springer.

Gardiner, L. R., & Steuer, R. E. (1994). Unified interactive multiple objective programming. *European Journal of Operational Journal, 74*(3), 391–406.

Geoffrion, A. M., Dyer, J. S., & Feinberg, A. (1972). An interactive approach for multicriterion optimization, with an application to the operation of an academic department. *Management Science, 19*(4), 357–368.

Isermann, H., & Steuer, R. E. (1988). Computational experience concerning payoff tables and minimum criterion values over the efficient set. *European Journal of Operational Journal, 33*(1), 91–97.

Klamroth, K., Tind, J., & Wiecek, M. (2002). Unbiased approximation in multicriteria optimization. *Mathematical Methods of Operations Research, 36*(3), 413–437.

Koksalan, M., & Karasakal, E. (2006). An interactive approach for multiobjective programming. *Journal of the Operational Research Society, 57*(5), 532–540.

Korhonen, P. J., & Laakso, J. (1986). A visual interactive method for solving the multiple criteria problem. *European Journal of Operational Journal, 24*(3), 277–287.

Korhonen, P. J., & Wallenius, J. (1988). A pareto race. *Naval Research Logistics, 35*(6), 615–623.

Luque, M., Ruiz, F., & Steuer, R. E. (2010). Modified Interactive Chebychev Algorithm (MICA) for convex multiobjective programming. *European Journal of Operational Journal, 204*(3), 557–564.

Miettinen, K. M. (1999). *Nonlinear multiobjective optimization*. Norwell: Kluwer.

Nakayama, H., & Sawaragi, Y. (1984). Satisficing trade off method for multiobjective programming. *Lecture Notes in Economics and Mathematical Systems, 229*, 113–122.

Spronk, J. (1981). *Interactive multiple goal programming*. Boston: Martinus Nijhoff Publishing.

Steuer, R. E. (1986). *Multiple criteria optimization: Theory, computation, and application*. New York: John Wiley.

Steuer, R. E., & Choo, E.-U. (1983). An interactive weighted Tchebycheff procedure for multiple objective programming. *Mathematical Programming, 26*(1), 326–344.

Wierzbicki, A. P. (1982). A mathematical basis for satisficing decision making. *Mathematical Modelling, 3*, 391–405.

Wierzbicki, A. P. (1986). On the completeness and constructiveness of parametric characterizations to vector optimization problems. *OR-Spektrum, 8*, 73–87.

Yu, P. L. (1985). *Multiple-criteria decision making: Concepts techniques and extensions*. New York: Plenum Press.

Zeleny, M. (1982). *Multiple criteria decision making*. New York: McGraw-Hill.

Zionts, S., & Wallenius, J. (1976). An interactive programming method for solving the multiple criteria problem. *Management Science, 22*(6), 652–663.

Multi-armed Bandit Problem

Sequential decision-making problem under uncertainty involving a set of machines (arms) each offering random unknown rewards, in which the decision maker must decide each period which machine (arm) to play (pull), with the objective of maximizing the total reward received. The problem is analogous to playing slot machines in a gambling casino, but has many practical OR/MS applications involving dynamic stochastic resource allocation. One of the basic trade-offs in these types of problems is between exploitation (e.g., playing the machine that has given the best mean reward thus far) versus exploration (playing a machine that has not been tried or one that has been tried infrequently with highly variable rewards).

Multi-attribute Utility Theory

Rakesh K. Sarin
University of California, Los Angeles, CA, USA

Consider a decision problem such as selection of a job, choice of an automobile, or resource allocation in a public program (education, health, criminal justice, etc.). These problems share a common feature—decision alternatives impact multiple attributes. The attractiveness of an alternative therefore depends on how well it scores on each attribute of interest and the relative importance of these attributes. Multi-attribute utility theory (MAUT) is useful in quantifying relative attractiveness of multi-attribute alternatives.

The following notation will be used:

X_i the set of outcomes (scores, consequences) on the ith attribute
x_i a specific outcome in X_i
X $X_1 \times X_2 \times \ldots \times X_n$ (Cartesian product)
u_i a single attribute utility function $u_i : X_i \rightarrow \mathbb{R}$
u the overall utility function, u: $X \rightarrow \mathbb{R}$
$\gtrsim$ "is preferred to"

A decision maker uses the overall utility function, u, to choose among available alternatives. The major emphasis of the work on multi-attribute utility theory has been on questions involving u: on conditions for its decomposition into simple polynomials, on methods for its assessment, and on methods for obtaining sufficient information regarding u so that the evaluation can proceed without its explicit identification with full precision.

The primitive in the theory is the preference relation $\gtrsim$ defined over X. Luce et al. (1965) and Fishburn (1964) provide conditions on a decision maker's preferences that guarantee the existence of a utility function u such that

$$\begin{aligned} &(x_1, \ldots, x_n) \gtrsim (y_1, \ldots, y_n), \\ &\qquad x_i, y_i \in X_i, i = 1, \ldots n \\ &\text{if and only if} \\ &u(x_1, \ldots, x_n) \geq u(y_1, \ldots y_n) \end{aligned} \tag{1}$$

Additional conditions are needed to decompose the multi-attribute utility function u into simple parts. The most common approach for evaluating multi-attribute alternatives is to use an additive representation. For simplicity, assume that there exist the most preferred outcome x_i^* and the least preferred outcome x_i^0 on each attribute $i = 1$ to n. In the additive representation, a real value u is assigned to each outcome $(x_1, \ldots, x_n)$ by

$$u(x_i, \ldots, x_n) = \sum_{i=i}^{n} w_i u_i(x_i) \tag{2}$$

where the $\{u_i\}$ are single attribute utility functions over X_i that are scaled from 0 to 1, i.e., $u_i(x_i^*) = 1, u_i(x_i^0) = 0$ for $i = 1$ to n, and the $\{w_i\}$ are positive scaling constants reflecting relative importance of the attributes with $\sum_{j=1}^{n} w_j = 1$.

If the interest is in simply rank-ordering the available alternatives, then the key condition for the additive form in (2) is mutual preferential independence. The resulting utility function is called an ordinal value function. Attributes X_i and X_j are preferentially independent if the tradeoffs (substitution rates) between X_i and X_j are independent of all other attributes. Mutual preferential independence requires that preference independence holds for all pairs X_i and X_j. Essentially, mutual preferential independence implies that the indifference curves for any pair of attributes are unaffected by the fixed levels of the remaining

attributes. Debreu (1960), Luce and Tukey (1964), and Gorman (1968) provide axiom systems and analysis for the additive form (2).

If, in addition to rank order, one is also interested in the strength of preference between pairs of alternatives, then additional conditions are needed. The resulting utility function is called a measurable value function, and it may be used to order the preference differences between the alternatives.

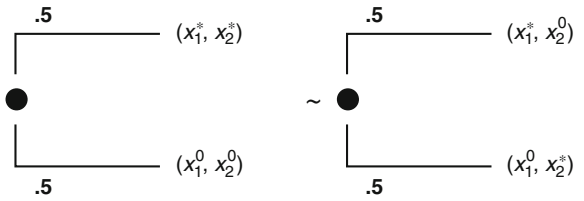
The key condition for an additive measurable value function is difference independence (see Dyer and Sarin 1979). This condition asserts that the preference difference between two alternatives that differ only in terms of one attribute does not depend on the common outcomes on the other $n-1$ attributes.

Finally, perhaps the most researched topic is the case of decisions under risk where the outcome of an alternative is characterized by a probability distribution over X. Denote $\tilde{X}$ as the set of all simple probability distributions over X. Assume that for any $p \in \tilde{X}$ there exists an alternative that can be identified with p, and thus p could be termed as a risky alternative. The outcome of an alternative $p \in \tilde{X}$ might be represented by the lottery which assigns probabilities $p_1, \ldots, p_l, \sum_{j=1}^{l} p_j = 1$, to the outcomes $x^1, \ldots, x^l \in X$, respectively. For the choice among risky alternatives $p, q \in \tilde{X}$, von Neumann and Morgenstern (1947) specified conditions on the decision maker's preference relation $\gtrsim$ over $\tilde{X}$ that imply:

$$\begin{aligned} &p \gtrsim q \\ &\text{if and only if} \\ &\sum_{x \in X} p(x)u(x) \geq \sum_{x \in X} q(x)u(x). \end{aligned} \tag{3}$$

Notice that the same symbol u has been used to denote ordinal value function, measurable value function, and now the von Neumann-Morgenstern utility function. The context, however, makes the interpretation clear.

A majority of the applied work in multi-attribute utility theory deals with the case when the von Neumann-Morgenstern utility function is decomposed into the additive form (2). Fishburn (1965a, b) derived necessary and sufficient conditions for a utility function to be additive. The key condition for additivity is the marginality condition, which states that the preferences for any lottery $p \in X$ should depend only on the marginal probability distributions over X_i and not on their joint probability distribution. Thus, for additivity to hold, the two lotteries below must be indifferent:



Notice that in either lottery, the marginal probability of receiving the most preferred outcome or the least preferred outcome on each attribute is identical. A decision maker may, however, prefer the right-hand side lottery over the left-hand side lottery if the decision maker wishes to avoid a 0.5 chance of the poor outcome (x_1^0, x_2^0) on both attributes.

The assessment of single attribute utility functions $\{u_i\}$ in (2) will require different methods depending on whether the overall utility represents an ordinal value function, a measurable value function, or a von Neumann-Morgenstern utility function. Keeney and Raiffa (1976) discuss methods for assessing multi-attribute ordinal value function and multi-attribute von Neumann-Morgenstern utility function. Dyer and Sarin (1979) and von Winterfeldt and Edwards (1986) discuss assessment of multi-attribute measurable value function.

Besides the additive form (2), a multiplicative form for the overall utility function has also found applications in a wide variety of contexts. In the multiplicative representation, a real value u is assigned to each outcome $(x_1, \ldots, x_n)$ by

$$1 + ku(x_1, \ldots, x_n) = \left[\prod_{i=1}^{n} [1 + kk_i u_i(x_i)]\right]$$

where the $\{u_i\}$ are single attribute utility functions over X_i that are scaled from zero to one, the $\{k_i\}$ are positive scaling constants, k is an additional scaling constant satisfying $k > -1$, and

$$1 + k = \prod_{i=1}^{n} [(1 + kk_i)].$$

If u is a measurable value function, then weak difference independence along with mutual

preference independence provides the desired result. An attribute is weak difference independent of the other attributes if preference difference between pairs of levels of that attribute do not depend on fixed levels of any of the other attributes. Thus, for $x_i, y_i, w_i, z_i \in X_i$, the ordering of preference difference between x_i and y_i, and w_i and z_i, remains unchanged whether one fixes the other attributes at their most preferred levels or at their least preferred levels.

If the overall utility function is used for ranking lotteries as in (3), then a utility independence condition, first introduced by Keeney (1969), is needed to provide the multiplicative representation (4). An attribute is said to be utility independent of the other attributes if the decision maker's preferences for lotteries over this attribute do not depend on the fixed levels of the remaining attributes. Mutual preferential independence and one attribute being utility independent of the others are sufficient to guarantee either the multiplicative form (4) or the additive form (2). The additive form results if in (4) $k = 0$ or $\sum_{j=1}^{n} k_j = 1$. Keeney and Raiffa (1976) discuss methods for calibrating the additive and multiplicative forms for the utility function. In the literature, other independence conditions have been identified that lead to more complex nonadditive decompositions of the utility function. These general conditions are reviewed in Farquhar (1977).

If utilities, importance weights, and probabilities are incompletely specified, then the approaches of Fishburn (1964) and Sarin (1975) can be used to obtain a partial ranking of alternatives.

The key feature of multi-attribute utility theory is to specify verifiable conditions on a decision maker's preferences. If these conditions are satisfied, then the multi-attribute utility function can be decomposed into simple parts. This approach of breaking the complex value problem (objective function) into manageable parts has found significant applications in decision and policy analysis. In broad terms, multi-attribute utility theory facilitates measurement of preferences or values. The axioms of the theory have been found to be useful in suggesting approaches for measurement of values. In physical measurements (e.g., length), the methods for measurement have been known for a long time and the theory of measurement has added little to suggesting new methods. In the measurement of values, however, several new methods have been developed as a direct result of the theory.

See

- ▶ Analytic Hierarchy Process
- ▶ Decision Analysis
- ▶ Multiple Criteria Decision Making
- ▶ Preference Theory
- ▶ Utility Theory

References

Debreu, G. (1960). Topological methods in cardinal utility theory. In *Mathematical methods in the social sciences* (pp. 16–26). Stanford, CA: Stanford University Press.

Dyer, J. S., & Sarin, R. K. (1979). Measurable multi-attribute value functions. *Operations Research, 27*, 810–822.

Edwards, W., & von Winterfeldt, D. (1986). *Decision analysis and behavioral research*. Cambridge: Cambridge University Press.

Farquhar, P. H. (1977). A survey of multiattribute utility theory and applications. *TIMS Studies in Management Science, 6*, 59–89.

Fishburn, P. C. (1964). *Decision and value theory*. New York: Wiley.

Fishburn, P. C. (1965a). Independence in utility theory with whole product sets. *Operations Research, 13*, 28–45.

Fishburn, P. C. (1965b). Utility theory. *Management Science, 14*, 335–378.

Gorman, W. M. (1968). Symposium on aggregation: The structure of utility functions. *Review of Economic Studies, 35*, 367–390.

Keeney, R. L. (1969). *Multidimensional utility functions: Theory, assessment, and applications (Technical Report No. 43)*. Cambridge: Operations Research Center, M.I.T.

Keeney, R. L., & Raiffa, H. (1976). *Decisions with multiple objectives: Preferences and value tradeoffs*. New York: Wiley.

Luce, R. D., & Tukey, J. W. (1964). Simultaneous conjoint measurement: A new type of fundamental measurement. *Journal of Mathematical Psychology, 1*, 1–27.

Luce, R. D., Bush, R. R., & Galantor, E. (1965). *Handbook of mathematical psychology* (Vol. 3). New York: Wiley.

Sarin, R. K. (1975). *Interactive procedures for evaluation of multi-attributed alternatives*. Working paper 232. Los Angeles: Western Management Science Institute, University of California.

von Neumann, J., & Morgenstern, O. (1947). *Theory of games and economic behavior*. Princeton, NJ: Princeton University Press.

Multi-Criteria Decision Making (MCDM)

- ▶ Multiple Criteria Decision Making

Multi-Echelon Inventory Systems

Inventory systems comprised of multiple stages of inventory control decision making, e.g., in a supply chain, there are inventory decisions to be made at the production facility, the distributor, and the retail outlet, among others.

See

► Inventory Modeling

Multi-Echelon Logistics Systems

Logistics systems comprised of several layers of individual logistics problems.

See

► Logistics and Supply Chain Management

Multiple Criteria Decision Making

Ramaswamy Ramesh and Stanley Zionts
University at Buffalo, The State University of New York, Buffalo, NY, USA

Introduction

Multiple Criteria Decision Making (MCDM) refers to making decisions in the presence of multiple, usually conflicting, objectives. Multiple criteria decision problems pervade almost all decision situations ranging from common household decisions to complex strategic and policy level decisions in corporations and governments. Prior to the development of MCDM as a discipline, such problems have been traditionally addressed as single-criterion optimization problems by (i) deriving a composite measure of the objectives and optimizing it, or (ii) by choosing one of the objectives as the main decision objective for optimization and solving the problem by requiring an acceptable level of achievement in each of the other objectives. MCDM as a discipline was founded on two key concepts of human behavior, introduced and explored in detail by Herbert Simon in the 1950s: satisficing and bounded rationality (Simon 1957). The two are intertwined because satisficing involves finding solutions that satisfy constraints rather than optimizing the objectives, while bounded rationality involves setting the constraints and then searching for solutions satisfying the constraints, adjusting the constraints, and then continuing the process until a satisfactory solution is found. The rest of this article overviews important aspects of MCDM, including basic concepts, a taxonomy, modeling techniques, and algorithms.

Basic Concepts

An MCDM problem can be broadly described as follows. Let $D = \{d_1,\ldots, d_n\}$ denote the decision space, comprising the set of possible decision alternatives to a problem. Let $C = \{C_1,\ldots, C_p\}$ denote the objective space, comprising of a set of p mutually conflicting objectives. Without loss of generality, assume all objectives are to be maximized. Let E: $D \rightarrow C$ be a mapping of the decision space on to the objective space, where $E(d_i)$ is the vector $(C_1^i,\ldots, C_p^i)$. Each element of this vector is an assessment, or the value of the corresponding objective provided by the decision alternative d_i. A fundamental concept in MCDM is that of dominance, defined as follows.

***Definition* 1 (*Dominance*).** A decision alternative d_i said to be dominated by another alternative d_j if $C_k^i \leq C_k^j$, $k = 1,\ldots, p$ with at least one strict inequality.

In the above definition, if all the inequalities hold as strict inequalities, then the dominance is said to be strong; otherwise, it is called weak. The following concept is a logical extension of the dominance concept.

***Definition* 2 (Convex *Dominance*).** An alternative d_i is said to be convex dominated by a subset $\hat{D} \subset D$ if it

is dominated by a convex combination of the alternatives in $\hat{D}$

The above definitions lead to a central theme of all MCDM techniques as follows.

***Definition* 3 (*Efficiency*).** An alternative d_j is said to be efficient or nondominated in D if there is no other alternative in D that dominates it, even weakly.

The concept of efficiency can be extended to convex dominance as well. In this case, an efficient alternative is known as convex-efficient or convex-nondominated. The following theorem of Geoffrion (1968) shows how the efficiency of an alternative can be determined. Zionts and Wallenius (1980) introduced a different but equivalent methodology that solves a number of problems including that one.

***Theorem* 1.** Consider any decision alternative d_i and its mapping on the objective space $(C^i_1,\ldots, C^i_p)$. The decision d_i is efficient if only if the following linear program is unbounded:

$$\text{Maximize} \sum_{j=1}^{p} w_j C^i_j$$

$$\text{subject to} \sum_{j=1}^{p} w_j C^k_j \leq 0,\ k = 1,\ldots,n, k \neq i$$

$$w_j \geq 0,\ j = 1,\ldots,p.$$

A Taxonomy of MCDM Methods

The MCDM methods proposed in the literature cover a wide spectrum, and there are several alternative ways of organizing them into a taxonomy. The taxonomy described here is based on Chankong et al. (1984), which is one of the interpretations of the world of MCDM models. At the outset, MCDM methods can be classified into two broad classes: vector optimization methods and utility optimization methods. Vector optimization is primarily concerned with the generation of all efficient decision alternatives. These methods do not require intervention of a decision maker. These methods do generate a subset of nondominated solutions. Some of the well-known vector optimization methods include those of Geoffrion (1968), Villarreal and Karwan (1981), and Yu and Zeleny (1976).

Multiple Criteria Decision Making, Table 1 A taxonomy of MCDM approaches

Decision outcome	Decision space	
	Explicit	Implicit
Deterministic	Deterministic Multiattribute Decision Analysis	Deterministic Multiobjective Mathematical Programming
Stochastic	Stochastic Multiattribute Decision Analysis	Stochastic Multiobjective Mathematical Programming

The utility optimization methods can be broadly organized according to the following dimensions (see, for example, Zionts 1979):

1. Nature of decision space: Explicit or Implicit; and
2. Nature of decision outcomes: Stochastic or Deterministic.

In an explicit decision space, decision alternatives are stated explicitly. A classical example is the home buying problem, where a decision maker/home buyer is faced with a set of possible homes to consider purchasing. For an implicit decision, alternatives are stated using a set of constraints, such as in linear or nonlinear programming where a feasible alternative must satisfy the constraints. An implicit decision situation can be further categorized as continuous or discrete. The decision outcomes are stochastic or deterministic depending on whether the mapping function E: $D \rightarrow C$ is stochastic or deterministic. Table 1 classifies MCDM methods broadly along the two dimensions. There are many approaches in the various segments of this classification. Here, the discussion focuses on the best-known methods.

Methodological Approaches

Deterministic Decision Analysis — Deterministic decision analysis is concerned with finding the most preferred alternative in decision space by constructing a value function representing a decision maker's preference structure, and then using the value function to identify the most preferred solution. A value function $v(C_1, C_2,\ldots, C_p)$ is a scalar-valued

function defined with the property that $v(C_1, C_2,\ldots, C_p) > v(C',\ldots, C'_p)$ if and only if $(C_1, C_2,\ldots, C_p)$ is at least as preferred as $(C'_1,\ldots, C'_p)$ (Keeney and Raiffa 1976). The construction of the value function involves choice decisions made by the decision maker. Generating value functions is simplified if certain conditions hold, in which case it is possible to decompose the above functions into partial value functions $v_k(C_k)$ for each value of k.

The decomposition and certain simplifications of the value function may be carried out if certain underlying assumptions on the decision maker's preference structure hold. One of these is preferential independence, which is stated as follows: Consider a subset of objectives denoted as $\hat{C}$. If the decision maker's preferences in the space $C - \hat{C}$ are the same for any set of arbitrarily fixed levels of the objectives $\hat{C}$, then $\hat{C}$ is said to be preferentially independent of $C - \hat{C}$. The set C is said to be mutually preferentially independent if every subset of C is preferentially independent of its complement with respect to C. When mutual preferential independence holds, an additive value function of the form

$$v(d_i) = \sum_{k=1}^{p} \lambda_k v_k\left(C_k^i\right) \text{ where } \lambda_k \text{ is a scalar constant}$$

is appropriate. There are other nonlinear forms that can be used as well. Of course, an additive value function, if appropriate, is highly desirable. Once the value function has been determined, it can be used to evaluate and rank the alternatives.

Stochastic Decision Analysis — Stochastic decision analysis is similar to the deterministic case, except that the outcomes are stochastic, and utility functions are constructed instead of value functions. The ideas are similar. There is an analogous condition to that described for the discrete case above. It involves utility independence. A subset of objectives $\hat{C}$ is utility independent of its complement if the conditional preference order for lotteries involving changes in $\hat{C}$ does not depend on the levels at which the objectives in $\hat{C}$ are fixed. Since utility independence refers to lotteries and preferential independence refers to deterministic outcomes, utility independence implies preferential independence, but not vice versa. Analogous to mutual preferential independence, the set C is said to be mutually utility independent if every subset of C is utility independent of its complement with respect to C. Keeney and Raiffa (1976) show that if C is mutually utility independent, then a multiplicative utility function is appropriate. This function is of the form

$$u(d_i) = \prod_{k=1}^{p} \mu_k u_k\left(C_k^i\right),$$

where $u(d_i)$ is the overall utility of the decision alternative d_i, $u_k(C_k^i)$ is the utility of its kth objective component, and μ_k is a scalar constant. A more stringent set of assumptions must hold in order that the utility function be additive. In the stochastic case, not only must a utility function be estimated, but probabilities of various outcomes must also be estimated by the decision maker.

Multiobjective Mathematical Programming — Considerable work has been done in the multiobjective mathematical programming area. These include Multiobjective Linear Programming (MOLP) and Multiobjective Integer Programming (MOIP). Goal programming (Lee 1972), the method of Zionts and Wallenius (1976, 1983), the Step Method of Benayoun et al. (1971), and the method of Steuer (1976) are some of the better-known MOLP methods. Goal programming and the method of Zionts and Wallenius are now described in more detail.

Goal programming is an extension of linear programming and was proposed by Charnes and Cooper in 1961. A description of this technique is as follows. Consider the following MOLP problem:

$$\begin{aligned} &\text{Maximize } \boldsymbol{Cx} \\ &\text{subject to } \quad \boldsymbol{Ax} \le \boldsymbol{b} \qquad \text{(MOLP)} \\ &\boldsymbol{x} \ge \boldsymbol{0} \end{aligned}$$

where $\boldsymbol{C} = (c_{kj})$ is a $(p \times n)$ matrix, $\boldsymbol{A}$ is an $(m \times n)$ matrix and $\boldsymbol{x}$ is an $(n \times 1)$ vector. Let $(\alpha_1,\ldots, \alpha_p)$ denote the goals with respect to the desired levels of attainment in the objectives specified by a decision maker. Introduce over and under attainment variables y_k^+ and y_k^- for each objective and add the following constraints, where $\boldsymbol{c}_k$ is the kth row of $\boldsymbol{C}$:

$$\boldsymbol{c}_k\boldsymbol{x} - y_k^+ + y_k^- = \alpha_k, \quad k - 1,\ldots, p.$$

Let w_k denote the penalty for the net deviation from the goal of objective $k = 1,\ldots, p$. Then the goal programming problem is formulated as follows:

$$\text{Minimize} \quad \sum_{j=1}^{p} w_k\left(y_k^+ + y_k^-\right) \qquad \text{(GP)}$$

$$\text{subject to} \quad \sum_{j=1}^{n} c_{kj}x_j - y_k^+ + y_k^- = \alpha_k, \quad k = 1,\ldots, p$$

$$\boldsymbol{Ax} \leq \boldsymbol{b}$$

$$\boldsymbol{x}, \boldsymbol{y} \geq \boldsymbol{0}$$

The above problem minimizes a weighted sum of deviations from the desired goals, where weights are required from the decision maker. The goal programming formulation is an attempt to find a solution that is closest to the decision maker's desired goals, while also responding to his differential emphasis on the nonattainment of the various goals.

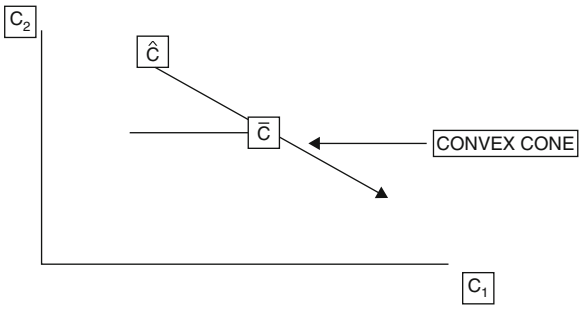
The Zionts and Wallenius method follows an interactive approach using pairwise evaluations of decision alternatives by a decision maker to solve problem MOLP. The method starts by choosing an initial set of weights $\lambda \in R^p$, and maximizing a linear composite objective $\lambda\boldsymbol{Cx}$. This generates a corner point of $\{\boldsymbol{Ax} \leq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}\}$ that is efficient. Call this solution $\boldsymbol{x}^0$. Next, the adjacent corner points of $\boldsymbol{x}^0$ that are also efficient (and whose edges leading to them are *also* efficient) are determined. Call this set S^0. The decision maker is asked to choose between $\boldsymbol{x}^0$ and a solution from S^0 until: (i) either he or she prefers $\boldsymbol{x}^0$ to all the points in S^0, or (ii) prefers some solution in S^0 to $\boldsymbol{x}^0$. If $\boldsymbol{x}^0$ is preferred to all the points in S^0, then the method stops with $\boldsymbol{x}^0$ as a "locally" best preferred corner-point solution. Otherwise, if some solution in S^0 is preferred to $\boldsymbol{x}^0$, then it is devoted as $\boldsymbol{x}'$. Linear constraints of the form $\lambda\,(\boldsymbol{Cx}' - \boldsymbol{Cx}'') \leq -\epsilon$ where $\boldsymbol{x}'$ is preferred to $\boldsymbol{x}''$ and ϵ is a small positive quantity are generated from the decision maker's pairwise preferences. A new set of weights that satisfy these constraints are then obtained. If these constraints are in conflict, then some of them are dropped in determining the new weights. Call the new set λ''. Maximizing the composite objective $\lambda''\boldsymbol{Cx}$, a new efficient corner point is generated, and the above steps are repeated until a corner point that is preferred to all its adjacent efficient corner points is obtained.

Compared to MOLP, research on MOIP is rather limited. Some of the earlier works on MOIP have been in the domain of vector optimization. Bitran and Rivera (1982) provided an implicit enumeration algorithm for determining the efficient set of 0-1 MOIP problems. Pasternak and Passy (1973) studied the vector optimization problem for two objectives. Klein and Hannan (1982) extended Pasternak and Passy's work to more than two objectives. Villarreal and Karwan (1981) generalized the classical dynamic programming recursions to a multicriteria framework. Ramesh et al. (1989) followed the utility optimization approach to find the most preferred solution to an MOIP problem.

The method of Ramesh et al. (1989) follows a branch-and-bound search strategy using the Zionts and Wallenius method for bounding. The decision maker's preference structure is assessed using pairwise evaluations and an internal representation of the preference structure is successively built during the course of the branch-and-bound search. This representation is used to deduce the decision maker's preferences wherever possible so that the cognitive load arising out of the pairwise judgments can be minimized. The internal representation is based on the concept of convex cones as described below (Korhonen et al. 1984).

Consider a two-dimensional objective space as shown in Fig. 1. Let $\bar{C}$ and $\hat{C}$ be two points in this space such that $\hat{C}$ is preferred to $\bar{C}$. Assuming a quasiconcave and nondecreasing utility function for the decision maker, it follows that every point falling on the ray $\left\{\hat{C} | \hat{C} = \hat{C} + \mu\,(\bar{C} - \hat{C}), \mu \geq 0\right\}$ is less preferred than $\hat{C}$ and no more preferred than $\bar{C}$. Consequently, every point in this ray and those dominated by it can be eliminated from consideration. This ray is called a convex cone, and is illustrated in Fig. 1. Every pairwise judgment of a decision maker yields a convex cone and the cones are ordered into a tree structured to eliminate search regions efficiently and minimize the need for the decision maker's pairwise evaluations throughout the search procedure.

Other Explicit Decision Space Methods — Several methods have been proposed for finding the most preferred alternative from an explicitly stated decision space without estimating a value function. These techniques are methods of deterministic decision analysis, and there is substantial interest in



Multiple Criteria Decision Making, Fig. 1 Illustration of convex cones

these problems. Three important methods in this category are the Multiple Criteria Decision Making (MCDM) Analytic Hierarchy Process (Saaty 1980), the method of Korhonen et al. (1984), and the AIM method (Lotfi et al. 1992).

The idea of Analytic Hierarchy Process (AHP) is that one can structure a problem hierarchically, and then make judgments regarding the relative importance of various aspects of the problem. As a result of these judgments, a ranking is produced. A simple decision problem would have a hierarchy that consists of three levels, from the top down: 1) the goal; 2) the criteria involved; and 3) the alternatives. The number of levels depends on the nature of the problem involved. In general, consider an n-alternative, p-criteria problem. Then the decision maker is asked to fill in entries in $p + 1$ reciprocal matrices as follows:

1. One $(p \times p)$ matrix relating each criterion to all others; and
2. P $(n \times n)$ matrices, each relating one criterion to all alternatives.

Each reciprocal matrix has all diagonal elements one, and off-diagonal elements reciprocal, that is, $a_{ij} = 1/a_{ji}$. Accordingly, the decision maker need only provide just less than half the entries, more specifically, the $[p(p-1)/2] + p[n(n-1)/2]$ off-diagonal (lower or upper) entries in the matrix. Though the amount can be reduced to as few as $(p-1) + p(n-1)$ entries (having no redundancy), the reduction in information required increases the cognitive load on the decision maker to provide entries, and does not provide the redundancy and cross checking that furnishing the complete input provides.

In filling in the matrices, the decision maker is asked to provide numbers between 1/9 and 9 reflecting the relative importance between the aspects involved. One of the matrices reflects the comparison among criteria and the p other matrices reflect evaluations of alternatives with respect to each criterion. AHP next solves for the right eigenvector, or characteristic vector, of each matrix. An eigenvector of a matrix may be estimated by taking the geometric mean of the elements of each row of the matrix (for a $p \times p$ matrix, the pth root of the product of the p elements of a row), and then normalizing the resulting vector so that the sum of the elements is unity. The consistency of the matrix (as differentiated from a matrix generated at random) may be tested using a calculation on the matrix. By the user furnishing fewer than all $p(p-1)/2$ entries required in the matrix, the test on consistency is compromised. The scaled eigenvectors are then used to score and rank each alternative.

Korhonen et al. (1984) presented an interactive method employing pairwise comparisons for solving the discrete, deterministic MCDM problem. Assuming a quasiconcave and nondecreasing utility function, they introduce the concept of convex cones. Choosing an arbitrary set of positive weights w_i, $i = 1,\ldots, p$, a composite linear utility function is initially generated. Using the composite as a proxy for the true utility function, the decision alternative maximizing the composite is generated. Call this solution d^0. Using the mapping E: $D \rightarrow C$, all adjacent efficient decision alternatives to d^0 (as in the Zionts-Wallenius method) are determined. This is done for the region that consists of all convex combinations of feasible solutions. Call the set of such solutions S^0. The decision maker is asked to choose between d^0 and some solution from S^0. Based on the response, a constraint on the weights is generated, as in the Zionts and Wallenius method for MOLP, and a convex cone is derived. Any solution in the set S^0 dominated by the cone is removed from S^0, and the above step is repeated until either d^0 is preferred to all solutions in S^0 or some solution in S^0 is preferred to d^0. The constraints on the weights and the convex cones generated at each iteration of this step are accumulated. The set of cones is used to deduce the decision maker's preferences wherever possible. This reduces the search space, while also minimizing the number of pairwise comparisons the decision maker has to perform.

Every solution in S^0 that is less preferred than d^0 is dropped from consideration. If d^0 is preferred to all the solutions in S^0, then it is denoted as d. If some solution in S^0 is preferred to d^0, then the preferred solution is denoted as d. If d is the only efficient solution remaining in the decision space, then the procedure stops with d as the most preferred decision. Otherwise, choosing a set of weights consistent with the weight constraints (after dropping any conflicting constraints), a new composite linear utility function is generated. Denoting the decision alternative maximizing this composite function as d, the decision maker chooses between d and d. Denoting the preferred solution as d^0, the above steps are repeated.

Lotfi et al. (1992) develop an eclectic method called the Aspiration-Level Interactive Method (AIM) for MCDM. It involves a philosophy that aspiration levels and feedback regarding the relative feasibility of the aspiration levels provide a powerful tool for decision making. The method is embodied in a computer program called AIM. The method provides the decision maker with various kinds of feedback as he explores the solutions. Several different kinds of objectives may be included: objectives to be maximized; objectives to be minimized; target objectives; any of the above kinds of objectives with thresholds, or levels beyond which the user is indifferent to further gains in the objective; and qualitative objectives. To further explain the idea of thresholds, suppose that in the purchase of a house, the age of the house is an attribute to be minimized. Suppose further that the buyer treats as equivalent, however, any houses ten years or less in age. In this case, there is a threshold of ten years, so that an eight-year-old house is considered to be no better than a ten-year-old house with respect to age.

To begin with, the decision maker has the following basic information:

1. A current goal or aspiration level for each objective, initially set to the median, together with the proportion of alternatives having values of the objective at least as good as that value.
2. Two other aspiration levels, the next better and the next worse than the current goal occurring in the data base.
3. The ideal and nadir solutions to the problem.
4. The proportion of alternatives that simultaneously satisfy aspiration levels given in 1 and 2.
5. A nearest nondominated solution to the current goal. The nearest solution is found by mapping the current goal to a solution on the efficient frontier or in the set of nondominated solutions.

The current goal may be (and should be) changed by the user, component by component, to any desired realizable level of any objective. The intention, however, is to keep the current goal near the efficient frontier and therefore nearly achievable. As the user changes the current goal, all but item(s) 3 above change.

The user can invoke various options to help in decision making. He or she can see which solutions, if any, satisfy his current goal. Second, he or she can obtain a ranking of solutions based on a function resulting from his choice of a current goal. Third, he or she can use a simplified version of a concept called outranking to identify neighbor solutions that are similar to his nearest solution. The decision maker may also review the weights implied by the current goal, see a quartile distribution of the problem by objective, and identify and possibly delete dominated solutions.

Concluding Remarks

The field of multiple criteria decision making has been an active since the 1960s. Many interesting approaches have been developed, explored, and implemented in solving problems. Implementation of MCDM methodologies include multiple criteria decision support systems (MCDSS) and negotiations, which may be regarded as multiple criteria problems involving multiple decision makers. MCDSS integrate the multiple criteria approaches in user-friendly microcomputer systems, such as the VIG/VIMDA system of Korhonen and Laakso (1986), the Expert Choice software that implements AHP, and the AIM package of Lotfi et al. (1992) implemented on the World Wide Web by Wang and Zionts (2005). An objective of most of the MCDSS is to provide inexpensive stand-alone software that is easy to use. A very useful set of computer MCDM method software may be found on the World Wide Web by a search on the word decisionarium; the software is housed at the

Helsinki University of Technology, now part of Aalto University.

Negotiations or multiperson MCDM is a natural extension of MCDM. Many decisions are made by groups, and negotiation theory involves using some of the MCDM concepts to simplify and assist negotiations; see for example, Wang and Zionts (2008).

In addition to the journals devoted to management science and operations research and behavioral science, there are two journals that contain articles more exclusively in this area: *Multi-Criteria Decision Analysis* and *Group Decision and Negotiation*. The paper by Wallenius et al. (2008) explores recent accomplishments and what lies ahead.

See

- ▶ Analytic Hierarchy Process
- ▶ Analytic Network Process
- ▶ Decision Analysis
- ▶ Decision Problem
- ▶ Goal Programming
- ▶ Multi-attribute Utility Theory
- ▶ Multiobjective Programming
- ▶ Utility Theory
- ▶ Value Function

References

Benayoun, R., De Montgolfier, J., Tergny, J., & Larichev, O. (1971). Linear programming with multiple objective functions: Step method (STEM). *Mathematical Programming, 1*, 366–375.

Bitran, G. R., & Rivera, J. M. (1982). A combined approach to solving binary multicriteria problems. *Naval Research Logistics, 29*, 181–201.

Chankong, V., Haimes, Y. Y., Thadathil, J., & Zionts, S. (1984). Multiple criteria optimization: A state of the art review. In *Decision making with multiple objectives* (pp. 36–90). Berlin: Springer.

Geoffrion, A. M. (1968). Proper efficiency and the theory of vector maximization. *Journal of Mathematical Analysis and Applications, 22*, 618–630.

Keeney, R. L., & Raiffa, H. (1976). *Decisions with multiple objectives: Preferences and value trade-offs*. New York: John Wiley.

Klein, D., & Hannan, E. (1982). An algorithm for the multiple objective integer linear programming problem. *European Journal of Operational Research, 9*, 378–385.

Korhonen, P., & Laakso, J. (1986). A visual interactive method for solving the multiple criteria problem. *European Journal of Operational Research, 24*, 277–287.

Korhonen, P., Wallenius, J., & Zionts, S. (1984). Solving the discrete multiple criteria problem using convex cones. *Management Science, 30*, 1336–1345.

Lee, S. M. (1972). *Goal programming for decision analysis*. Philadelphia: Auerbach Publishers.

Lotfi, V., Stewart, T. J., & Zionts, S. (1992). An aspiration-level interactive model for multiple criteria decision making. *Computers and Operations Research, 19*, 671–681.

Lotfi, V., Yoon, Y. S., & Zionts, S. (1997). Aspiration-based search algorithm (ABSALG) for multiple objective linear programming problems: Theory and comparative tests. *Management Science, 43*, 1047–1059.

Pasternak, H., & Passy, V. (1973). Bicriterion mathematical programs with boolean variables. In *Multiple criteria decision making*. Columbia: University of South Carolina Press.

Ramesh, R., Karwan, M. H., & Zionts, S. (1989). Preference structure representation using convex cones in multicriteria integer programming. *Management Science, 35*, 1092–1105.

Saaty, T. L. (1980). *The analytic hierarchy process*. New York: McGraw-Hill.

Simon, H. (1957). *Administrative behavior*. New York: The Free Press.

Steuer, R. E. (1976). Multiple objective linear programming with interval criterion weights. *Management Science, 23*, 305–316.

Villarreal, B., & Karwan, M. H. (1981). Multicriteria integer programming: A (hybrid) dynamic programming recursive approach. *Mathematical Programming, 21*, 204–223.

Wallenius, J., Dyer, J., Fishburn, P., Steuer, R., Zionts, S., & Deb, K. (2008). Multiple criteria decision making/ multiattribute utility theory: Recent accomplishments and what lies ahead. *Management Science, 54*, 1336–1349.

Wang, J. G., & Zionts, S. (2005). WebAIM: An online aspiration-level interactive method. *Multi-Criteria Decision Analysis, 13*, 51–63.

Wang, J. G., & Zionts, S. (2008). Negotiating wisely: Considerations based on multi-criteria decision making/ multi-attribute utility theory. *European Journal of Operational Research, 188*, 191–205.

Yu, P. L., & Zeleny, M. (1976). Linear multiparametric programming by multicriteria simplex method. *Management Science, 23*, 159–170.

Zionts, S. (1979). MCDM: If not a roman numeral, then what? *Interfaces, 9*, 94–101.

Zionts, S., & Wallenius, J. (1976). An interactive programming method for solving the multiple criteria problem. *Management Science, 22*, 652–663.

Zionts, S., & Wallenius, J. (1980). Identifying efficient vectors: Some theory and computational results. *Operations Research, 28*, 788–793.

Zionts, S., & Wallenius, J. (1983). An interactive multiple objective linear programming method for a class of underlying nonlinear utility functions. *Management Science, 29*, 519–529.

Multiple Optimal Solutions

In an optimization problem, when different feasible solutions yield the same optimal value for the objective function, the problem has multiple optimal solutions. If a linear-programming problem has multiple optimal solutions, then such solutions correspond to extreme point solutions and their convex combinations.

See

▶ Unique Solution

Multiple Pricing

When solving a linear-programming problem using the simplex method, it is computationally efficient to select a small number, say 5, possible candidate vectors from which one would be chosen to enter the basis. The candidate set consists of columns with large (most negative or most positive) reduced costs, and the vector in this set that yields the largest change in the objective function is selected. Succeeding iterations only consider candidate basis vectors from the vectors that remain in the set that have properly signed reduced costs. When all vectors in the set are chosen or none can serve to change the objective function in the proper direction, a new set is determined.

See

▶ Partial Pricing
▶ Simplex Method (Algorithm)

Multiplier Vector

For a given feasible basis $\boldsymbol{B}$ to a linear-programming problem, let the row vector $\boldsymbol{c_B}$ be the ordered set of cost coefficients for the vectors in $\boldsymbol{B}$. The multiplier vector is defined as $\boldsymbol{\pi} = \boldsymbol{c_B}\,\boldsymbol{B}^{-1}$. If $\boldsymbol{B}$ is an optimal basis, then the components of $\boldsymbol{\pi}$ are the dual variables associated with the corresponding primal constraints. The vector $\boldsymbol{\pi}$ is also called the simplex multiplier vector, with the components of $\boldsymbol{\pi}$ being the simplex multipliers.

See

▶ Simplex Method (Algorithm)

Multivariate Quality Control

Francis B. Alt[1] and Scott D. Grimshaw[2]
[1]University of Maryland, College Park, MD, USA
[2]Brigham Young University, Provo, UT, USA

Introduction

A frequent quality control application in the chemical and process industries is the simultaneous monitoring of several correlated quality measurements. For example, González and Sánchez (2010) apply multivariate quality control to manufacturing the window frame for the door of a vehicle, where the five gaps on the window frame are measured at seven locations on the frame. Control charts that simultaneously evaluate all the information available on a process are based on the foundational work of Hotelling (1947) in a military application. While one could create univariate control charts for each measurement, ignoring the correlation between measurements impacts the statistical properties in many ways. Jackson (1956) showed that the use of univariate control charts can be misleading even when the measured characteristics are uncorrelated. Alt (1985) points out that not only is it statistically inefficient to monitor each measurement on its own control chart because the proper out-of-control region is elliptical, the process may exhibit frequent false out-of-control alarms.

Multivariate quality control procedures can be classified into two broad categories: (1) Shewhart procedures designed to quickly detect large out-of-control shifts from the in-control mean vector, and (2) Multivariate EWMA procedures that can be designed to efficiently detect persistent small

and moderate shifts. These are discussed in turn, followed by a discussion of other important methods for multivariate quality control.

Shewhart Charts

At regular time intervals, observe a rational subgroup of size n on p quality characteristics denoted by the vector $\mathbf{x}_i$. When the process is in-control, the quality characteristics will have mean $\boldsymbol{\mu}_0$ and variance-covariance matrix $\boldsymbol{\Sigma}_0$.

The Shewhart χ^2 chart produces an out-of-control signal when

$$\chi^2 = n(\bar{\mathbf{x}} - \boldsymbol{\mu}_0)'\boldsymbol{\Sigma}_0^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu}_0)$$

exceeds the upper control limit. The $\bar{\mathbf{x}}$ is the mean of each quality characteristic for the rational subgroup assembled as a $p \times 1$ vector.

The performance of a control chart is judged by its average run length (ARL), which is the average number of time periods taken before an out-of-control signal is given. A control chart is designed to have a large in-control ARL and a small out-of-control ARL. For multivariate Shewhart charts the upper control limit defines the in- and out-of-control ARL. The run length of Shewhart control charts follows a geometric distribution since each time interval is independent and the probability of an out-of-control signal is identical for each time interval. If ARL_0 denotes the in-control ARL, the upper control limit (UCL) is $\chi^2(1/ARL_0; p)$, the $100(1-(1/ARL_0))\%$ percentile of the χ^2 distribution with p degrees of freedom, if the $\bar{\mathbf{x}}$ is multivariate normal. The most frequent choice is $ARL_0 = 200$, so the upper control limit is the 95% percentile of the χ^2_p. When the process is out-of-control with mean $\boldsymbol{\mu}_1$, the multivariate Shewhart statistic has a non-central χ^2 distribution with p degrees of freedom and non-centrality parameter

$$\lambda = \sqrt{n(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)'\boldsymbol{\Sigma}_0^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)},$$

and the out-of-control ARL, denoted by ARL_1, can be computed $ARL_1 = 1/[1 - F(UCL; p, \lambda)]$, where $F(\cdot; p, \lambda)$ is the cdf of a non-central χ^2.

A frequent obstacle to applying the Shewhart χ^2 control chart is the need for the in-control variance-covariance matrix $\boldsymbol{\Sigma}_0$. The Hotelling T^2 distribution, a generalization of the Student's t distribution, allows the estimated variance-covariance matrix $\mathbf{S}$ to replace $\boldsymbol{\Sigma}_0$. The Shewhart T^2 control chart compares the statistic

$$T^2 = n(\bar{\mathbf{x}} - \boldsymbol{\mu}_0)'\mathbf{S}^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu}_0)$$

to the upper control limit

$$\text{UCL} = \frac{p(n-1)}{n-p} F(1/\text{ARL}_0; p, n-p)$$

which uses a well-known relationship between the Hotelling T^2 distribution and the F distribution.

In many applications, the in-control mean $\boldsymbol{\mu}_0$ and the in-control variance-covariance matrix $\boldsymbol{\Sigma}_0$ are unknown, but are estimated from data collected while the process is believed to be in-control. For this Phase I data of m time periods of rational subgroup size n, Alt (1982) proposed estimating $\boldsymbol{\mu}_0$ by the mean of the m sample mean vectors, denoted by $\bar{\bar{\mathbf{x}}}$, and estimating $\boldsymbol{\Sigma}_0$ by the pooled variance-covariance matrix which is the mean of the m sample variance-covariance matrices, denoted by $\mathbf{S}_p$. Because the in-control parameters are estimated, the upper control limit is inflated to

$$UCL = \frac{p(m-1)(n-1)}{mn-m-p+1} F(1/\text{ARL}_0; p, mn-m-p+1).$$

If any time period in Phase I has an out-of-control signal and an assignable cause is found, this time period is omitted and $\bar{\bar{\mathbf{x}}}$ and $\mathbf{S}_p$ are recomputed. This step is iterated until all $m^* < m$ time periods are considered in-control.

At this time, the monitoring of future time periods begins by using the statistic

$$T_f^2 = n\left(\bar{\mathbf{x}}_f - \bar{\bar{\mathbf{x}}}\right)'\mathbf{S}_\mathbf{p}^{-1}\left(\bar{\mathbf{x}}_f - \bar{\bar{\mathbf{x}}}\right)$$

with

$$\text{UCL} = \frac{p(m^*+1)(n-1)}{m^*n-m^*-p+1} F(1/\text{ARL}_0; p, m^*n-m^*-p+1)$$

where $\bar{\mathbf{x}}_f$ is a vector of sample means based on data for a time period after m^*. It is suggested that $\bar{\bar{\mathbf{x}}}$ and $\mathbf{S}_p$ be

updated fairly often in the beginning, as the number of future subgroups accumulates.

A common follow-up to a large T^2 statistic is to use standardized coefficients of the discriminant function (Rencher 2002, Chap. 5). That is, compute

$$\mathbf{a} = \text{sqrt}[\text{diag}(\mathbf{S})] \cdot \mathbf{S}^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu}_0),$$

where *sqrt* is the elementwise square root of the vector and diag($\mathbf{S}$) creates a diagonal matrix from the diagonal elements of the $\mathbf{S}$ matrix. The absolute values of the coefficients in $\mathbf{a}$ give relative contributions of each quality measurement to T^2. Another approach to interpreting a large T^2 value is a decomposition proposed by Mason et al. (1995, 1997). The T^2 can be written as p independent terms, each of which reflects the contribution of an individual quality characteristic. Runger et al. (1996) use this decomposition to improve diagnostics of an out-of-control signal.

MEWMA Charts

The multivariate exponentially weighted moving average (MEWMA) control charts are well suited to observing a single observation ($n = 1$) at each time period t and combining the information from a window of time to make a decision. The generalization from the univariate EWMA was formulated by Lowry et al. (1992). A weighted average of the observed $\mathbf{x}_t$ is formed by

$$\mathbf{Z}_t = \lambda(\mathbf{x}_t - \boldsymbol{\mu}_0) + (1 - \lambda)\mathbf{Z}_{t-1}$$

where the value λ is chosen in designing the control chart to represent the amount of smoothing ($0 < \lambda < 1$) and $\mathbf{Z}_0 = 0$. Small values of λ pool the data over a wide time interval and produce a control chart that effectively identifies small, persistent changes from the in-control mean, $\boldsymbol{\mu}_0$, or a gradual drift from $\boldsymbol{\mu}_0$. Large values of λ yield a $\mathbf{Z}_t$ with high weight on the current observation so the control chart is sensitive to immediate large shifts from $\boldsymbol{\mu}_0$.

The MEWMA chart signals a process is out-of-control at time t when

$$T_t^2 = \mathbf{Z}_t \boldsymbol{\Sigma}_{\mathbf{Z}}^{-1} \mathbf{Z}_t$$

exceeds an upper control limit. The variance-covariance matrix $\boldsymbol{\Sigma}_{\mathbf{Z}}$ depends on λ and t, and is given by

$$\boldsymbol{\Sigma}_{\mathbf{Z}} = \left(\frac{\lambda\left[1 - (1 - \lambda)^{2t}\right]}{2 - \lambda} \right) \boldsymbol{\Sigma}_0,$$

where $\boldsymbol{\Sigma}_0$ is the in-control variance-covariance matrix of $\mathbf{x}_t$. For a given λ, the upper control limit is chosen to provide an ARL for a specified out-of-control mean $\boldsymbol{\mu}_1$. Tables of the ARL for different p, λ, and upper control limit are given by Prabhu and Runger (1997) for in-control $ARL_0 = 200$.

In the univariate case, the CUSUM (cumulative sum) control charts are quite similar to the EWMA control charts. Although a number of multivariate CUSUM procedures have been proposed, an early suggestion by Woodall and Ncube (1985) was to monitor each of the p quality characteristics simultaneously with individual CUSUM charts. The ARL of this collection of p CUSUM control charts is the minimum of $\{ARL_1, ARL_2, \ldots, ARL_p\}$ if the quality characteristics are independent. If the quality characteristics are correlated, reduce the p dimensional space to the $p' < p$ largest principal components. An improvement to this collection of p CUSUMs is to update the CUSUM at each observation and shrink toward the zero vector as described in Crosier (1988).

Control Charts for Variance-Covariance

While monitoring the mean of p correlated quality characteristics has been well researched, less work has been performed on control charts for the variance-covariance matrix (the generalization from univariate control charts on process variability). The most common approach summarizes the $p(p + 1)/2$ variances and covariances in $\boldsymbol{\Sigma}$ into a scalar by defining the generalized variance $|\boldsymbol{\Sigma}|$, which is the determinant of $\boldsymbol{\Sigma}$. Montgomery and Wadsworth (1972) proposed control limits based on the asymptotic normality of $|\mathbf{S}|$, the determinant of the sample variance-covariance matrix based on the n observations in the rational subgroup. Control limits for the typical Shewhart control charts were proposed by Alt (1985) and are $E(|\mathbf{S}|) \pm 3\sqrt{Var(|\mathbf{S}|)}$ where $E(|\mathbf{S}|) = b_1|\boldsymbol{\Sigma}|$ and $Var(|\mathbf{S}|) = b_2|\boldsymbol{\Sigma}|^2$ with

$$b_1 = \frac{1}{(n-1)^p} \prod_{i=1}^{p} (n-i)$$

and

$$b_2 = \frac{1}{(n-1)^{2p}} \left[\prod_{i=1}^{p} (n-i) \right] \times \left[\prod_{j=1}^{p} (n-j+2) - \prod_{j=1}^{p} (n-j) \right].$$

Profile Monitoring

Many manufacturing processes in the chemical process and semiconductor industries have finite-duration processing periods under controlled conditions which result in the final product. With improved metrology these processes can be monitored during the processing time. In these applications the collection of measurements taken on each quality characteristic during processing when plotted over time creates a profile.

Nomikos and MacGregor (1995a) organized the large amount of profile data as a three-dimensional array whose n rows correspond to the different runs, t columns correspond to the measurements taken over processing time for a given run and the third dimension (depth) is the p different quality characteristics. While this is perhaps the organization of the data in a database, multivariate statistical methods require the expression of $\mathbf{Y}$ as a vector, and an 'unfolded' structure generates a tp vector of each quality characteristic at each processing time. Instead of monitoring this extremely large vector, one approach is to reduce the dimensionality to a set of summary scores $\mathbf{T}$. Nomikos and MacGregor (1995a) use principal components of $\mathbf{Y}$ to form $\mathbf{T}$, and Nomikos and MacGregor (1995b) use partial least squares to obtain linear combinations of $\mathbf{Y}$ which are highly correlated with a product's quality measurements taken after processing. Grimshaw et al. (1998) allow changing inputs that affect the profile and provide a real-time processing control chart statistic.

When there is a hypothesized relationship between the profile and an explanatory variable, the profile can be modeled using the parameters of the relationship. For example, if the relationship is linear the estimated regression coefficients are monitored using a Hotelling T^2 following Kang and Albin (2000). In a Phase II control chart where the profile has been estimated from historical data, Kim et al. (2003) address the linear case. The nonlinear profile case has been modeled by multiple regression and higher-order polynomials in Zou et al. (2007) and Kazemzadeh et al. (2008); nonparametric regression methods are used in Zou et al. (2008); and nonlinear profiles for dose–response applications are in Jensen and Birch (2009). Colosimo et al. (2008) monitor profiles of geometric specifications such as roundness, cylindricity, and flatness.

See

- ▶ Quality Control
- ▶ Total Quality Management

References

Alt, F. B. (1982). Multivariate quality control: State of the art. *ASQC Quality Congress Transactions – Detroit, MI*, pp. 886–893.

Alt, F. B. (1985). Multivariate quality control. In S. Kotz & N. S. Johnson (Eds.), *Encyclopedia of statistical sciences* (Vol. 6, pp. 110–122). New York: John Wiley.

Colosimo, B. M., Semeraro, Q., & Pacella, M. (2008). Statistical process control for geometric specifications: On the monitoring of roundness profiles. *Journal of Quality Technology, 40*(1), 1–18.

Crosier, R. B. (1988). Multivariate generalizations of cumulative sum quality control scheme. *Technometrics, 30*(3), 291–303.

González, I., & Sánchez, I. (2010). Variable selection for multivariate statistical process control. *Journal of Quality Technology, 42*(3), 242–259.

Grimshaw, S. D., Shellman, S. D., & Hurwitz, A. M. (1998). Real-time process monitoring for changing inputs. *Technometrics, 40*(4), 283–296.

Hotelling, H. (1947). Multivariate quality control, Illustrated by the air testing of sample bombsights. In C. Eisenhart, M. W. Hastay & W. A. Willis (Eds.), *Selected techniques of statistical analysis*. New York: McGraw–Hill.

Jackson, J. E. (1956). Quality control methods for two related variables. *Industrial Quality Control, 12*, 2–6.

Jensen, W. A., & Birch, J. B. (2009). Profile monitoring via nonlinear mixed models. *Journal of Quality Technology, 41*(1), 18–34.

Kang, L., & Albin, S. L. (2000). On-line monitoring when the process yields a linear profile. *Journal of Quality Technology, 32*(4), 418–426.

Kazemzadeh, R. B., Noorossana, R., & Amiri, A. (2008). Phase I monitoring of polynomial profiles. *Communications in Statistics: Theory and Methods, 37*(10), 1671–1686.

Kim, K., Mahmoud, M. A., & Woodall, W. H. (2003). On the monitoring of linear profiles. *Journal of Quality Technology, 35*(3), 317–328.

Lowry, C. A., Woodall, W. H., Champ, C. W., & Rigdon, S. E. (1992). A multivariate exponentially weighted moving average control chart. *Technometrics, 34*(1), 46–53.

Mason, R. L., Tracy, N. D., & Young, J. C. (1995). Decomposition of T^2 for multivariate control chart interpretation. *Journal of Quality Technology, 27*(2), 99–108.

Mason, R. L., Tracy, N. D., & Young, J. C. (1997). A practical approach for interpreting multivariate T^2 control chart signals. *Journal of Quality Technology, 29*(4), 399–406.

Montgomery, D. C., & Wadsworth, H. M. (1972). Some Techniques for Multivariate Quality Control Applications. *ASQC Technical Conference Transactions*, Washington, D. C, pp. 427–435.

Nomikos, P., & MacGregor, J. F. (1995a). Multivariate SPC charts for monitoring batch processes. *Technometrics, 37*(1), 41–59.

Nomikos, P., & MacGregor, J. F. (1995b). Multi-way partial least squares in monitoring batch processes. *Chemometrics and Intelligent Laboratory Systems, 30*, 97–108.

Prabhu, S. S., & Runger, G. C. (1997). Designing a multivariate EWMA control chart. *Journal of Quality Technology, 29*(1), 8–15.

Rencher, A. C. (2002). *Methods of multivariate analysis*. New York: Wiley.

Runger, G. C., Alt, F. B., & Montgomery, D. C. (1996). Contributors to a multivariate statistical process control chart signal. *Communications in Statistics: Theory and Methods, 25*(10), 2203–2213.

Woodall, W. H., & Ncube, M. M. (1985). Multivariate CUSUM quality control procedures. *Technometrics, 27*(3), 285–292.

Zou, C., Tsung, F., & Wang, Z. (2007). Monitoring general linear profiles using multivariate EWMA schemes. *Technometrics, 49*(4), 395–408.

Zou, C., Tsung, F., & Wang, Z. (2008). Monitoring profiles based on nonparametric regression methods. *Technometrics, 50*(4), 512–526.

Music

► Digital Music

N

Nash Equilibrium

In (noncooperative) game theory, a set of (pure or mixed) strategies in which no player can gain by deviating unilaterally. In general, a game may have a unique, multiple, or no equilibrium. John Nash, who shared the 1994 Nobel Prize in Economic Sciences for his work in game theory, proved that there must exist as least one such mixed strategy equilibrium in a finite-action game.

References

Nash, J. (1951) Non-cooperative games. *The annals of mathematics*, *second series* (Vol. 54, No. 2, pp. 286–295).

Nash Saddle-Point

► Game Theory
► Nash Equilibrium

Natural Resources

Andrés Weintraub
University of Chile, Santiago, Chile

Introduction

The field of natural resources covers various related areas: agriculture, fishing, forestry, mining. Though usually viewed separately, they share common problems, such as ecological concerns, use of scarce resources, and sustainability. There is also a common thread in what has happened since 1990. On the one hand, driven in part by population growth and economic development, many natural resources are beginning to reach or exceed sustainable levels of exploitation, or in the case of non-renewable resources there are limits on known reserves. A second main issue is the new awareness of the need to preserve natural habitats, protect endangered species, provide water and air quality, and promote biodiversity. This has often led to serious conflicts between production goals and ecological impacts, with increased public participation in decision processes. A third basic issue is the emergence of global, competitive markets with the need to derive efficient production processes. In this context, operations research and management science (OR/MS) have played a significant role in managing natural resources. It must be distinguished between methodological proposals through case studies and actual applications. This is an issue of importance. Typically, these problems are often complex, involve uncertainty and consider multiple objectives. Also, natural resources problems are often of large size and scope, with reliable data difficult to obtain. This partly explains the important gap that exists in some areas between modeling proposals shown through representative examples and actual use in planning and production processes. The introduction of personal computers, new information gathering systems, improved data processing, geographic information systems, satellite communication and algorithmic software plays a vital role in supporting

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

the used of OR/MS. A wide range of problems has been approached using typical OR/MS techniques in each area. In some cases, their solution has led to algorithmic developments. It is interesting to analyze the nature of the main problems in each area and the techniques proposed for their solutions, with linear programming, mixed integer programming and simulation being the techniques most commonly used across the areas. In the last decade the issue of integrating the supply chain has starting to emerge in some of the areas as important to improve productivity. Another emerging issue is the explicit introduction of uncertainty in some areas. In the last decade, climate change, carbon sequestration have become issues of increased research (Matthews et al. 2002). A state of the art in these four areas is presented in the Handbook of Operations Research in Natural Resources (Weintraub et al. 2007) and in Bjorndal et al. (2010).

Forestry

Most of the forestry issues are related to the management of forests. Native forests, often publicly owned, are viewed as multiple use entities, considering timber and range production, recreation, wildlife habitat preservation, water and soil quality. Plantations, such as pine or eucalyptus, are usually privately owned, sometimes integrated with pulp and sawmill processing plants, with timber production as their main objective within legal preservation regulations. Decisions in forestry management go from long-range planning to short-range operations.

Long-range planning, which, depending on the species under consideration, can go from 40 to more than 200 years to include up to two tree rotations, reflects basic silvicultural and economic options and in the case of plantations can include decisions on high-level investments in plants. Issues like sustainability of production, ecosystems, landscapes viewed in a strategic way are incorporated (Gunn 2007). A main objective is to maximize sustained long range production compatible with environmental preservation. Mathematical tools have been used successfully in this area. For the purpose of predicting tree growth under different management alternatives, simulation models based on regression techniques and sampling plots have proved reasonably accurate. Decision making has been supported mostly by linear programming models. Timber Ram (Navon 1971) was the first widely used LP model used for planning by the US Forest Service. FORPLAN, a multiple output linear programming model (Kent et al. 1991) and Spectrum (USDA 1995) were later introduced by the US Forest Service to incorporate increasingly environmental issues. Other LP models have been developed by private enterprises oriented towards managing plantations and are used in the USA, Canada, Europe, New Zealand, Chile, Brazil, and Australia.

Medium-range management decisions must consider spatial decisions, such as road building to access areas to be harvested. These decisions are the interface between strategic and operational decisions (Church 2007). One major spatial issue that has emerged during the last decades is that of spatial location of activities. Road building constitutes a major cost to reach areas to be harvested, in particular in native forests. Models have been proposed to integrate harvesting decisions and road building needed to reach harvest areas. These problems lead to mixed-integer programming (MIP) problems that have been successfully solved and used by forest enterprises (Kirby et al. 1986; Andalaft et al. 2003). Another spatial issue relates to the environment. To favor wildlife habitat or scenic beauty, adjacent blocks should be harvested in different periods to allow for new growth to establish itself. In this form, areas without tree growth have a maximum size. For example, some animals will graze only near cover provided by mature trees. These blocks were originally created manually by forests engineers clustering basic cells using a geographic information system (GIS). This problem adds considerable combinatorial complexity to the planning problems, particularly when combined with road building.) In the 1990s, meta-heuristic approaches, mainly Tabu search were used in practice. Exact approaches based on column generation techniques adding lifting constraints (Barahona et al. 1992), or strengthening of the formulation via cliques proved successful (Murray and Church 1996). In the late 1990s developments were proposed were the forming of the harvesting blocks were included into the problem. This approach led to better solutions, but to even more complex combinatorial problems. Again, meta-heuristic techniques were developed, mostly simulated annealing and tabu search (Murray 2007).

Exact algorithms have been proposed which can solve medium size problems (Goycoolea et al. 2009). Other applications consider problems as optimally locating wildlife habitat, selecting and locating vegetative seral stages to enhance faunal species diversity, (Hof and Bevers 1998). Wildlife protection can be incorporated through considering other spatial effects, such as the perimeter of forest areas where edge effects are important for some species, areas of old growth that allow wildlife to prosper, and corridors between old growth areas to let animals move between old growth areas. These problems lead to more difficult combinatorial models. (Hof and Haight 2007).

Short-term operations involve problems such as selection of units to cut, volumes to harvest, selection of bucking patterns, log allocation, selection of harvesting equipment and integration with downstream operations. (Epstein et al. 2007b) and transportation scheduling from forest sites to plants (Epstein et al. 2007a) A variety of models and algorithms have been proposed. Most used are linear programming based models for harvesting and allocation. For stem-bucking problems, dynamic programming and heuristic algorithms have been proposed. Scheduling of harvesting equipment such as towers, skidders or helicopters, decisions on road building, soil characterization and others related to locational aspects have been increasingly been carried out interacting with digital terrain models or geographic information systems. A successful application of OR/MS models to support operational decisions in transportation, machine harvesting scheduling and short term harvesting for Chilean forest firms was reported in Epstein et al. (1999). Since 2000, integrating the supply chain has started to be analyzed. Unlike other areas where sophisticated integration of supply chain activities has been successfully implemented, there have only been a few efforts in the forestry sector integrating forest harvesting, sawmills, pulp plants and secondary transformations such as panels (Carlsson and Ronnqvist 2005; D'Amours et al. 2008). In general at the operational level, OR has been most successful with multiple reported applications. Some of these problems are difficult to solve in exact formulations and heuristics have been necessary. For example, the problem of locating harvesting machinery and building access roads is a combination of a plant location problem and a network flow problem with fixed costs. A successful application solved using GIS-based data, a friendly graphical user interface and a heuristic algorithm is reported in Epstein et al. (2006).

Given the multiple uses of forests, it is only natural to view forest management as a multi-objective problem, considering diverse issues such as timber and range production, recreation, scenic beauty, preservation of endangered species, wildlife habitat, water quality, costs, income, carbon emissions and social impacts. The most common approaches to handle these problems have been through goal programming, or multi-objective linear programming. Multicriteria methods (Diaz-Balteiro and Romero 2007), where preferences are elicited from decision makers via comparisons, such as AHP have also been proposed. However, these developments have seldom been adopted by practitioners, mostly due to the difficulties in implementation.

The explicit treatment of risk and uncertainty has received increased attention of forest planners. The main issues related to uncertainty are in future timber markets, timber growth and yield projections and the possibility of catastrophes such as large fires or pests. Basic approaches proposed to handle uncertainty are : (a) parametric or scenario analysis (b) probability-based models such as stochastic dynamic programming, portfolio theory, chance-constrained linear programming, and simulation; and (c) fuzzy models, in which a certain ambiguity is assumed for restrictions or parameters. These efforts are still mostly at a developmental stage with few applications reported (Martell et al. 1998; Lohmander 2007).

Hierarchical Planning. Forestry problems range from decisions involving spatial concerns over 20 acres to entire forests of 2,000,000 acres, from short-term horizons of a few days or months to long-range planning over 150 or 200 years. Decision levels go from high-level management to operations on the ground. At first, large-scale monolithic models were proposed to solve global models. Given the difficulties in running and analyzing these models, several hierarchical decomposition approaches of global problems have been proposed to handle in a separate but linked way problems at different decision levels (Martell et al. 1998; Church 2007).

Consideration of fire effects in forests started in the 1980s (Martell 1982), where OR/MS was used in prevention of fires, fuel management, detection of

fires, resources acquisition, initial attack dispatching, extended attack management and training. In the last decade the concept of managing fires as system effort has become prevalent., with several systems in use in the US and Canada (Martell 2007).

Agriculture

Mathematical models have been proposed extensively to deal with agricultural problems (Hazell and Norton 1986). The main areas where quantitative approaches have been proposed are overviewed here.

Crop Production Problems at Farm Level. These include the determination of cropping patterns, planning of harvesting operations, design of harvesting equipment, and control of pests and diseases. These interrelated decisions include planting design, use of fertilizer, irrigation schemes and capital investment. The main techniques proposed to handle these problems are mainly linear programming and simulation, and also mixed integer programming, dynamic programming, and decision theory.

Uncertainty in crop yields and prices has also been introduced via portfolio theory, stochastic dominance, stochastic dynamic programming, and games against nature. Risk management considers how farmers perceive risk and act on it (Huirne et al. 2007). Another important issue is that of multiple criteria, relevant in most areas of agricultural decisions, handled mostly using goal programming and compromise programming (Romero and Rehman 1989; Hayashi 2007).

These proposed models have not been applied intensively. One main reason is the farming tradition of using judgment based on experience, rather than looking for technical optimality, which is also constrained by the lack of accurate information.

Regional Planning Problems. These are oriented toward centralized decisions such as the evaluation of development projects, determination of tax or price support policies, to analyze the trade-offs between economic returns and environmental impacts (Teague et al. 1995) or to determine and make operational the concept of sustainable agriculture (Pandey and Hardaker 1995). Spatial market equilibrium models serve for analysis of domestic or international trade. These approaches, however, have mostly had indirect influence on practice or are of research interest only (Campos et al. 2007).

Livestock Production. In this area, mathematical models have been widely and successfully used. In the classical diet and ration formulation problems, a variety of models, mostly linear programming and also quadratic programming, have been proposed for different animal stocks. Simulation has been used for modeling pasture-based livestock systems. The problem of livestock breeding and replacement has been approached through simulation, linear programming, deterministic and stochastic dynamic programming. Most applications are in the area of evaluation of replacement policies, particularly in large-scale dairy, egg production and poultry. Multi-objective considerations for diet problems include different nutrient requirements and costs (Peña et al. 2007).

Agriculture and the Environment. Since around 2000, environmental issues have become prevalent, as the negative aspects towards the environment due to agriculture became more evident. Crop simulation models as well as optimization models help quantify the environmental effects (such as soil erosion or pesticide use) of management practices. GIS systems have provided important support as decision systems integrated with OR models (Zekri and Boughanmi 2007).

Mining

Quantitative techniques have played a significant role in the mining industry (Lane 1988), accelerated since the 1980s. In particular, advances in computational power and OR software have resulted in an increased and successful use of mainly MIP models into mine planning (Newman et al. 2010). Mining is carried out as open pit or underground. In some cases lately, as in copper mines open pit and underground mining are integrated. Some major decision problems in mining are now described.

The optimal design of open-pit mines. The goal is to determine the feasibility of operations and the contours in mining extraction processes, where extraction is viewed as a series of nested blocks in three dimensions, so as to maximize the difference between sale value and extraction and processing costs within geological and mining restrictions.

Graph theory, linear programming, and heuristic methods have been used for this problem (Hochbaum and Chen 2000).

Optimization techniques have been used also for ore-body modeling and reserve estimations, optimal production schedules, capacity planning, machine scheduling, machine maintenance, production planning, and transportation. (Caccetta 2007).

Underground mine planning is more complex, as different extraction techniques are employed. The main problems approached are on how to plan the extraction of the mineral. MIP models have been successfully employed (Alford et al. 2007).

The introduction of large scale MIP models has led to increasingly consider the whole production chain, from mine extraction to plant processing (Caro et al. 2007).

Fishing

Fisheries Management present a different perspective from other natural resources in that since the resource is of free access, production is usually shared among different enterprises, and its allocation process is difficult and fuzzy. The long-term conservation of fish stocks is a high-priority issue, and a set of regulations to protect them has been developed worldwide. Fisheries systems are concerned mainly with two basic issues; biological analysis of fish stock behavior and the allocation and exploitation of the resources (Lane 1989; Bjorndal et al. 2004).

Biological issues include all aspects of population dynamics to understand how fish stock evolves (growth and mortality rates, reproductive properties) and fish stock assessment, given environmental impacts (pollution, warming or cooling trends), stock interactions, and exploitation.

The problem of resource allocation involves assigning and regulating fishing rights (quotas, licenses, capture taxes, area closures). Exploitation or management decisions include: fleet design and harvesting operations, determination of catching effort (the response of fishing captures to fishing effort provides important information for stock assessment), design of fish plants. (Arnason 2007).

Quantitative approaches have been widely proposed for all these problems: Descriptive mathematical modeling (in particular for the biological aspects), mathematical programming methods such as linear programming, nonlinear programming, optimal control and dynamic programming, statistical estimation and simulation. While the range of methodological proposals is wide, actual applications lag behind, mainly due to the lack of reliable data. Most applications are in the areas of exploitation and allocation. In this area there is also growing concern about explicit incorporation of uncertainty (Nostbakken and Conrad 2007) and multi-criteria decision making (Lane 2007).

See

- ▶ Agriculture and the Food Industry
- ▶ Environmental Systems Analysis
- ▶ Fuzzy Sets, Systems, and Applications
- ▶ Global Models
- ▶ Goal Programming
- ▶ Linear Programming
- ▶ Metaheuristics
- ▶ Multiobjective Programming
- ▶ Simulated Annealing
- ▶ Tabu Search

References

Alford, C., Brazil, M., & David, L. (2007). Optimisation in underground mining. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 595–609). New York: Springer.

Andalaft, N., Andalaft, P., Guignard, M., Magendzo, A., Wainer, A., & Weintraub, A. (2003). A problem of forest harvesting and road building solved through model strengthening and Lagrangean relaxation. *Operations Research, 51*(4), 613–628.

Arnason, R. (2007). Fisheries management. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 157–180). New York: Springer.

Barahona, F., Weintraub, A., & Epstein, R. (1992). Habitat dispersion in forest planning and the stable set problem. *Operations Research, 40*(1), S14–S21.

Bjorndal, T., Herrero, I., Newman, A., Romero, C., & Weintraub, A. (2010). Operations research in the natural resource industry, *International Transaction in Operational Research*, 1–24.

Bjorndal, T., Lane, D., & Weintraub, A. (2004). Operational research models and the management of fisheries and aquaculture: A review. *European Journal of Operational Research, 156*(3), 533–540.

Caccetta, L. (2007). Application of optimisation techniques in open pit mining. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 547–559). New York: Springer.

Campos, P., Caparros, A., Cerda, E., Huntsinger, L., & Standiford, R. (2007). Modeling multifunctional agroforestry systems with environmental values: Dehesa in Spain and Woodland Ranches in California. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 33–52). Springer: New York.

Carlsson, D., & Ronnqvist, M. (2005). Supply chain management in forestry, case studies at Sodra Cell AB. *European Journal of Operational Research, 163*, 589–616.

Caro, R., Epstein, R., Santibañez, P., & Weintraub, A. (2007). An integrated approach to the long-term planning process in the copper mining industry. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 595–609). Springer: New York.

Church, R. (2007). Tactical-level forest management models. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 343–363). Springer: New York.

D'Amours, S., Rönnqvist, M., & Weintraub, A. (2008). Using operational research for supply chain planning in the forest products industry. *Information System Operational Research, 46*(4), 265–281.

Diaz-Balteiro, L., & Romero, C. (2007). Multiple criteria decision-making in forest planning: Recent results and current challenges. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 473–488). Springer: New York.

Epstein, R., Karlsson, J., Ronnqvist, M., & Weintraub, A. (2007a). Harvest operational models in forestry. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 365–377). Springer: New York.

Epstein, R., Morales, R., Serón, J., & Weintraub, A. (1999). Use of OR systems in the Chilean forest industries. *Interfaces, 29*, 7–29.

Epstein, R., Ronnqvist, M., & Weintraub, A. (2007b). Forest transportation. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 391–403). Springer: New York.

Epstein, R., Weintraub, A., Sapunar, P., et al. (2006). A combinatorial heuristic approach for solving real-size machinery location and road design problems in forestry planning. *Operations Research, 54*(6), 1017–1027.

Goycoolea, M., Murray, A., Vielma, J. P., & Weintraub, A. (2009). Evaluating approaches for solving the area restriction model in harvest scheduling. *Forest Science, 55*(2), 149–165.

Gunn, E. (2007). Models for strategic forest management. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 371–341). Springer: New York.

Hayashi, K. (2007). Dealing with multiple objectives in agriculture. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 17–32). Springer: New York.

Hazell, P. B. R., & Norton, R. D. (1986). *Mathematical programming for economic analysis in agriculture*. New York: Macmillan.

Hochbaum, D. S., & Chen, A. (2000). Performance analysis and best implementations of old and new algorithms for the open-pit mining problem. *Operations Research, 48*(6), 894–914.

Hof, J. G., & Bevers, M. (1998). *Spatial optimization for managed ecosystems*. New York: Columbia University Press.

Hof, J., & Haight, R. (2007). Optimization of forest wildlife objectives. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 405–418). Springer: New York.

Huirne, R., Meissen, M., & Van Asseldonk, M. (2007). Importance of whole-farm risk management in agriculture. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 3–16). Springer: New York.

Kent, B., Bare, B. B., Field, R. C., & Bradley, G. A. (1991). Natural resource land management planning using large-scale linear programs: The USDA forest service experience with FORPLAN. *Operations Research, 39*, 13–27.

Kirby, M. W., Hager, W. A., & Wong, P. (1986). Simultaneous planning of wildland management and transportation alternatives. *TIMS Studies in the Management Science, 21*, 371–387.

Lane, K. F. (1988). *The economic definition of ore*. London: Mining Journal Books.

Lane, D. E. (1989). Operational research and fisheries management. *European Journal of Operational Research, 42*, 229–242.

Lane, D. E. (2007). Planning in fisheries related systems. Arnason, R. Fisheries management. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 237–271). New York: Springer.

Lohmander, P. (2007). Adaptive optimization of forest management in a stochastic world. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 525–543). Springer: New York.

Martell, D. (1982). A review of operational research studies in forest fire management. *Canadian Journal of Forest Research, 12*(2), 119–140.

Martell, D. (2007). Forest fire management. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 489–509). Springer: New York.

Martell, D., Gunn, E., & Weintraub, A. (1998). Forest management challenges for operational researchers. *European Journal of Operation Researcher, 104*(1), 1–17.

Matthews, S., O'Connor, R., & Plantiga, A. J. (2002). Quantifying the impacts on biodiversity of policies for carbon sequestration in forests. *Ecological Economics, 40*(1), 71–87.

Murray, A. (2007). Spatial environmental concerns. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 419–429). Springer: New York.

Murray, A., & Church, R. (1996). Analyzing cliques for imposing adjacency restrictions in forest models. *Forest Science, 42*, 166–175.
Navon, D. (1971). *Timber RAM a long range planning method for commercial timber lands under multiple-use management, USDA, Forest Service Research Paper PSW-70*. Berkeley, CA: USDA.
Newman, A., Rubio, E., Caro, R., Weintraub, A., & Eurek, K. (2010). A review of operations research in mine planning. *Interfaces, 40*(3), 222–245.
Nostbakken, L., & Conrad, J. (2007) Uncertainty in bioeconomic modelling. Arnason, R. Fisheries management. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 217–235). New York: Springer
Pandey, S. B., & Hardaker, J. B. (1995). The role of modelling in the quest for sustainable farming systems. *Agricultural Systems, 47*, 439–450.
Peña, T., Castrodeza, C., & Lara, P. (2007). Environmental criteria in pig diet formulation with multi-objective fractional programming. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 53–68). Springer: New York.
Romero, C., & Rehman, T. (1989). *Multiple criteria analysis for agricultural decisions*. Amsterdam: Elsevier.
Teague, M. L., Bernardo, D. J., & Mapp, H. (1995). Farm-level economic analysis incorporating stochastic environmental risk assessment. *American Journal of Agricultural Economics, 77*, 8–19.
USDA Forest Service. (1995). *Spectrum user's guide*. Ecosystem Management Analysis Centre.
Weintraub, A., Romero, C. Bjornald, T., & Epstein, R., (Eds.), (2007). *Handbook of operations research in natural resources*. New York: Springer.
Zekri, S., & Boughanmi, H. (2007). Modeling the interactions between agriculture and the environment. In A. Weintraub, C. Romero, T. Bjornald, & R. Epstein (Eds.), *Handbook of operations research in natural resources* (pp. 69–91). Springer: New York.

Near-Optimal Solution

For an optimization problem, a near-optimal solution is a feasible solution with an objective function value within a specified range from the (usually unknown) optimal objective function value.

Neighboring Extreme Point

In a convex set of solutions to a linear-programming problem, two extreme points are neighbors if they are connected by an edge of the convex set. The path of solutions determined by the simplex method is one that moves from one neighboring extreme point to another.

See

► Simplex Method (Algorithm)

Nested Partitions Method

A metaheuristic search approach for discrete optimization that employs sampling and iterative partitioning and recombining of the feasible solution space.

References

Shi, L., & Ólafsson, S. (2000). Nested partitions method for global optimization. *Operations Research, 48*, 390–407.
Shi, L., & Ólafsson, S. (2008). *Nested partitions optimization: Methodology and applications*. New York: Springer.

Network

A network is a pair of sets (N, A), where N is a set of nodes (points, vertices) and A is a set of arcs (edges, lines, links). If i and j are nodes, then the arc joining them is denoted by the ordered pair (i, j). An arc may have a cost c_{ij} that denotes the cost per unit flow across that arc, and an upper bound flow capacity denoted by u_{ij}. For some applications, a node may be a supply (source) node in which goods enter the network, a demand (sink) node in which goods leave the network, or a transshipment node through which goods are shipped without a gain or a loss. In most network applications, it is assumed that the flow of goods that enter a node is equal to the flow that leaves the node. This is the conservation of flow assumption. However, in some applications, the amount of goods that enter a node can be more than the amount that leaves the node (e.g., due to the expansion of a liquid) or can be less than the amount that leaves a node (e.g., due to a leak or pilferage).

These latter situations are termed networks with gains or losses. In most instances, network problems are special forms of linear-programming problems.

See

► Network Optimization

Network Design

A decision problem concerning the configuration (the nodes and links to be included/excluded) of a logistics network.

See

► Network Optimization
► Network Planning

Network Optimization

Thomas L. Magnanti
Massachusetts Institute of Technology,
Cambridge, MA, USA

Introduction

Networks are omnipresent in everyday life, e.g., highways, telephone lines, railways, electric power systems, airline route maps, computer and cable television networks. Networks also arise in other, perhaps less visible settings: manufacturing or distribution networks determine the flow of products through plants or between plants, warehouses, and retail outlets; and networks of interconnected components in integrated semiconductor chips and printed circuit boards provide electronic processing capabilities in thousands of commercial products.

In these settings, two sets of network optimization issues typically need to be addressed:

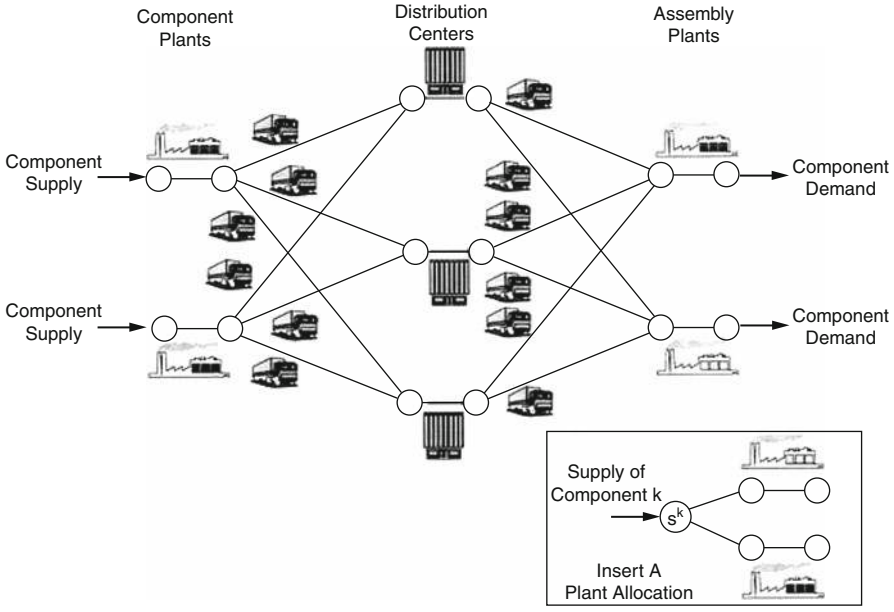
1. *Operational Planning* — How to use a given (distribution, telecommunication, or manufacturing) network as efficiently as possible? In this setting, the underlying network structure (topology and facilities) is known and the objective is to find the best way to route flow on it. For this reason, the set of optimization models for supporting these decisions have become known as network flow problems.
2. *System Design* — What is the best design of a network, one that will offer cost efficient and yet effective service to its users? In this setting, the objective is to simultaneously create the network structure and route flow on it. These models have become generally known as network design problems.

The OR/MS community has developed a rich array of network models and solution methods for operational planning and system design, applying these techniques in thousands of applications. Indeed, network optimization has served as one of the most active and fertile application, modeling, and theoretical domains within the fields of applied mathematics, computer science, engineering, and OR/MS.

Network Models

Figure 1 illustrates an application that contains the basic ingredients of network optimization. In this application context, which is typical of the automotive, computer and many other industries, a company produces product components in several plants/countries and assembles the products in other plants/countries. For convenience, any product component will be referred to as a commodity. One shipping option is to send all commodities directly from each component plant to each assembly plant. However, to achieve economies of scale, the firm uses a set of intermediate distribution centers (or warehouses). The distribution centers could also hold inventory and, thereby, permit the company to meet fluctuating demand requirements in the assembly plants.

To formulate this problem mathematically, first define an underlying network. In general, a network is (i) a set N of nodes, together with (ii) a set E of directed edges (i, j) that connect certain pairs i and j of the nodes. The application in Fig. 1 has a special network structure with one node corresponding to an



Network Optimization, Fig. 1 Network model of a production/distribution system

input and to an output for each plant and distribution center; the edges are of two types: those that connect plants and distribution centers and those that connect the input and the output node of each plant or distribution center. Each edge (i, j) has an associated per unit flow cost c_{ij}^k for each commodity $k = 1, 2, \ldots, K$, a flow capacity u_{ij}^k imposed upon commodity k, and flow capacity u_{ij} imposed upon the total flow of all commodities. For edges connecting plants and distribution centers, these quantities model the flow of commodities between the facilities. For edges joining the input and output nodes of each distribution center, these quantities model the throughput costs and capacities of the distribution center (similarly, for the plants). The use of two nodes to model the throughput at any node is a common modeling device for representing node costs and capacities as edges costs and capacities.

The arrows in Fig. 1 directed into the component plant input nodes specify the supplies of the commodities at the component plants. This model assumes that the production of each component in each component plant has already been determined. To use the network optimization model to allocate the production of each commodity among the component plants, one could introduce an additional component supply node s^k for each commodity k with the total supply of that commodity as the node's input. The flow on edges (s^k, q) connecting this node to the plants q would allocate the total supply of that component to the available plants (see Insert A in Fig. 1). The introduction of additional nodes and edges like this is another modeling device used frequently in practice.

To model a general network optimization problems (and thus various versions of the production and distribution planning problem), let f_{ij}^k denote the flow of commodity k from node i to node j (i.e., the flow on edge (i, j) in the direction i to j). Also let b_i^k denote the net supply of commodity k at node i; this quantity is positive at the input nodes of the network (component plants in the example), is negative (to model demand) at the output nodes of the network (the assembly plants in the example), and is zero at all the other nodes. The model has the following general form:

$$\text{minimize} \sum_{k=1}^{K} \sum_{(i,j)\in E} c_{ij}^k f_{ij}^k + \sum_{(i,j)\in E} F_{ij} y_{ij} \tag{1}$$

$$\text{subject to} \sum_{j:(i,j)\in E} f_{ij}^k - \sum_{j:(j,i)\in E} f_{ji}^k = b_i^k \quad \text{for all } i \in N \text{ and } k = 1, \ldots, K \tag{2}$$

$$\sum_{k=1}^{K} f_{ij}^k \le u_{ij} y_{ij} \text{ for all } (i,j) \in E \tag{3}$$

$$f_{ij}^k \le u_{ij}^k y_{ij} \text{ for all } (i,j) \in E \text{ and } k = 1, \ldots, K \quad (4)$$

$$f_{ij}^k \ge 0 \text{ for all} (i,j) \in E \text{ and } k = 1, \ldots, K \quad (5)$$

$$0 \le y_{ij} \le 1 \text{ and } y_{ij} \text{ integer for all } (i,j) \in E. \quad (6)$$

This model has the following interpretation. The flow conservation equation (2) for node i states the total flow out of that node minus the total flow into that node must equal the node's supply. The binary (0 or 1) decision variables y_{ij} model the network design decision, "should edge (i, j) be included in the network ($y_{ij} = 1$) or not ($y_{ij} = 0$)?" (In the application in Fig. 1, these variables model two types of decisions: (i) whether or not to locate a plant or a distribution center at one of the available locations and whether the network contains the corresponding throughput edge; and (ii) whether or not to use a particular transportation lane joining a plant to distribution center combination or distribution center to plant combination). The fixed cost F_{ij} associated with edge (i, j) is the cost for constructing/renting/ operating that edge (independent of its flow). The forcing constraints (3) and (4) force the flow on edges (i, j) for each commodity k to be zero if the network does not contain that edge ($y_{ij} = 0$). If $y_{ij} = 1$, constraint (3) states that the total flow on edge (i, j) cannot exceed the installed capacity u_{ij} of that edge, and constraint (4) states that the flow of commodity k on edge (i, j) cannot exceed the flow capacity u_{ij}^k for that commodity.

This model assumes that the edges are directed, i.e., flow on any edge goes in only one direction. But, in some applications, the edges will be undirected. To model these situations, impose the condition that $y_{ij} = y_{ji}$ and replace f_{ij}^k in constraints (3) and (4) with $f_{ij}^k + f_{ji}^k$, the total flow in both directions on edge (i, j).

The network optimization model (1)–(6) can be used to make facility location decisions (as in the example) and to make routing decisions (the choice of transportation lanes). Moreover, the model can be used in telecommunication and other applications to design a physical network, for example, to determine where to locate fiber optic cables in a telecommunication system.

The model (1)–(6) is a special type of mixed integer programming problem, i.e., it contains both continuous and integer/binary variables. In practice, solving this network optimization problem is a challenging (on the surface almost daunting) task. The model has a flow conservation equation for each node and commodity. Since networks with thousands of nodes arise frequently in practice, even for situations with a single commodity, the model often has thousands of equations. Many telecommunications and transportation applications require the flow of a commodity (message or freight) between every pair of nodes in the network. Therefore, with as few as one hundred nodes, the problem will have $100 * 99 = 9{,}900$ commodities and $9{,}900 * 100 \approx 1$ million flow conservation equations (one for each combination of commodity and node).

The problems become even more difficult when they have design variables. With as few as twenty nodes and a binary design variable y_{ij} for each of the $20(19)/2 = 190$ possible edges connecting these nodes, the model has 2,190 different design alternatives (since any design can include or exclude each of the 190 edges). This number is as large as the number of grains of sand needed to fill the solar system! Therefore, solving these problems requires considerable ingenuity. Since it is impossible to enumerate all possible solutions, the methods must consider them only implicitly.

Types of Models

The network optimization model (1)–(6) has many specializations and variants, each generating a considerable literature on its own (applications, solution methods, and underlying theory). Tables 1 and 2 show some of these models and indicates typical solution times for solving them (on a modern computer workstation).

The tables separate the models into two categories:

1. *Network flow models* — For these models, each binary variable y_{ij} is fixed at value 0 or 1 (and so the network topology is fixed) and the problem becomes a linear program. Notice that the problem has a very special structure since each flow variable f_{ij}^k appears in exactly two flow conservation equations, as an output of node i and an input of node j. Researchers have been able to use this feature to develop special purpose algorithms that solve the problems much more efficiently

Network Optimization, Table 1 Network flow models (each y_{ij} is fixed at value zero or one)

Model type	Problem description	Solution methods	Computational experience. Number of nodes: Solution time
Multicommodity flows	General flow model (1)–(6) with multiple commodities	Linear programming, decomposition methods	Hundreds: minutes
Minimum cost flows	Single commodity ($K = 1$)	Specialized path flow methods	Thousands: seconds
Maximum flows	Single commodity, no flow cost; send maximum flow between single source and destination node pair	Specialized node labeling (sequential search) methods	Thousands: seconds
Shortest paths	Single commodity, single origin, no flow capacities	Specialized node labeling (sequential search) methods	Tens of thousands: seconds

Network Optimization, Table 2 Network design models

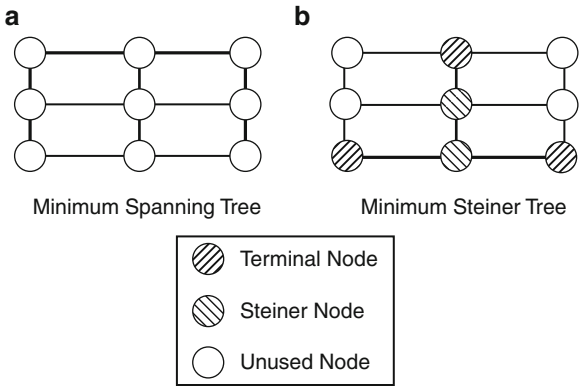
Model type	Problem description	Solution methods	Computational experience. Number of nodes: Solution time
Fixed cost network	General model (1)–(6)	Integer programming, heuristics	Tens: minutes or hours
Network loading	No flow costs, load network to meet required point-to-point demands (y_{ij} as integers, not binary)	Integer programming, heuristics	Tens: minutes or hours
Network connectivity	Find prescribed number of edge disjoint paths between various node pairs	Integer programming, heuristics, and linear programming dual ascent methods	Hundreds: minutes
Network synthesis	Given flow requirements, determine capacities on edges (at minimum cost) so that the network has the capability to meet prescribed demands between various node pairs	Minimum spanning tree if capacity on every edge costs the same	Thousands: seconds if all capacities have same cost
			Tens to hundreds: minutes (in general)
Minimum spanning trees	No flow costs, no capacities; find a network that connects all nodes	Specialized one-pass greedy algorithms	Thousands: seconds
Steiner trees	No flow costs, no capacities; find a network that connects prescribed set of nodes (and possibly others)	Heuristics and linear programming dual ascent methods	Thousands: seconds

For all network design problems, except the minimum spanning tree problem, the methods generally produce approximately optimal, not globally optimal solutions

than solving them using general purpose linear-programming software.

2. *Network design problems* –In these models, both design decisions and flow decisions are relevant. For some of these models, the flow costs are zero and the problems become that of finding a least cost network configuration that meets the required flow requirements.

Figure 2 gives examples of the minimum spanning tree and Steiner tree problems, assuming that the cost of each edge is proportional to its length. The underlying network in these examples is typical of those in printer circuit board applications that have East-west and North-south channels for making wiring connections (therefore, all the edges are in a rectangular pattern). Note that the minimum spanning tree needs to connect all the nodes and the Steiner tree needs to connect only a subset of the nodes (so called terminal nodes), but can optionally use some of the other nodes (so called Steiner nodes). In both cases, the goal is to find the least cost network configuration, as measured by the total cost of the



Network Optimization, Fig. 2 Minimum spanning tree and Steiner tree problems

chosen edges (flow costs are irrelevant). The other network design problems also want to find optimal network configurations, but might include flow costs as well.

The discussion has shown that the network optimization problem (1)–(6) has a wide range of applications and has shown how to introduce additional nodes and edges to enhance the model's ability to capture varied application features (e.g., to allocate production at the component plants or to represent the throughput of a node). Many other such modeling techniques have proven to be useful in practice. As indicated in Table 1, computer software is available for solving large-scale network flow problems very quickly. Except for spanning tree and Steiner trees, capabilities for solving network design problems is much more limited.

Solution Methods

Solving network optimization problems requires considerable ingenuity, in developing solution methods, implementing them efficiently on a computer, and analyzing them to determine their efficiency (in theory or practice). To illustrate these issues, consider one of the easiest network flow problems, the shortest path problem. After describing a basic algorithm for solving this problem, it will be shown how to organize the computations to implement the algorithm more efficiently and then how to improve on it even further when more is known about the underlying data (the edge lengths).

Suppose one is given a network with nonnegative lengths d_{ij} on the edges (i, j) and the goal is to find the shortest path between two designated nodes, a source node s and a terminal node t. To solve the problem, one could use the following algorithm (solution method). If node j is the node closest to the source node, then the shortest path distance $d(j)$ from the source node to this node is the direct path on the edge (s, j) whose distance is d_{sj}. Next consider the node k that is closest to node s either along the direct edge (s, k) or on the shortest path through node j, i.e., with the distance $d_{sj} + d_{jk} = d(j) + d_{jk}$. To choose the best of these two alternatives, compute $d(r) = \min\{d_{sr}, d(j) + d_{jr}\}$ for each node $r \neq s$ or j and select as the next node k, a node r with the smallest value of $d(r)$. It is easy to see that this choice gives the shortest distance along any path from node s to node k. In general, suppose that after several steps, the shortest path distances $d(j), d(k), \ldots, d(p)$ from the source node s to each of the nodes $j, k, \ldots, p$ have been found. Then to find the shortest path distance to the next node q, for all nodes $r \neq s, j, k, \ldots, p$, compute

$$d(r) = \min\{d_{sr}, d(j) + d_{jr}, d(k) + d_{kr}, \ldots, d(p) + d_{pr}\}. \tag{7}$$

Choose node q to be a node r with the smallest of the values $d(r)$. Once node t has been chosen in any of these steps, the problem is solved, i.e., the shortest path distance from node s to node t has been found. (See Ahuja et al. 1993, for a proof.)

This algorithm computes the shortest path distance to one more node at each step. If the network contains a total of n nodes and the shortest path distance to v of them have been found, then the computation (7) requires v additions and comparisons for each of $n - v$ nodes and so $v(n - v)$ computations. Therefore, to find the shortest path distance to all nodes, the algorithm requires $1(n-1) + 2(n-2) + 3(n-3) + \cdots + (n-1)(1) = n^2(n-1)/6$ computations. Can this be improved? Yes, by noticing that this algorithm performs many redundant computations. For example, after the first step, for each node r not yet chosen in one of the previous steps, the algorithm computes the quantity $d(j) + d_{jr}$. Note that after node q has been chosen, the computation (7) becomes

$$d^{\text{new}}(r) = \min\{d_{sr}, d(j) + d_{jr}, d(k) + d_{kr}, \ldots, d(p) + d_{pr}, d(q) + d_{qr}\}. \tag{8}$$

Comparing (7) and (8) shows that $d^{\text{new}}(r) \leftarrow \min\{d(r), d(q) + d_{qr}\}$. Therefore, if the values of $d(r)$ from one step to another are stored and one carries out the computation

$$d(r) \leftarrow \min\{d(r), d(q) + d_{qr}\}, \tag{9}$$

then at step v when there are $n - v$ nodes that are candidates to be chosen next, there are $v - r$ computations, and so overall the algorithm now requires only $(n-1) + (n-2) + \cdots + 1 = n(n-1)/2$ computations. As this simple example shows, by organizing the computations intelligently, one can often considerably reduce the computational requirements of an algorithm. Much of the literature of network flow algorithms involves the use of similar ideas for designing and analyzing algorithms (though, in general, the ideas are much more complex).

To illustrate further how researchers have used problem structure to design efficient algorithms, suppose that the cost structure for the shortest path problem were even simpler, such that each distance d_{ij} is limited to the values 1 or 2. Notice that in this case, the shortest path distance from the source node s to any node k is one of the integers 1, 2, …, $2(n-1)$. To obtain an improved algorithm, use this fact and implement the computations (9) in a more streamlined fashion by maintaining a collection of $2(n-1)$ buckets, storing all the nodes r whose distance $d(r)$ is k in the kth bucket. Then choose the buckets one at a time from smallest to largest, starting with bucket number 0. If the bucket is nonempty, select a node q from it and then any edge (q, r) incident to node r. Then use the expression (9) to update the distance $d(r)$ of node r and if the distance of node r increases, move it to a new bucket. Note that this algorithm considers each edge (i, j) only once and must search at most $2(n-1)$ buckets to see if they are empty or not and to extract their contents. Therefore, for a network with m edges, this implementation of the algorithm requires $m + 2(n-1)$ computations (assuming one can effectively transfer nodes between buckets, which is easy to do). Since m is often far less than its maximum possible value of n^2, this algorithm is typically much faster than the implementation embodied by the previous implementation of the computations (8). When the edge lengths are limited within some range $0 \leq d_{ij} \leq C$ for some constant C, a similar type of bucket implementation can be very efficient and produces some of the most effective algorithms for solving shortest path problems.

The design and analysis of network optimization algorithms has an enormous literature. This brief introduction to the topic has illustrated several important aspects of this field:

- Network algorithms often use simple computations, such as those invoked in expressions (7) and (9), for solving problems rather than the more sophisticated methods needed to solve other optimization problems such as general linear programs. Indeed, software based upon specialized methods like these are able to solve shortest path problems with thousands of nodes in just a few seconds of computational time (Table 1), even though these problems are linear programs with thousands of constraints (one conservation equation in the model (1)–(6) for each node).
- In solving a particular problem, it is often just as efficient to solve a broader class of problems (the algorithm described here finds the shortest paths from the source node to all other nodes, not just the terminal node).
- Organizing computations carefully can improve an algorithm's efficiency. In the shortest path example, the number of computations were reduced from $n^2(n-1)/6$ to $n(n-1)/2$ by merely avoiding redundant computations (see (8) and (9)).
- The creative use of data structures (buckets in the example) often leads to more efficient algorithms [$m + 2(n-1)$ instead of $n(n-1)/2$ computations in the example].
- Often one can design algorithms to exploit the nature of the data and not just the type of problem being solved. Two illustrations of this: (i) the algorithm described in the example here might not solve a shortest path problem when some the edge lengths are negative, so it has exploited the fact that all the edge lengths are nonnegative; and (ii) when the data have restricted ranges (e.g., the edge costs are between 0 and C in the example), one can frequently can devise more efficient algorithms.

- The shortest path algorithm described here is polynomial in the sense that the number of computations is a polynomial function of the number of nodes n and the number of edges m in the underlying network. One of the great challenges in network optimization is to discover polynomial algorithms with the lowest possible degree (or fastest running time). For most design problems, the research community has not been able to find polynomial algorithms (the minimal spanning tree problem is a notable exception). Indeed, most design problems are, in the parlance of computer science, *NP*-complete, implying that it is quite unlikely that a polynomial time algorithm exists. Nevertheless, the community has been able to design algorithms that are efficient in practice for many of these problems.

Further Readings

Several books amplify on the topics described here and introduce many other applied and theoretical aspects of network optimization. Ford and Fulkerson (1962) provide a seminal account of early developments in this field. Ahuja et al. (1993) offer a modern treatment of this subject covering both theory and applications. Glover et al. (1992) provide valuable insight into network modeling and applications. The handbooks edited by Ball et al. (1995a, b) contain comprehensive reviews by many leading researchers in network optimization. Lawler (1976) draws valuable connections between network flows and a related topic in combinatorial optimization known as matroids.

See

- ▶ Facility Location
- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Location Analysis
- ▶ Maximum-Flow Network Problem
- ▶ Minimum-Cost Network-Flow Problem
- ▶ Multicommodity Network-Flow Problem
- ▶ Shortest Path Problem
- ▶ Steiner Tree Problem

References

Ahuja, R. K., Magnanti, T. L., & Orlin, J. B. (1993). *Network flows: Theory, algorithms and applications*. Englewood Cliffs: Prentice Hall.

Ball, M., Magnanti, T. L., Monma, C., & Nemhauser, G. L. (1995a). *Network models. Vol. 7: Handbooks in operations research and management science*. New York: Elsevier.

Ball, M., Magnanti, T. L., Monma, C., & Nemhauser, G. L. (1995b). *Network routing. Vol. 8: Handbooks in operations research and management science*. New York: Elsevier.

Ford, L. R., & Fulkerson, D. R. (1962). *Flows in networks*. New Jersey: Princeton University Press.

Glover, F., Klingman, D., & Phillips, N. (1992). *Network models in optimization and their applications in practice*. New York: Wiley.

Lawler, E. L. (1976). *Combinatorial optimization: Networks and matroids*. New York: Holt, Rinehart and Winston.

Network Planning

Willy S. Herroelen[1] and Graham K. Rand[2]
[1]Katholieke Universiteit Leuven, Leuven, Belgium
[2]Lancaster University, Lancaster, UK

Introduction

Network planning is a generic name for methods that study projects as a set of interconnected activities with the purpose of assisting in planning, scheduling and controlling projects. These methods are based on models describing projects as activity networks and include well-known techniques such as the Critical Path Method (CPM) and the Program Evaluation and Review Technique (PERT). CPM determines the critical path that includes the so-called critical activities (activities deserving maximal attention as any delay causes a delay of the project's completion date), whereas PERT estimates the probability distribution of the project's completion date. Essentially, network planning involves a planning phase, a scheduling phase and a project control phase. The planning process involves the identification of the project activities, the estimation of time and resources, the identification of the precedence relationship between the activities and the identification of the schedule and resource constraints.

Project scheduling involves the construction of a project base plan that specifies for each activity the precedence and resource feasible start and completion dates, the amounts of the various resource types that will be needed during each time period, and as a result the project budget. Once the project starts, the project must be monitored and controlled. Project control involves the difficult task of measuring actual progress and comparing it to planned progress. If this comparison reveals that the project is likely to run behind schedule, to overrun the budget, or to violate the original technical specification, corrective action must be taken to get the project back on track.

History

The need to improve planning techniques to help control major projects was recognized in the 1950s. CPM arose from a jointly sponsored venture of E.I. du Pont de Nemours and Company and the Sperry-Rand Corporation. By September 1957, an actual application was conducted on a pilot system using the UNIVAC I computer, and from this CPM evolved (Kelley and Walker 1959; Kelley 1961). At the same time, the U.S. Navy was developing a system to plan and coordinate the Polaris missile program. From this, PERT evolved, and was credited with helping to advance Polaris by at least two years (Malcolm et al. 1959). Further details of these early developments can be found in Moder et al. (1983) and Sculli (1989).

Since these early days, variations of these methods have been developed (VERT, GERT, SCERT), and continuing research has led to the development of new models and techniques allowing for more adequate procedures to deal with general types of precedence and resource constraints, new planning objectives, and better ways to cope with uncertainty. Excellent texts and surveys include Elmaghraby (1977); Slowinski and Węglarz (1989); Özdamar and Ulusoy (1995); Herroelen et al. (1998), Brucker et al. (1999); Klein (2000); Kolisch and Padman (2001), Demeulemeester and Herroelen (2002), Dorndorf (2002), Neumann et al. (2003); Herroelen and Leus (2004); Herroelen (2005); Schwindt (2005); Jozefowska and Węglarz (2006); Artigues et al. (2008); Hartmann and Briskorn (2010); Węglarz et al. (2011).

Planning

Construction of the project network — There are two possible modes of representation of a project network: the activity-on-arc representation (AoA), which uses a set of arcs to represent the activities and a set of nodes to represent events, and the mostly used activity-on-node representation (AoN). The AoN representation (Fondahl 1961; Roy 1964) allows for the representation of various types of precedence relations: finish-start precedence relations with zero time-lag (used in PERT and CPM), start-start, finish-start, start-finish and finish-finish relations with minimal and maximal time-lags. A minimal time-lag specifies that an activity can only start (finish) when its predecessor has already started (finished) for a certain time period, whereas a maximal time-lag specifies that an activity should be started (finished) at the latest a number of time periods beyond the start (finish) of another activity.

Time estimates — Deterministic project planning models assume that activity durations can be estimated with certainty, typically as a single-time estimate. It should be well understood that the use of a single duration estimate assumes an implicit choice of a particular execution mode for the activity corresponding with a particular resource allocation (single mode). Instead of working with single-time estimates, several possible execution scenarios (multiple execution modes) may be defined, each mode reflecting a feasible way to combine an activity duration and a resource allocation.

Dealing with uncertainty – The PERT approach — The originators of PERT proposed a stochastic approach to cope with probabilistic activity durations. The assumption made by PERT is that activity durations are beta-distributed. Three activity duration estimates are used: an optimistic estimate (t_o), an estimate of the most likely duration (t_m), and a pessimistic estimate (t_p). An approximation to the expected time can be found by taking a weighted average of the three estimates (in the ratio 1:4:1), $(t_o + 4t_m + t_p)/6$, which has a standard deviation $(t_p - t_o)/6$.

Resource estimates — Project activities require resources for their execution. Different resource categories have been defined in the literature (Węglarz et al. 2011), the most common ones being

renewable and non-renewable resources. Renewable resources (manpower, machines, equipment, tools) are available on a per-period basis; non-renewable resources (money, raw materials, energy) are available on a total project basis.

Generating a Feasible Baseline Schedule

The fundamental objective of network planning is the creation of a precedence and resource-feasible baseline schedule that establishes the planned start and finish times of the individual activities, meeting as much as possible the objectives set forward by project management. One of the most common time-based objectives is the minimization of the planned project duration.

In many real-world situations the time-oriented objectives may be replaced by resource-based or financial objectives. An important example of a resource-based objective occurs in the resource-availability cost problem, where the capacities of the renewable resources are to be determined such that a given project deadline is met and the resource availability costs are to be minimized. Another example is the resource leveling problem, which involves the generation of a time-feasible schedule for which the resource profiles for the various resources are as level as possible, without violating the project deadline. An important financial objective is related to the incoming and outgoing cash flows that are generated during the execution of a project. This results in models that aim at maximizing the net present value of a project. It should be clear that the construction of a baseline schedule should eventually be done under multiple objectives, using a multi-objective or multi-criteria approach.

Critical path analysis — Using the activity-duration estimates, and taking into account the precedence relations (PERT and CPM assume finish-start relations with zero time-lag), the longest path in the project network can be computed. This is the so-called critical path, which determines the planned project duration.

The PERT approach has been widely criticized on theoretical grounds (Elmaghraby 1977; Golenko-Ginzburg 1989; Sculli 1989). In the PERT system, the mean value for the project duration is taken as the sum of the mean values of the durations of the activities on the critical path. This assumption is only correct for a project that consists of a single chain of activities, but progressively underestimates the mean project duration as the complexity of the network increases. In the PERT system, the variance for the project duration is taken as the sum of the variances of the activities on the critical path. Again, this assumption is only correct for a project that consists of a single chain of (independent) activities, but progressively underestimates the variance as the complexity of the network increases. There have been serious doubts expressed as to the appropriateness of the beta distribution. The PERT approach does not take into account the probability of completion of sub-critical paths. This inability to take account of sub-critical paths is the most serious criticism of PERT. Because of this PERT typically underestimates the true statistical project mean duration and also seriously underestimates the probability of meeting a deadline.

Resource-constrained scheduling — The introduction of renewable resources into the analysis complicates matters considerably. Computing a precedence and resource-feasible deterministic schedule that minimizes the project duration, the infamous resource-constrained project scheduling problem (RCPSP), is *NP*-hard in the strong sense. Both exact and suboptimal procedures have been presented in the literature (Özdamar and Ulusoy 1995; Herroelen et al. 1998; Brucker et al. 1999; Kolisch and Padman 2001).

Exact procedures for solving the RCPSP typically rely on branch-and-bound (see e.g. Demeulemeester and Herroelen 1992, 1997). At the time of writing, the best exact results have been reported by so-called hybrid approaches, combining for example constraint programming and satisfiability testing. Constraint programming techniques learn about logical implications between variable settings, which are used to strengthen the bounds on variables. Satisfiability testing draws from unsatisfiable or conflicting structures, which helps to quickly find reasons for and excluding infeasible parts of the search space (Schutt et al. 2009; Berthold et al. 2010).

The complexity of the RCPSP has motivated numerous research efforts on the design of heuristic scheduling procedures (Hartmann and Kolisch (2000) and Kolisch and Hartmann (2006)), which have demonstrated that the best results are obtained by hybrid metaheuristics, yielding a 25–30% deviation from the critical path-based lower bound.

Robust project scheduling — The probability of a precomputed baseline schedule being executed exactly as planned is low: activities may take more or less time than originally anticipated, resources may become unavailable, material may arrive behind schedule, new activities may have to be incorporated or activities may have to be dropped due to changes in the project scope, ready times and due dates may be modified, etc.

The aim of proactive project baseline scheduling is to generate stable baseline schedules that are protected against the disruptions that may occur during project execution. Most commonly this is achieved by inserting buffers in the baseline schedule (Herroelen and Leus 2004).

The critical chain scheduling methodology, introduced by Goldratt (1997), uses aggressive duration estimates and computes the critical chain in the generated precedence and resource feasible baseline schedule. The critical chain is the chain of precedence and resource dependent activities that determines the overall duration of the project. The safety in the durations of the activities that are on the critical chain, which was cut away by selecting aggressive duration estimates, is shifted to the end of the critical chain in the form of a project buffer to protect the project due date against variability in the critical chain activities. Feeding buffers are inserted whenever a non-critical chain activity joins the critical chain. The working principles of critical chain have been evaluated by Herroelen and Leus (2001) and Herroelen et al. (2002).

Research efforts on the development of reliable proactive scheduling procedures include Herroelen and Leus (2004), Herroelen (2007).

Project Control

Once the project has started, control is maintained by a system of status reporting. Some activities will take longer than estimated and some shorter. Sometimes estimates for activities not yet completed require revision. At regular intervals the network must be updated, reanalyzed and new schedules prepared, taking into account all this new information.

The corrective actions needed when the schedule lags behind or the built-in protection breaks may involve activity crashing, giving rise to the so-called time/cost trade-off problems. The importance of time/cost trade-offs was recognized from the very start of CPM, when the developers of CPM recognized that the majority of activities encountered in real-life project settings can be performed in shorter or longer durations by increasing or decreasing the resources available to them. Most often the acceleration in the execution of activities comes at a cost.

It is usually assumed that the cost/duration relationship is linear between a normal and a crash duration, and that any intervening duration may be attained. The objective is, by selection of activity durations and their corresponding costs, to minimize total activity costs for a given project duration. A related problem is concerned with detecting the shortest project duration available within a given budget. The problem may be formulated as a linear program and solved by the simplex method, but a more efficient network flow algorithm was developed by Fulkerson (1961). Other approaches are described by Ritchie (1985). In most practical cases, resources are available in discrete units, such as number of machines, number of workers, etc. The resulting discrete time/cost trade-off problem is a hard nut to crack (De et al. 1995).

Computer Software Packages

A wide range of commercial project planning software packages is available on the market (Wasil and Assad 1988; De Wit and Herroelen 1990): among them Microsoft Office Project is probably the best known. The software relies on simple priority rules for resolving resource conflicts. As far as can be determined (the scheduling methodology incorporated in commercial software is generally proprietary and hence unavailable), the software computes the earliest start schedule and checks for resource overloads, which are resolved by delaying some of the involved activities. In some of the packages, the user may select one of the alternative priority rules (Maroto and Tormos (1994).

See

- ▶ Critical Path Method (CPM)
- ▶ Gantt Charts
- ▶ GERT

- ▶ Heuristics
- ▶ Metaheuristics
- ▶ Network
- ▶ Program Evaluation and Review Technique (PERT)
- ▶ Project Management
- ▶ SCERT
- ▶ Scheduling and Sequencing
- ▶ Theory of Constraints
- ▶ VERT

References

Artigues, C., Demassey, S., & Néron, E. (Eds.). (2008). *Resource-constrained project scheduling – models, algorithms, extensions and applications*. Hoboken, NJ: Wiley.

Berthold, T., Heinz, S., Lübbecke, M.E., Möhring, R., & Schulz, J. (2010). "A constraint integer programming approach for resource-constrained project scheduling," In Andrea Lodi, Michela Milano, Paolo Toth (Eds.), *Integration of AI and OR techniques in constraint programming for combinatorial optimization problems*, 7th International Conference, CPAIOR 2010, Bologna, Italy, June 14–18, 2010. Proceedings of the Lecture Notes in Computer Science 6140 Springer pp. 313-317.

Brucker, P., Drexl, A., Möhring, R., Neumann, K., & Pesch, E. (1999). Resource-constrained project scheduling: Notation, classification, models and methods. *European Journal of Operational Research, 112*, 3–41.

De Wit, J., & Herroelen, W. (1990). An evaluation of microcomputer-based software packages for project management. *European Journal of Operational Research, 49*, 102–139.

De, P., Dunne, E. J., Gosh, J. B., & Wells, C. E. (1995). The discrete time/cost trade-off problem revisited. *European Journal of Operational Research, 81*, 225–238.

Demeulemeester, E., & Herroelen, W. (1992). A branch-and-bound procedure for the multiple resource-constrained project scheduling problem. *Management Science, 38*, 1803–1818.

Demeulemeester, E., & Herroelen, W. (1997). New benchmark results for the resource-constrained project scheduling problem. *Management Science, 43*, 1485–1492.

Demeulemeester, E. L., & Herroelen, W. S. (2002). *Project scheduling – a research handbook*. New York: Springer-Verlag.

Dorndorf, U. (2002). *Project scheduling with time windows – from theory to applications*. Heidelberg: Physica-Verlag.

Elmaghraby, S. E. (1977). *Activity networks: Project planning and control by network models*. New York: Wiley.

Fondahl, J. W. (1961). A noncomputer approach to the Critical Path Method for the construction industry, Dept. of Civil Engineering, Stanford University, Stanford, California.

Fulkerson, D. R. (1961). A network flow computation for project cost curves. *Management Science, 7*, 167–178.

Goldratt, E. M. (1997). *Critical chain*. Great Barrington, MA: The North River Press.

Golenko-Ginzburg, D. (1989). PERT assumptions revisited. *Omega, 17*, 393–396.

Hartmann, S., & Briskorn, D. (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research, 207*(1), 1–14.

Hartmann, S., & Kolisch, R. (2000). Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem. *European Journal of Operational Research, 127*, 394–407.

Herroelen, W. (2005). Project scheduling – theory and practice. *Production and Operations Management, 14*, 413–432.

Herroelen, W. (2007). *"Generating robust baseline schedules", INFORMS tutorials in operations research 2007*. Hanover, MD: INFORMS.

Herroelen, W., & Leus, R. (2001). On the merits and pitfalls of critical chain scheduling. *Journal of Operations Management, 19*, 557–577.

Herroelen, W., & Leus, R. (2004). Robust and reactive project scheduling: A review and classification of procedures. *International Journal of Production Research, 42*(8), 1599–1620.

Herroelen, W., De Reyck, B., & Demeulemeester, E. (1998). Resource-constrained project scheduling – a survey of recent developments. *Computers and Operations Research, 25*(4), 279–302.

Herroelen, W., Leus, R., & Demeulemeester, E. (2002). Critical chain scheduling: Do not oversimplify. *Project Management Journal, 33*(4), 48–60.

Jozefowska, J., & Węglarz, J. (Eds.). (2006). *Perspectives in modern project scheduling*. Berlin: Springer-Verlag.

Kelley, J. E. (1961). Critical-path planning and scheduling: Mathematical basis. *Operations Research, 9*, 296–320.

Kelley, J. E. & Walker, M. R., (eds.), (1959). "Critical path planning and scheduling," In *Proceedings of Eastern Joint Computer Conference*, Boston, December 1–3, 1959, 160–173.

Klein, R. (2000). *Scheduling of resource-constrained projects*. Boston: Kluwer Academic Publishers.

Kolisch, R., & Hartmann, S. (2006). Experimental, investigation of heuristics for resource-constrained project scheduling – an update. *European Journal of Operational Research, 174*, 23–37.

Kolisch, R., & Padman, R. (2001). An integrated survey of deterministic project scheduling. *Omega, 29*, 249–272.

Malcolm, D. G., Roseboom, J. H., Clark, C. E., & Fazar, W. (1959). Application of a technique for research and development program evaluation. *Operations Research, 7*, 646–669.

Maroto, C., & Tormos, P. (1994). Project management: An evaluation of software quality. *International Transactions in Operational Research, 1*, 209–221.

Moder, J. J., Phillips, C. R., & Davis, E. W. (1983). Project management with CPM, PERT and Precedence Diagramming, Van Nostrand, New York.

Neumann, K., Schwindt, C., & Zimmermann, J. (2003). *Project scheduling with time windows and scarce resources*. Berlin: Springer-Verlag.

Özdamar, L., & Ulusoy, G. (1995). A survey on the resource-constrained project scheduling problem. *IIE Transactions, 27*, 574–586.
Ritchie, E. (1985). Network based planning techniques: A critical review of published developments. In G. K. Rand & R. W. Eglese (Eds.), *Further developments in operational research* (pp. 34–56). Oxford: Pergamon.
Roy, B. (1964). Contribution de la théorie des graphes a l'étude des problems d'ordonnancement. In B. Roy (Ed.), *Les problems d'ordonnancement: Applications et methodes* (pp. 109–125). Paris: Dunod.
Schutt, A., Feydy, T., Stuckey, P.J., & Wallace, M. (2009). Why cumulative decomposition is not as bad as it sounds. In I. Gent, (Ed.), *Proceedings of the 15th International Conference on Principles and Practice of Constraint Programming*, volume 5732 of LNCS, (pp. 746–761). Berlin: Springer-Verlag.
Schwindt, C. (2005). *Resource allocation in project management*. Berlin: Springer-Verlag.
Sculli, D. (1989). A historical note on PERT times. *Omega, 17*, 195–196.
Slowinski, R., & Węglarz, J. (1989). *Advances in project scheduling, studies in production and engineering economics*. Amsterdam: North Holland.
Trautmann, N. & Baumann, P. (2010). "Resource allocation capabilities of software packages for project management," *European Journal of Operational Research*, to appear.
Wasil, E. A., & Assad, A. A. (1988). Project management on the PC: Software, applications, and trends. *Interfaces, 18*(2), 75–84.
Węglarz, J., Józefowska, J., Mika, M., & Walligóra, G. (2011). Project scheduling with finite or infinite number of activity processing modes – a survey. *European Journal of Operational Research, 208*, 177–205.

Network Simplex Algorithm

For the minimum-cost network-flow problem, a special adaptation of the simplex method that takes advantage of the mathematical structure of the network constraints to produce a computationally fast and efficient solution algorithm. The main idea behind this algorithm is the recognition that a basic feasible solution to the network problem, treated as a linear-programming problem, corresponds to a spanning tree of the defining network.

See

► Minimum-Cost Network-Flow Problem
► Network Optimization
► Simplex Method (Algorithm)

Networks of Queues

Richard F. Serfozo
Georgia Institute of Technology, Atlanta, GA, USA

Introduction

Many important operational issues in communication and manufacturing systems concern random movements of discrete units called customers in networks of service stations with queueing. Examples of such queueing networks are:

- Computer and telecommunications networks — data packets, read/write transactions, files, or telephone calls move among computers, buffers, operators or switching stations;
- Manufacturing networks — parts, orders, or material move among workstations, inspection points, automated guided vehicles or storage areas;
- Equipment maintenance networks — parts or subsystems move among usage sites and repair facilities;
- Logistics and supply-chain networks — parts, material, personnel, trucks or equipment move among sources, storage depots and production facilities; and
- Parallel simulation and distributed processing systems — messages, data packets and signals move among buffers and processors.

Other areas in which queueing networks arise include biology (movements of animals, fish or diseases) and economics (movements of labor, people, capital or shopping centers).

Common questions about a queueing network are as follows. Where are its bottlenecks or major delays? How does one network design compare with another? What is a good set of rules for operating the network (e.g., customer priorities or routings)? What is a least-cost network (e.g., numbers of machines, tools or workers)? Typical aims or performance objectives include the following: the probability of a busy signal in a telecommunications network should be less than 1%; the expected waiting times in a computer system should be less than certain values; the probability of meeting manufacturing deadlines should be above 90%.

To address such issues requires an understanding of the behavior of the network in terms of the equilibrium (or stationary) probability distribution of the numbers of customers at the queueing stations, which will usually be referred to as the nodes of the network. These distributions are used to evaluate a variety of performance measures such as throughputs, expected costs, and percentage of time a station is overloaded. The equilibrium distribution is a basic ingredient for constructing objective functions or constraints used in mathematical programming algorithms to select optimal network designs and protocols. The quality of a network is also determined by the duration of travel and sojourn times in it, such as the time for a customer to travel from one sector to another or the amount of time it takes for a customer to visit a certain set of nodes. Equilibrium distributions are used in describing the means or distributions of such travel times.

Many queueing network models have been developed using the theories of multi-dimensional Markov processes, point processes, and stationary processes. By their nature, stochastic networks have myriad dependencies that are analytically intractable. There are several types of network models, however, that have closed-form equilibrium distributions, some of which are described in the following sections, with pointers provided to others.

Jackson Networks

Consider an m-node network, where discrete units called customers move among the nodes where they are processed or served. The evolution of the network is represented by a continuous-time Markov process $\{X_t : t \geq 0\}$ whose states are vectors $\boldsymbol{x} = (x_1,\ldots, x_m)$, where x_j denotes the number of customers at node j (in service or waiting in queue). The network is a closed Jackson network if it contains a fixed number of customers that move as follows. Whenever there are x_j customers at node j, the time to the next departure from that node is exponentially distributed with rate $\phi_j(x_j)$, independent of the rest of the network. A standard service rate is $\phi_j(x_j) = \mu_j \max\{x_j, s_j\}$, which represents s_j independent servers whose service times are exponentially distributed with mean $1/\mu_j$. When a customer departs from node j, it moves immediately to node k with probability p_{jk}. In other words, the sequence of states that each customer visits forms a discrete-time Markov chain with transition probabilities $\{p_{jk}\}$. Without loss of generality, assume this routing chain is irreducible.

Under these assumptions, whenever the network process $\boldsymbol{X}$ is in state $\boldsymbol{x}$, a transition is triggered by a customer moving from some node j to another node k, and the time to such a transition is exponentially distributed with rate $\phi_j(x_j)p_{jk}$. Thus, $\boldsymbol{X}$ is a Markov process. It is positive recurrent because its state space is finite and the routing chain is irreducible. Its equilibrium distribution is

$$\pi(\boldsymbol{x}) = c \prod_{j=1}^{m} \alpha_j^{x_j} \prod_{n=1}^{x_j} \phi_j(n)^{-1}, \tag{1}$$

where $\{\alpha_j\}$ is the stationary distribution of the routing probabilities $\{p_{jk}\}$. The factor c is the normalization constant under which these terms sum to 1, and $\prod_{n=1}^{k} a_n = 1$ when $k = 0$.

Next, consider an open version of the network in which customers enter certain nodes in the network from outside (called node 0) according to independent Poisson processes, where λ_{0j} is the arrival rate for an entry node j. Then the probability that an arbitrary arrival enters node j from the outside is $p_{0j} = \lambda_{0j}/\Sigma_k \lambda_{0k}$. Assume the service and routing is done as above, and that there is a probability p_{j0} that a customer departing from j exits the network. Assume the Markov routing probabilities $\{p_{jk}\}$ are irreducible on $\{0, 1,\ldots, m\}$. The network is now an open Jackson network with unlimited capacity. In this case, the equilibrium distribution of the network process $\boldsymbol{X}$ is

$$\pi(\boldsymbol{x}) = \prod_{j=1}^{m} c_j \alpha_j^{x_j} \prod_{n=1}^{x_j} \phi_j(n)^{-1},$$

where

$$c_j^{-1} \equiv \sum_{k=0}^{\infty} \alpha_j^k \prod_{n=1}^{k} \phi_j(n)^{-1},$$

which is assumed to be finite, and α_j is the solution to the traffic equations

$$\alpha_j = \lambda_{0j} + \sum_{k=1}^{m} \alpha_j p_{jk}, j = 1, \ldots, m, \tag{2}$$

i.e., $\alpha_j = p_j/p_0$, where $p_0, p_1, \ldots, p_m$ is the stationary distribution of the Markov routing probabilities $\{p_{jk}\}$.

Another variation of this network is an open Jackson network with capacity v. This network operates like the open network described above except that when there are v customers in the network, the Poisson arrival streams are turned off (arrivals are turned away) — the total number of customers in the network may be from 0 to v. The equilibrium distribution of this network is given by (1), where α_j is the solution to (2) and the normalizing constant c is different.

Example: Single-Server Stations. Suppose each node is a single-server queue with service rate $\phi(x_j) = \mu_j$. If the network is closed with v customers, then

$$\pi(\boldsymbol{x}) = c\rho_1^{x_1} \ldots \rho_m^{x_m}, \tag{3}$$

where $\rho_j = \alpha_j/\mu_j$. In case the $\{\rho_j\}$ are distinct, $c = c_v^{-1}$, where

$$c_v = \sum_{j=1}^{m} \rho_j^{n+m-1} \prod_{k \neq j} (\rho_j - \rho_k)^{-1}.$$

There is a more complicated formula for c for non-distinct $\{\rho_j\}$. If the network is open with finite capacity v, then the equilibrium distribution is as in (3) with $c^{-1} = \sum_{n=0}^{v} c_n$ for distinct $\{\rho_j\}$. If the network is open with unlimited capacity and $\rho_j < 1$ for each j, then $\pi(\boldsymbol{x}) = \prod_{j=1}^{m}(1-\rho_j)\rho_j^{x_j}$. This is a product of equilibrium distributions of individual birth-death processes with birth rates α_j and death rates μ_j, and thus is called a product-form solution.

One can compute various quantities of interest from the equilibrium distribution π such as the marginal distributions, means and variances of quantities of customers at a node or in a sector of the network (a subset of nodes) and expected costs associated with network loads and movements. For closed or finite-capacity open networks, there are algorithms for computing these quantities (the unlimited-capacity open network is simpler because of its underlying product-form decomposition). Another approach is to estimate the quantities by a Monte Carlo simulation of a Markov chain (e.g., a Metropolis Markov chain) that has the same equilibrium distribution as the network; this is useful for large networks.

Important performance measures for a network are its throughputs. The throughput from node j to node k is the average number of network transitions per unit time in which a customer moves from j to k. This quantity is also the expected number of these transitions per unit time when the system is in equilibrium. The throughput from j to k is $\lambda_{jk} = \alpha_j p_{jk}$ when the Jackson network is open with unlimited capacity, and it is $\lambda_{jk} = c(v)c\,(v-1)^{-1}\alpha_j p_{jk}$ when the network is closed with v customers (or open with capacity v). Here $c(n)$ is the normalizing constant for a closed network with n customers (or an open network with capacity n). The throughput of node j is $\lambda_j = \Sigma_{k \neq j}\lambda_{jk}$. Then $\lambda_j = \alpha_j$ in case the network is open with unlimited capacity, and $\lambda_j = c(v)c\,(v-1)^{-1}\alpha_j$ for the other two cases. Similarly, the throughput of a sector J is $\lambda_J = \Sigma_{j \in J}\lambda_j$.

The main customer performance measures are sojourn times at the nodes, when the network is in equilibrium. The sojourn time T_j of a customer at node j is its service time plus time waiting for service. The expected sojourn time $W_j = E[T_j]$ is obtained by Little's law $L_j = \lambda_j W_j$, where L_j is the expected number of customers at node j, which is computed from the equilibrium distribution π. Similarly, the expected sojourn time W_J in a sector J can be obtained via Little's Law applied to the sector: $L_J = \lambda_J W_J$. Also, in an open network with unlimited capacity, the expected time a customer spends in a sector (in all of its visits) is $\overline{W}_J = L_J/\Sigma_{j \in J}\lambda_{0j}$

What is known about travel times in Jackson networks? Consider an open Jackson network with unlimited capacity in equilibrium. A simple route in the network is a set of nodes $1, \ldots, l$ such that a customer is able to traverse them in that order. Assume this is an overtake-free route in the sense that each node consists of a single server with a first-come-first-serve discipline, and once a customer is on the route, it cannot be overtaken by another customer. Then the sojourn times $T_1, \ldots, T_l$ at the respective nodes for an arbitrary customer that traverses the nodes in that order are independent exponential random variables and $E[T_j] = (\mu_j - \alpha_j)^{-1}$, where μ_j is the service rate at node j. When the network is closed or open with finite capacity, then $T_1, \ldots, T_l$ are dependent, but they have a known closed-form, multi-dimensional generating function. There are no comparable results for non-overtake-free routes.

In addition, there are closed-form expressions for a customer's expected travel time on a variety of very complicated routes. Examples include the time to travel from one sector J to another sector K, the time it takes to make n visits to a node j, and the time a customer spends in an open network while avoiding a sector J.

The preceding results for travel times are based on the MUSTA property that a "moving unit sees a time average." That is, suppose the Jackson network is open with unlimited capacity and is in equilibrium. Then at a transition in which a customer moves from one node to another, the probability distribution of the "other" customers in the network is the same as the equilibrium distribution of the network. Similarly, if the network is closed or open with finite capacity, then when a customer moves, the distribution of the other customers is the same as the equilibrium distribution of the network with one less customer. This MUSTA property is analogous to the PASTA property that Poisson arrivals to a single service queueing station see a time average.

Although the flow of customers over time from one node to another is generally not a Poisson process, there are some exceptions. Suppose the Jackson network is open with unlimited capacity and it is in equilibrium. Then the flow of customers departing the network from node j is a Poisson process with rate $\alpha_j p_{j0}$ (provided the rate is positive), and these Poisson departure processes are independent. Such a result might be of use for production planning for a manufacturing network, where the departures are customers of a finished product. Some internal flows are also Poisson processes. Suppose that customer j is such that a departure from it can never return. Then the flow of customers from node j to node k is a Poisson process with rate $\alpha_j p_{jk}$, and all these Poisson flows out of j are independent. There are no Poisson flows in closed or finite-capacity open networks, because the total number of customers is constrained.

Other Network Models

The following is a summary of other types of Markovian network models that have closed-form equilibrium distributions.

Whittle Networks

These networks operate like the Jackson network described above with the generalization that the service rate at node j is a function $\phi_j(\boldsymbol{x})$ of the entire network state $\boldsymbol{x}$ that satisfies a certain balance condition. Such system-dependent service rates are useful for modeling congestion dependent services.

Reversible Networks

Suppose the m-node network process $\boldsymbol{X}$ discussed above is a positive recurrent Markov process with transition rates $q(\boldsymbol{x}, \boldsymbol{y})$ from state $\boldsymbol{x}$ to state $\mathbf{y}$. The state x_j at node j may contain more information than just the quantity at the node, and the transitions can be very general (not necessarily like the single-customer movement in Jackson networks). The process is reversible if there is a probability measure π that satisfies the detailed balance equations $\pi(\boldsymbol{x})\, q(\boldsymbol{x}, \boldsymbol{y}) = \pi(\boldsymbol{y})\, q(\boldsymbol{y}, \boldsymbol{x})$ for each $(\boldsymbol{x}, \boldsymbol{y})$. In this case, π is the equilibrium distribution of the process, and there is a closed-form expression for π in terms of the transition rates. Such reversible processes can model dependent stations and batch movements, as well as single-customer movements of customers, provided the entire system is reversible.

Multi-Class Networks

Jackson, Whittle and reversible networks described above with homogeneous customers have analogues with multiple types of customers. In these networks, the state is a vector with components x_{γ_j} that represent the number of customers of class γ at node j. The network dynamics are the same, but the single subscript j is simply changed to a double subscript γj. For instance, in a Jackson network, a transition consists of a type γ customer at node j moving to node k and arriving there as an η customer — the rate of this transition is $\phi_{\gamma_j}(x_{\gamma_j}) p_{\alpha_j \eta_k}$. These models can represent fixed routes of customers fed by Poisson arrivals; a customer's type is $\gamma = rs$, where r is the route it is traversing and s is the stage (or node) on the route.

Networks with Batch or Concurrent Customer Movements

Multiple-customer movements in a network are represented by transitions from a state $\boldsymbol{x}$ to a state $\boldsymbol{x} + \boldsymbol{a} - \boldsymbol{d}$, where $\boldsymbol{a}$ and $\boldsymbol{d}$ are vectors that are added

and subtracted from the state $\boldsymbol{x}$. The models described above have analogues with certain types of multiple-customer movements that still lead to tractable stationary distributions.

Quasi-reversible Networks and Product-Form Equilibrium Distributions

Loosely speaking, a service station is quasi-reversible if its input and output processes in equilibrium are Poisson. Such stations can be connected with certain types of routings to produce a quasi-reversible network process whose equilibrium distribution is a product form $\pi(\boldsymbol{x}) = c\pi_1(x_1)\cdots\pi_m(x_m)$. The state x_j may contain more information than the number of customers at node j. This product-form distribution is a generalization of the Jackson product-form distributions above. There are also more general network models with product-form equilibrium distributions, where the $\{\pi_j\}$ are equilibrium distributions of the stations in isolation whose parameters are linked by certain traffic equations.

Networks with String Transitions

In this type of network, a transition is determined by a string of vectors representing multi-stage subtractions or additions of vector quantities at the nodes, and all of this is done instantaneously in a transition. The strings are randomly selected from an arbitrary family of variable-length strings. The equilibrium distributions contain parameters, like the $\{\alpha_j\}$ in the Jackson network, that are determined by nonlinear traffic equations.

Concluding Remarks

Jackson networks and some of the other models are discussed in Kelly (1979), Walrand (1988), Whittle (1986), Wolff (1989), and Boucherie and van Dijk (2011), and all of the models are discussed in Serfozo (1999). Disney and Kiessler (1987) study traffic flows in networks, and van Dijk (1993) discusses modeling by a systems approach.

Topics related to networks of queues that were not discussed include Brownian motion models for approximating networks in heavy traffic, fluid models of (discrete or continuous) flows in networks, polling systems in which servers move among stations, stochastic PERT networks, interacting particle systems, Petri net formulations of networks, space-time Poisson processes for modeling networks with no queueing, and spatial queueing systems. Some of these can be found in Boucherie and van Dijk (2011). Another excellent source for reading about developments in queueing networks is the journal *Queueing Systems: Theory and Applications*. For example, a special issue of the journal in 1998 reviews the state-of-the-art of Brownian queueing network models.

See

- ▶ Jackson Network
- ▶ Little's Law
- ▶ Markov Chains
- ▶ Markov Processes
- ▶ PASTA
- ▶ PERT
- ▶ Point Stochastic Processes
- ▶ Queueing Theory

References

Boucherie, R. J., & van Dijk, N. M. (Eds.). (2011). *Queueing networks: A fundamental approach*. New York: Springer.

Chao, X., Miyazawa, M., & Pinedo, M. (1999). *Queueing networks: Negative customers signals and product form*. New York: John Wiley & Sons.

Chen, H., & Yao, D. D. (2001). *Fundamentals of queueing networks: Performance, asymptotics, and optimization*. New York: Springer (paperback version 2010).

Disney, R. L., & Kiessler, P. C. (1987). *Traffic processes in queueing networks: A Markov renewal approach*. Baltimore: Johns Hopkins University Press.

Kelly, F. P. (1979). *Reversibility and stochastic networks*. New York: John Wiley.

Serfozo, R. F. (1999). *Introduction to stochastic networks*. New York: Springer.

van Dijk, N. M. (1993). *Queueing networks and product forms: A systems approach*. New York: John Wiley.

Walrand, J. (1988). *Introduction to queueing networks*. New Jersey: Prentice-Hall.

Whittle, P. (1986). *Systems in stochastic equilibrium*. New York: John Wiley.

Wolff, R. W. (1989). *Stochastic modeling and the theory of queues*. New Jersey: Prentice-Hall.

Neural Networks

Alice E. Smith[1] and Sarah S. Lam[2]
[1]Auburn University, Auburn, AL, USA
[2]Binghamton University, Binghamton, NY, USA

Introduction

A field that was started in the 1940s, when McCulloch and Pitts (1943) designed the first neural networks where artificial neurons are combined into a network structure, has attracted researchers from diverse disciplines. Inspired by biological neural networks, artificial neural networks crudely imitate human brains in processing information, recognizing patterns and retrieving stored information. For simplicity, artificial neural networks will be referred to as neural networks.

The first learning law for neural networks was designed by Hebb (1949), and later expanded by McClelland and Rumelhart (1988). The field of neural networks has gone through several stages since its beginnings, including some quiet years in the 1970s (Fausett 1994). A large number of neural network paradigms have appeared in the literature, including the Hopfield networks, multilayer perceptrons (also known as the backpropagation networks), self-organizing maps, adaptive resonance theory networks, radial basis function networks and general regression neural networks.

Neural networks are largely empirical models whose robustness and flexibility are prime advantages. On the debit side, neural networks generally require substantial amounts of data, require considerable expertise to properly construct and validate, and offer little in the way of discernible model structure to better understand the relationship modeled. Neural networks have been successfully applied to prediction, forecasting, process modeling, financial and business applications, combinatorial optimization, classification and control.

After describing several of the most popular neural networks, namely, backpropagation networks, self-organizing maps, and general regression neural networks, recent applications of neural networks to operations research type of problems will be discussed in the following sections.

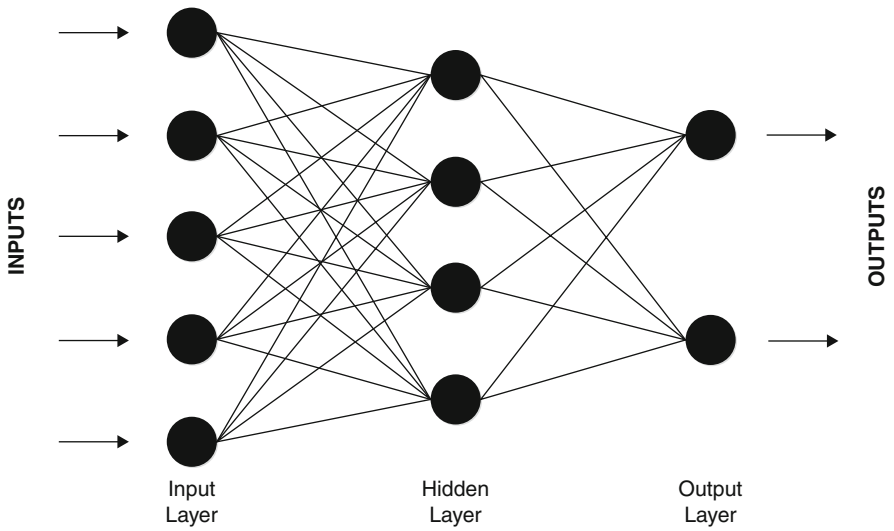
Neural Network Basics

An example of a network structure is shown in Fig. 1. External information is used as inputs (i.e., independent variables) to the network. The interconnections (in the form of real valued, or occasionally binary valued, weights) between the nodes in the input layer and the nodes in the hidden layer, and that between the hidden nodes and nodes in the output layer, represent the knowledge acquired or learned during the iterative training process of the network. The training algorithm for adjusting the connected weights is dependent on the specific network paradigm. During the training process, the interconnections or weights are adjusted according to some algorithm(s). Once the network is trained, it is then able to produce an output (or a set of outputs) for a given input set.

If the training set contains the target information (i.e., known values of the dependent variables), the learning algorithm is a supervised learning algorithm. Examples of supervised networks are backpropagation networks, radial basis function networks and general regression neural networks. On the other hand, if the training set does not contain the target information, the learning algorithm is an unsupervised learning algorithm. Examples of unsupervised networks are self-organizing maps and some versions of the adaptive resonance theory networks. Human and animal learning incorporates both supervised and unsupervised learning but in neural networks, a given paradigm will use one form of learning or the other.

Building and Validating a Neural Network

The process of constructing a neural network for an application generally involves three steps: data-preprocessing, model design and model validation. The first step is to determine which variables are to be modeled and these are generally divided into input variables and output variables. It is usually preferred to use the minimum number of variables that provide adequate characterization of the relationship to be modeled. Neural networks depend on data to establish the model. In fact, they normally require greater amounts of data than traditional statistical methods such as least squares



Neural Networks, Fig. 1 Network structure

regression or kriging. The dataset may require investigation on the characteristics of outliers, and may require preprocessing to extract the important features for the problem at hand. The dataset is normally divided into a training set and a validation set, though there are approaches which use all data for both training and testing (see for example, the resampling techniques used in Twomey and Smith (1998)).

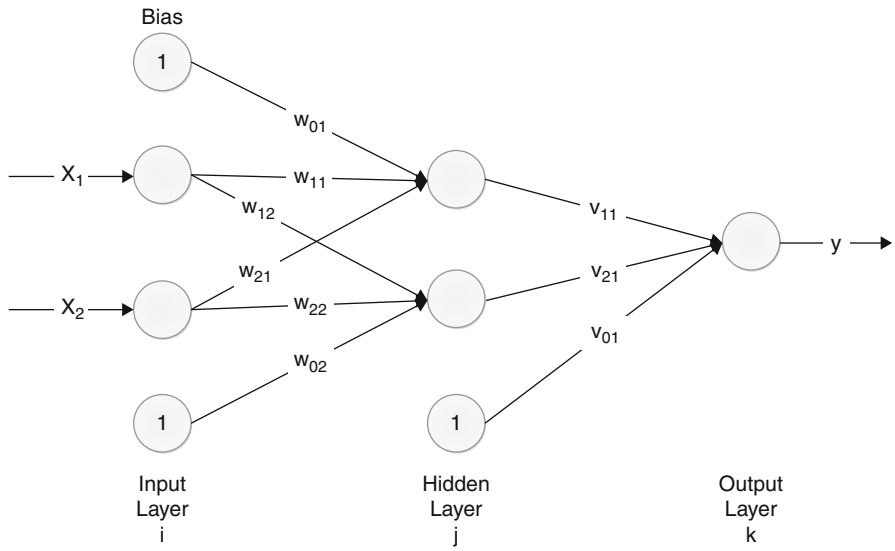
Once the data has been pre-processed, one or multiple networks may be selected for model building. This step often depends on the network modeler's experience and the network paradigms that have been successfully utilized for the application considered. Designing a network includes selection of the network paradigm, network size and structure, learning method and parameters, and stopping criteria for learning. A design of experiments could be used to fine-tune the network parameters. Designing and building a neural network is normally an artful and iterative task. There is little in the way of useful general guidelines and it is often difficult to choose a priori a superior set of network structures and parameters.

After the network has been trained using a training-set, the trained network is usually validated using a separate set of data (i.e., a validation set). The performance on the validation set is often used as an estimate of the network's generalization ability. Achieving a low error on the training set but a substantially high error on the validation set generally indicates overtraining and/or over specification of the network model. This often implies that the network memorizes the training set too well, and its ability to generalize on the validation set has significantly degraded.

After the neural network has been trained and validated, its structure and parameters are then fixed and it can be used for the task intended. Some neural network paradigms have the ability to continue learning even while in the operational state (see, for example, the adaptive resonance theory network). From a usage perspective, neural networks have the disadvantage of being a "black box" technology. That is, little useful insight can be gained by examining its parameters. This is unlike a regression model where intercept and slopes have a readily identifiable interpretation. There are also few statistical properties that can be calculated from neural networks.

Common Neural Network Paradigms

The so-called backpropagation network (the training method is actually backpropagation and it is generally used on the network structure termed multi-layer perceptrons), one of the most popular network paradigms, has been applied in many different areas in the literature. A simple 2-2-1 network that consists of 2 input nodes in the input layer, 2 hidden nodes in

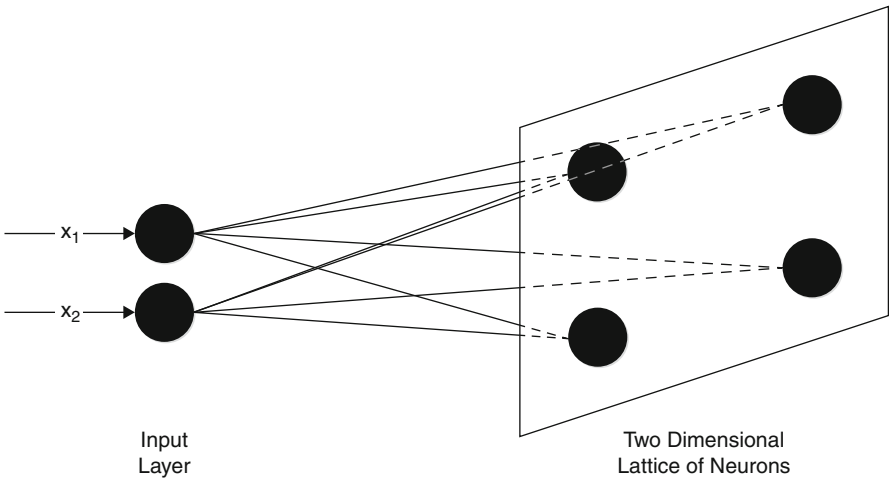


Neural Networks, Fig. 2 A 2-2-1 network

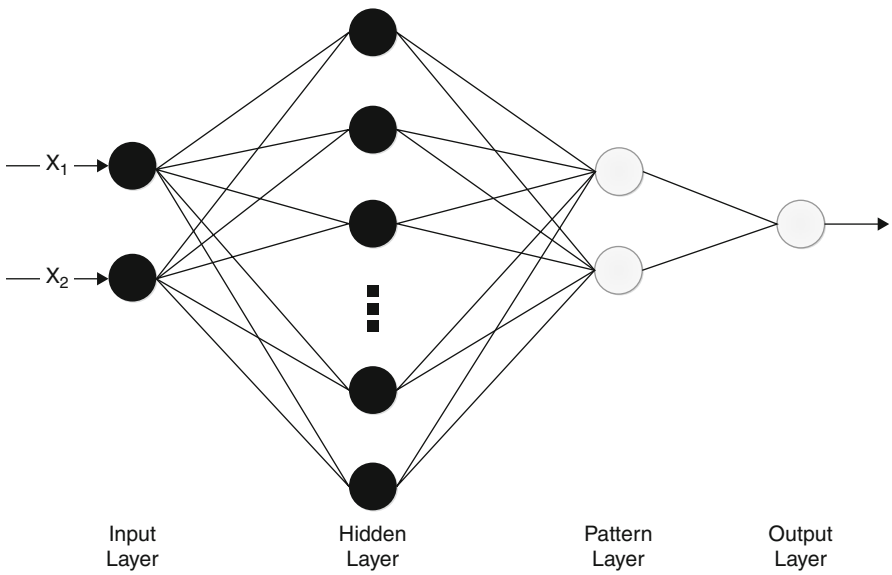
the hidden layer, and 1 output node in the output layer, is shown in Fig. 2. During the training process, each training vector is fed to the input layer. The inputs, weighted by the connected weights w_{ij} along with the bias are fed forward to the hidden nodes in the hidden layer. An activation function is then applied to the input signal to each hidden node to produce an output (outgoing) signal to the following layer. Common activation functions are the hyperbolic tangent and sigmoid. The output signals, weighted by v_{jk} along with the bias term are fed forward to the output node in the output layer. The received signal at the output node undergoes a transformation, via an activation function, to produce a network output in the output layer. The difference between the network output and the target output is then propagated backward to adjust the connected weights in the network. The adjustment is dependent on the squared error in the output layer, and a learning factor chosen for the backpropagation learning algorithm. The training process of the network continues until the weight adjustments become negligible or a predetermined number of iterations of training have been completed. The training process is equivalent to the determination of optimal weights that minimize the error between the network output and the target output. A mathematical formulation of the backpropagation learning algorithm is provided by Fausett (1994). These types of networks are multi-purpose and are usually used for prediction, classification and control.

The **self-organizing map**, also known as the Kohonen's self-organizing map, is based on a competitive learning algorithm (Kohonen 1982, 1997). The neurons in the self-organizing map (shown in a two-dimensional lattice in Fig. 3) are selectively tuned or adjusted according to the input information received during training. According to a distance metric (e.g., Euclidean distance), the neuron that is closest to the input vector wins the competition and is allowed to update its connected weights to the input nodes in the input layer. Over the course of the training process, the neurons are topologically ordered and their weight vectors represent an approximation of the input space. This approximation is also known as the feature map. One of the basic ideas of the self-organizing map is to extract the features from the input space and represent the features using a smaller set of neurons in the output space. Variations on the neighborhood structure of the neurons in the lattice allow for the winning neuron along with its nearest neighbors to "learn" from the input vector. The adjustment of the weight vectors gradually diminishes over time, which would signify that the training of the network has been completed. These types of networks are generally used for classification or clustering tasks.

The general regression neural network, developed by Specht (1991), is a probabilistic neural network. This network requires only a fraction of the training samples that a backpropagation network would



Neural Networks, Fig. 3 Kohonen's self-organizing map



Neural Networks, Fig. 4 General regression neural network

normally require (Specht 1991) (Fig. 4). Each neuron in the hidden layer uses a Gaussian probability distribution, centered at one of the training samples. Hence the connected weights between the input and the hidden layers are encoded as the input vectors (i.e., $\vec{x}$) in the training set. A training set of size n would require a total of n neurons in the hidden layer. One of neurons in the pattern layer, the numerator neuron, has its connected weights between the hidden and the pattern layers encoded as the target values of the input vectors (i.e., $\vec{y}$) in the training set. The vector $\vec{y}$ is a one-dimensional vector that has the same number of elements as the size of the training set. In the event that the original training samples have more than one output variable, each output variable would require a separate general regression neural network. The other neuron in the pattern layer, the denominator neuron, has its weights between the hidden and the pattern layers encoded as unit weights. The final output of the network (in the output layer) is the ratio of the weighted sum of inputs received at the numerator neuron in the pattern layer and the sum of inputs received at the denominator neuron in the pattern layer. Each hidden neuron contributes to the final output of the network. These types of networks are usually used for prediction and forecasting.

Prediction Applications

Probably the preponderance of applications of neural networks can be classified as prediction. Often these neural networks are used when a comparable statistical model (such as regression) proves to be too restrictive. These applications also have the property of ample data with which to train and validate the network model.

Brockett et al. (1997) discuss the possible advantages and feasibility of using neural networks to predict insurer insolvency (inability to pay debts). Their research examines a backpropagation network with a goal of obtaining early warnings (early being up to 3 years before) for both insolvency predictions, and priority ranking of insurance firms for potential auditing. The case study is constrained to insurance companies only located in Texas, and also insurance companies that only offer property and casualty insurance (as opposed to life and health insurance). Their model is designed having an input vector of eight financial variables to differentiate between healthy and weak insurers, with an output of the solvency results. The optimal number of iterations was based on the process of stopping the training once peak network performance is achieved. The results showed that the network was able to learn patterns corresponding to financial distresses of the companies, had a 95.45% accuracy rate in the 44 companies evaluated in determining the insolvency rate. Their research concluded with the benefits of using neural networks, specifically the ability to update without training the network from the beginning, using the current weights as a starting point for future iterations when more data becomes available, and the ability to adapt to changing economic influences.

Condition based maintenance is the act of keeping equipment maintained properly so that it does not fail. Remaining useful life (RUL) prediction of the equipment is instrumental in condition based maintenance. There exists two forms of RUL prediction: model-based (or physics-based), which relies on calculations involving the mechanics of the equipment, and data-based, which aims at predicting RUL based on a model of the relationship between RUL and equipment age, condition monitoring data, and equipment degradation. The adaptability, nonlinearity, and arbitrary function approximation ability of neural networks have been considered to be promising tools for RUL prediction. Tian et al. (2010) propose an approach for predicting RUL of equipment using a neural network with age and condition monitoring data as inputs and life percentage (1 - RUL) as the output. In this case, condition monitoring data includes some available failure history, as well as suspension history. Suspension history is information about a piece of equipment that has been suspended (removed) from use in its respective system. It provides useful insight into degradation of the piece of equipment and can lead to more accurate RUL predictions. The optimal predicted life is determined for each suspension history. The trained network was validated by using real-world vibration monitoring data collected from pump bearings. Their results showed that the neural network devised in their research can produce accurate RUL predictions. Another neural network application to condition-based maintenance can be found in Smith et al. (2010). This group developed hardware and software to estimate degradation of the door mechanism on airport people mover vehicles using a backpropagation neural network. This neural-based system was field tested at an airport and resulted in multiple patents issued.

Neural nets are known for their ability to estimate continuous functions well but the model is of a "black box" approach where the end-user does not know precisely how the model came to its conclusion. Setiono and Thong (2004) demonstrate the extraction of the knowledge that the neural network has learned in order to achieve better system understanding. The first step is to estimate the hyperbolic tangent activation function for each hidden unit of the hidden layer from a trained and pruned neural net by means of a three-piece linear function. Then the input region is divided and a linear function is estimated for each sub-region. Since the estimated linear function obtained for each region has inputs that are weighted, i.e., the weights from the input layer to the hidden layer, Setiono and Thong (2004) propose optionally the use of the C4.5 decision tree to get a set of rules independent of the weights.

According to Ladstatter et al. (2010), organizational research problems are seldom studied via neural networks and burnout has never been studied with neural networks, thus making their research unique. If burnout could be predicted accurately

then preventative measures could be used in an attempt to alleviate the problem. Burnout is described as being three dimensional, namely, emotional exhaustion, depersonalization and lack of personal accomplishment. The radial basis function neural network in this case uses seven input variables (conflictive interaction, workload, experience with pain and death, role ambiguity, age, job status and hardy personality) to predict the level of each of the three dimensions of burnout. Three types of neural networks, namely backpropagation and radial basis function networks with traditional and hybrid training algorithms, were constructed to see which model was the best. A hierarchical stepwise regression model was also developed from the same data for comparison purposes. The results were modest in that the radial basis function neural network, being the best neural network in this research, slightly outperformed the regression model in two of three dimensions of burnout.

A backpropagation network was used to predict internet traffic over internet-protocol (IP) networks in Chabaa et al. (2010). Multiple learning algorithms were implemented for comparison, namely, gradient descent, conjugate gradient, one step secant, Levenberg-Marquardt and resilient backpropagation. Additionally four variants of the conjugate gradient algorithm were used, namely, Fletcher-Reeves updates, Polak-Ribiere updates, Powell-Beale restarts and scaled. All conjugate gradient algorithms start searching in the area of steepest descent followed by combining the next (new) search direction with previous search direction. The four conjugate gradient variants differ in how they combine the new with the previous search direction. The multiple learning algorithms were then compared by using the following statistical measures: root mean square error, scatter index, relative error and mean absolute percentage error. The results showed that the Levenberg-Marquardt and resilient backpropagation performed the best for the prediction task based on the above statistical measures.

Forecasting Applications

Similar to prediction problems, forecasting applications are most specific to a series of data where the ordering is important. There have been a number of papers in which neural networks have been used effectively for this task. There are also reports of applications of neural networks managing time series data to make investment decisions (see, for example, Evensky (1997)).

Hansen and Nelson (2003) show the use of neural networks to improve forecasts of time-series components that are not effectively forecasted using classical decomposition of the forecast components (trend, seasonality, irregularity (error or residuals)). Traditional forecasts do not work as well when the trend/cycle component is not linear, there is variation in cycles, the seasonality component evolves over time, or when the irregularity component (residuals) does not appear to be from white noise. Hansen and Nelson (2003) noted that it could be possible to apply neural networks to extract information from the seemingly noisy or irregularity component of the forecast. Their framework was based on a "stacked generalization" model. First classical decomposition was conducted providing estimates for trend, seasonality, and irregularity. Then each estimated component was input into a separate neural network for each of the forecast components (i.e., a trend neural network, seasonality neural network, irregularity neural network). The result of the stacked generalization model is a non-linear combination/forecast of the time-series components. Their stacked generalization model was tested and compared to results obtained for classical decomposition and ARIMA models.

Tiwari and Chatterjee (2010) used bootstrap-based neural networks to predict flooding at multiple time intervals based on hourly water-level data taken from five different locations. The authors used statistical techniques to reduce the dimensionality of the data set and then a log transformation followed by linear scaling to achieve inputs ranging from 0 to 1. The network design consisted of the input and output layers defined by the problem and one hidden layer with the number of nodes in the hidden layer determined by cross-validation. The initial network model showed good results in its ability to predict water levels at different lead times (0, 1, . . ., 10 hrs into the future). Furthermore, Tiwari and Chatterjee (2010) also introduced a method to incorporate confidence intervals for the output obtained from the bootstrap ensembles which helped reduce the model uncertainty.

Process Modeling

Process modeling is an active area of neural network applications. A great variety of neural network paradigms and approaches have been used on a variety of industrial processes.

Maksoud et al. (2003) established a system of two backpropagation networks for monitoring and controlling grinding operations. The first network is used for the grinding process design to achieve the required workplace surface roughness, with an output of suitable values of machine variables. The second network is used for the grinding process control, with an output of modifying the cutting variables or flagging for automatic dressing activation. Tests were run to evaluate response to change variables in order to keep the surface roughness within the desired tolerance. It was found that the network model kept the surface roughness within the desired tolerance, and the model also alerted the machine when it needed to stop the process and flag it for automatic dressing activation. The research work could potentially be implemented in other grinding operations with different sets of variables.

Coit et al. (2002) developed a neural network system for process modeling and control of a wave soldering line in the production of printed circuit boards. They developed a hierarchy of neural networks each with a specialized task. The ultimate network predicted quality of solder connection based on both printed circuit board design characteristics and on solder process parameters. Lam and Smith (2001) discuss three diverse applications in manufacturing and how systems of neural network models were successful in improving process control. These are ceramic casting of sanitary ware (Lam et al. 2000), abrasive flow machining for automotive engine parts, and chemical oxidation of cyclohexane in a reactor (Lam et al. 2001).

Gupta (2010) devised empirical models for the prediction of surface roughness, tool wear, and power required using response surface methodology, neural networks and support vector regression. Descriptive statistics and hypothesis testing were used to compare and evaluate the model building methods. The results showed that neural network and support vector regression models were superior to regression and response surface methodology in the prediction of surface roughness, tool wear, and power required.

Financial and Business Applications

While some of the application areas cited above can span this sector, a good review article is followed by sample applications specific to finance. Smith and Gupta (2000) review neural network applications in the business domain and indicate that they are really a tool for the operations researcher. The authors identify five stages of neural network research development, with the first stage being related to computing paradigms and the last stage being the research of neural networks in business applications. An overview of business application areas (non-exhaustive) are given as marketing, retail, banking and finance, insurance, telecommunications and operations management. Interesting and successful examples in each of these industry sectors are given to show the diversity of neural network applications.

West et al. (2005) investigated neural network ensembles to produce better decision support mechanism for financial decisions. The argument for "ensembles" is that predictions from multiple experts (models) can produce results with less error than when only using the best fitting model. The ensemble methods discussed include cross-validation, bootstrap aggregation (or bagging), and adaptive boosting (or boosting). Cross-validation is where multiple, similar neural networks are trained on the same data. Bagging and boosting methods involve perturbing the training data such that the different neural networks in the ensemble are each trained on unique training data sets that are subsets of the original training set. Bagging creates the training data by randomly selecting data with replacement from the original training data set, whereas boosting creates multiple, unique training data sets that each contain multiple examples of hard-to-classify examples that a single neural network could not learn itself from the original training data. After the ensemble member networks are trained, the outputs are combined, or aggregated in a single decision. The research of this paper included creating different ensemble neural networks for three different financial data sets and comparing the results of the ensembles with each other and the single "best fitting" model. The results of their research show that cross-validation and bagging outperform the single "best fitting" model in all three data sets while the boosting method did not always outperform

the "best fitting" model, most likely due to outliers and possibly the need for better boosting algorithms. In all cases the reduction in generalization error was modest (~3%), but it was argued that in the case of financial decisions in a trillion dollar annual industry can mean significant savings.

Grznar et al. (2007) use neural networks to model complex, organization systems where statistical models have proved inadequate. In particular the authors use multilayered feedforward neural networks with backpropagation learning to model the relationship between inter & intra team processes, organizational context, and team size to team effectiveness. The basic argument that statistical procedures are not suited for this problem is due to outliers in the data and the non-linear relationships in complex systems. Grznar et al. (2007) created the training data by means of a survey that included 102 different teams from different organizations. The data was modeled by the neural network and also with traditional and robust regression (robust regression uses least median of squares to make the model less susceptible to outliers) in order to compare the results. The traditional and robust regression models each found the intra-team processes to be significant and the regression models had R^2 values of 0.247 and 0.313, respectively, which apparently are good R^2 values for organizational models. It is suspected that low R^2 values of organizational models are due to non-linear behavior that cannot be explained with these models. The neural network model resulted in a R^2 of 0.414, which was significantly better than the previous models. The authors were also able to experiment with multiple data streams on the validated model and found some interesting non-linear relationships. Among these relationships is that as team size increases it initially decreases team effectiveness but then as the team size becomes even larger team effectiveness begins to increase again.

Classification Applications

Fisch et al. (2010) made use of radial basis function neural networks to classify system behaviors, in particular to detect network intrusion. The radial basis function neural network is very similar to that of a backpropagation network except that the activation of the hidden neurons is calculated by radial basis functions instead of sigmoid or hyperbolic tangent functions. The training and test data for this research were from a DARPA project conducted in 1998, which contained over 300 examples of 38 different types of intrusions. This paper used a subset of the DARPA dataset which had a higher proportion of intrusions versus safe actions in order to make classification easier. Fisch et al. (2010) compared multiple methods including the two neural networks, neuro-fuzzy, decision trees, fuzzy-k-means, support vector machines, Bayesian networks and k-nearest-neighbor to report on which performs the best. The results showed that the radial basis function neural network performed slightly better than the k-nearest-neighbor approach but that there was no clear winner among the classification approaches.

Zhang (2000a) compared backpropagation networks with statistical classification and showed the superiority of neural networks when the model is non-linear or when the underlying distribution function is unknown. Additionally it was shown that with appropriate architectures one can essentially mimic the most widely used statistical classifiers with a neural network, which further shows their utility. The discussion addresses learning and generalization for future predictions and how they are related to model bias and variance. Overall, Zhang's paper is a good review of backpropagation networks' application to classification problems.

Control and Optimization

Sensor fault detection and isolation is an integral part of flight control systems, especially for an unmanned aircraft. The most common solution for sensor failure is using redundant hardware and output limit checks to see which sensors are operating as expected. Due to high cost and additional weight loads of redundant hardware it is preferred to use a model based approach. The majority of model-based senor fault detection and isolation use linear models which obviously have drawbacks when the system is not linear. Samy et al. (2010) used an extended minimum resource allocation network radial basis function neural network for sensor fault detection and accommodation. Their neural network was implemented in a simulated environment and only one type of fault (pitch gyro) was considered. The neural network model has six inputs including:

angle of attack, normal acceleration, airspeed, altitude, elevator angle, and throttle. There was a single output node corresponding to pitch rate. The model had to be able to detect eight different fault types corresponding to the output (pitch rate) exceeding certain thresholds. The neural network model was trained through batch learning to find all model parameters. Once the model was batch trained it was connected in parallel with a simulated model of an unmanned air vehicle. Sensor readings (six inputs) entered the unmanned air vehicle and neural network model, which is now switched to online learning, to obtain actual and estimated outputs, respectively. A residual was then generated from comparison that is used for fault detection, i.e., if this residual exceeds a certain threshold. The results indicate that the neural network was able to detect faults adequately. This work has shown the potential to use neural networks for unmanned air vehicle flight control, although it is not expected to replace hardware redundancy in the near future.

In optimization, neural networks were first noted as possibilities in the seminal paper by Hopfield and Tank (1985). Since that time there have been various approaches made and these are summarized and presented in the book by X. S. Zhang (2000b). It seems that other computational methods, however, are more well suited to optimization and the research community has largely moved on to those, such as the nature-based metaheuristics of genetic algorithms and ant colonies.

See

- ▶ Approximate Dynamic Programming
- ▶ Control Theory
- ▶ Metaheuristics
- ▶ Regression Analysis
- ▶ Response Surface Methodology
- ▶ Simulation Metamodeling

References

Brockett, P. L., Cooper, W. W., Golden, L. L., & Xia, X. (1997). A case study in applying neural networks to predicting insolvency for property and casualty insurers. *Journal of the Operational Research Society, 48*(12), 1153–1162.

Chabaa, S., Zeroual, A., & Antari, J. (2010). Identification and prediction of internet traffic using artificial neural networks. *Journal of Intelligent Learning Systems and Applications, 2*, 147–155.

Coit, D. W., Turner Jackson, B., & Smith, A. E. (2002). Neural network open loop control system for wave soldering. *Journal of Electronics Manufacturing, 11*(1), 95–105.

Evensky, H. (1997). *Wealth management: The financial advisor's guide to investing and managing client assets*. New York: McGraw-Hill.

Fausett, L. V. (1994). *Fundamentals of neural networks: Architectures, algorithms, and applications*. Englewood Cliffs, NJ: Prentice Hall.

Fisch, D., Hofmann, A., & Sick, B. (2010). On the versatility of radial basis function neural networks: A case study in the field of intrusion detection. *Information Sciences, 180*, 2421–2439.

Grznar, J., Prasad, S., & Tata, J. (2007). Neural networks and organizational systems: Modeling non-linear relationships. *European Journal of Operational Research, 181*, 939–955.

Gupta, A. K. (2010). Predictive modelling of turning operations using response surface methodology, artificial neural networks and support vector regression. *International Journal of Production Research, 48*(3), 763–778.

Hansen, J. V., & Nelson, R. D. (2003). Forecasting and recombining time-series components by using neural networks. *Journal of the Operational Research Society, 54*(3), 307–317.

Hebb, D. O. (1949). *The organization of behavior*. New York: Wiley.

Hopfield, J. J., & Tank, D. W. (1985). 'Neural' computation of decisions in optimization problems. *Biological Cybernetics, 52*(3), 141–152.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics, 43*(1), 59–69.

Kohonen, T. (1997). *Self-organizing maps* (2nd ed.). Secaucus, NJ: Springer.

Ladstatter, F., Garrosa, E., Badea, C., & Moreno, B. (2010). Application of artificial neural networks to a study of nursing burnout. *Ergonomics, 53*(9), 1085–1096.

Lam, S. S. Y., Petri, K. L., & Smith, A. E. (2000). Prediction and optimization of a ceramic casting process using a hierarchical hybrid system of neural networks and fuzzy logic. *IIE Transactions, 32*(1), 83–91.

Lam, S. S. Y., & Smith, A. E. (2001). Neural network predictive process models: Three diverse manufacturing examples. In A. Kusiak & J. Wang (Eds.), *Handbook of computational intelligence in design and manufacturing* (pp. 11-1–11-12). Boca Raton: CRC Press LLC, Chapter 11.

Lam, S. S. Y., Smith, A. E., & Morsi, B. I. (2001). Estimation of a mass transfer coefficient for nylon manufacture using multiple neural networks. *Journal of Manufacturing Systems, 20*(5), 349–356.

Maksoud, T. M. A., Atia, M. R., & Koura, M. M. (2003). Applications of artificial intelligence to grinding operations via neural networks. *Machining Science and Technology, 7*(3), 361–387.

McClelland, J. L., & Rumelhart, D. E. (1988). *Explorations in parallel distributed processing*. Cambridge, MA: MIT Press.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics, 5*, 115–133.
Samy, I., Postlethwaite, I., & Gu, D.-W. (2010). Sensor fault detection and accommodation using neural networks with application to a non-linear unmanned air vehicle model. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 224*(4), 437–447.
Setiono, R., & Thong, J. Y. L. (2004). An approach to generate rules from neural networks for regression models. *European Journal of Operational Research, 155*, 239–250.
Smith, A. E., Coit, D. W., & Liang, Y.-C. (2010). Neural network models to anticipate failures of airport ground transportation vehicles. *IEEE Transactions on Automation Science and Engineering, 7*(1), 183–188.
Smith, K. A., & Gupta, J. N. D. (2000). Neural networks in business: Techniques and applications for the operations researcher. *Computers and Operations Research, 27*, 1023–1044.
Specht, D. F. (1991). A general regression neural network. *IEEE Transactions on Neural Networks, 2*, 568–576.
Tian, Z., Wong, L., & Safaei, N. (2010). A neural network approach for remaining useful life prediction utilizing both failure and suspension histories. *Mechanical Systems and Signal Processing, 24*, 1542–1555.
Tiwari, M. K., & Chatterjee, C. (2010). Uncertainty assessment and ensemble flood forecasting using Bootstrap Based Artificial Neural Networks (BANNs). *Journal of Hydrology, 382*, 20–33.
Twomey, J. M., & Smith, A. E. (1998). Bias and variance of validation methods for function approximation neural networks under conditions of sparse data. *IEEE Transactions on Systems, Man, and Cybernetics, Part C, 28*(3), 417–430.
West, D., Dellana, S., & Qian, J. (2005). Neural network ensemble strategies for financial decision applications. *Computers and Operations Research, 32*, 2543–2559.
Zhang, G. P. (2000a). Neural networks for classification: A survey. *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, 30*(4), 451–462.
Zhang, X. S. (2000b). *Neural networks in optimization*. Dordrecht, The Netherlands: Kluwer.

Neuro Dynamic Programming

Another name for approximate dynamic programming or reinforcement learning, where the "neuro" comes from the fact that a neural network might be used to approximate the value function.

See

- ▶ Approximate Dynamic Programming

Newsboy Problem

Items, here newspapers, have to be procured at the beginning of a time period and are discarded (or sold at a discounted price) at the end of the time period. The demand is assumed to be a random variable with known distribution. The problem is to determine how many items to stock at the beginning of the time period to minimize expected cost. This leads to a closed-form, single-period inventory model with stochastic demand. The problem statement also applies to items such as Christmas trees, time-dependent fashions, and items that can be stored until the next season like snow tires and Chanukah candles.

See

- ▶ Inventory Modeling

Newsvendor Problem

- ▶ Inventory Modeling
- ▶ Newsboy Problem

Newton's Method

Local search method for root finding or optimization requiring higher-order information, e.g., (inverse) Hessian in multi-dimensional steepest descent or ascent methods.

See

- ▶ Convex Optimization
- ▶ Interior-Point Methods for Conic-Linear Optimization
- ▶ Nonlinear Programming
- ▶ Quadratic Programming
- ▶ Stochastic Approximation
- ▶ Unconstrained Optimization

NLP

► Nonlinear Programming

Node

An element of a graph or network, pairs of which are connected by arcs or edges. Nodes are sometimes referred to as points or vertices. In queueing networks, a node is also called a station and represents a simple queueing subsystem consisting of a service center with one or more servers, a queueing capacity (infinite or finite), and a queue discipline. In a project network plan, a node is shown graphically as a circle depicting the beginning or end of an activity, and represents an instantaneous point in time at the junction of arrows.

See

► Graph Theory
► Networks of Queues
► Network Optimization
► Network Planning

Node-Arc Incidence Matrix

For the minimum-cost network-flow problem, this is a matrix in which the rows i correspond to the nodes and the columns j correspond to the arcs. For an arc (i, j), with its flow directed from i to j, the entry in matrix location (i, j) is $a + 1$ and the entry in location (j, i) is $a - 1$. All other entries are zero. Thus, every column has only two nonzero entries. Such matrices are unimodular.

See

► Minimum-Cost Network-Flow Problem
► Multicommodity Network Flows
► Network Optimization

Nonactive (Nonbinding) Constraint

An inactive constraint.

See

► Active Constraint
► Inactive Constraint

Nonbasic Variable

Given a feasible basis to a linear-programming problem, a variable is nonbasic if it does not correspond to one of the vectors in the basis.

See

► Basic Variables

Nondegenerate Basic Feasible Solution

A feasible basis to a linear-programming problem is nondegenerate if all basic variables are strictly positive.

See

► Basic Feasible Solution
► Degeneracy
► Degenerate Solution

Nondominated Solution

► Efficient Solution

Nonlinear Goal Programming

A goal programming methodology used to solve goal programming problems that have nonlinear elements in their model formulation.

See

► Goal Programming

Nonlinear Programming

Anthony V. Fiacco
The George Washington University,
Washington, DC, USA

Introduction

Nonlinear programming, a term coined by Kuhn and Tucker (Kuhn 1991), has come to mean the collection of methodologies associated with any optimization problem where nonlinear relationships may be present in the objective function or the constraints. Since maximization and minimization are mathematically equivalent, without loss of generality the nonlinear programming problem discussed throughout will be the problem of finding a solution point or optimal value of

$$\text{minimize } f(\boldsymbol{x}) \text{ subject to } g_i(\boldsymbol{x}) \geq 0 \quad (i = 1, \ldots, m)$$
$$\text{and } h_j(\boldsymbol{x}) = 0 \quad (j = 1, \ldots, p) \qquad (P)$$

where all problem functions are real valued. The underlying space can be more general but is here assumed $\boldsymbol{x} \in R^n$. In this terminology and context, problem (P) is a linear program (LP) if f, g_i and h_j are linear (technically linear-affine, i.e., linear plus a constant) for all i and j.

Another very important instance of problem (P) is where the constraints g_i and h_j are not present or where every point in the domain of f is feasible (i.e., satisfies the constraints). This is called an unconstrained problem and goes back to the very early days of mathematics.

Simple Examples

Finding the highest point of a pyramid may be viewed as a linear programming problem. Assuming that equations of the planes that contain the sides and base of the pyramid can be found, the pyramid is essentially the feasible region, the set of points in the volume contained by these planes, and the problem is to find the point in the region that yields the greatest height.

A somewhat analogous example is that of finding the deepest point in a lake, where the shoreline is the constraint, the surface of the water is the feasible region, and the depth is the objective function. This example would generally be highly nonlinear.

Another readily understandable example is that of finding a point where the maximum or minimum altitude is attained in a given area, e.g., find the highest point in the state of Virginia. The constraints are determined by the state boundaries. Lines of equal altitude are often displayed on a map and correspond to isovalue contours or level curves of the objective function (altitude), in mathematical programming terminology. Thus, the goal is to find the latitude and longitude of a location in Virginia (the feasible region) on a level curve of maximum value, a problem that is (logically) equivalent to a realization of a nonlinear problem of the form of problem (P). There are many local maxima (i.e., hills and peaks) in this problem that are not global, a formidable challenge to solving (P).

Mathematical Examples

Obtaining a solution to a system of equations $h_1(\boldsymbol{x}) = 0, \ldots, h_p(\boldsymbol{x}) = 0$ where $\boldsymbol{x} \in R^n$ may be posed as the unconstrained NLP, P: min $\Sigma_{j=1}^{q} h_j(x)^2$ s.t. $x \in R^n$, or equivalently, P: min $\|\boldsymbol{h}(\boldsymbol{x})\|^2$, where $\boldsymbol{h} = (h_1, \ldots, h_p)^{\mathrm{T}}$ and the norm is the usual Euclidian norm. Alternatively, one could choose to solve min $\|\boldsymbol{h}(\boldsymbol{x})\|^2$ or min $\|\boldsymbol{h}(\boldsymbol{x})\|$ for any suitable choice of norm. There may be no solution to the system of equations, but the indicated NLP problems can still be addressed, and their solutions would yield points that minimize the residual error, i.e., the deviation of $\boldsymbol{h}(\boldsymbol{x})$ from $\mathbf{0}$, in the sense of the given norm. The choice Σh_j^2 leads to a so-called least-squares solution and is undoubtedly the most popular, yielding a smooth (i.e., differentiable) problem if $\boldsymbol{h}$ is differentiable (the other

measures usually being nonsmooth). If the solutions are underdetermined, one could use the degrees of freedom to seek to determine a solution having a desired quality such as minimum norm, i.e., a solution of the constrained NLP, min $\|\boldsymbol{x}\|$ s.t. $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}$.

The minimum norm-residual idea turns out to be extremely fruitful and is a driving mechanism for several important classes of NLP problems, including regression, also known as parameter estimation or data fitting and essentially a type of curve-fitting, minimum distance problems, and eigenvalue problems. The idea of regression is to assume a functional form, say $y = F(\alpha,\ x) + \varepsilon$ that relates observation $\boldsymbol{y}$ to the input data vector $\boldsymbol{x}$, parameter vector α, and a random experimental error ε. If y_i $(i = 1,\ldots, r)$ is observed, respectively, when x_i $(i = 1,\ldots, r)$ occurs, then, under suitable assumptions, a least-squares regression problem can be formulated. The goal is to find a parameter vector $\bar{\alpha}$ that solves the unconstrained NLP, $\min_\alpha \Sigma_{i=1}^{r}[y_i - F(\alpha, x_i)]^2$. If F is linear, then this is called linear regression and is well known, heavily used, and has a rich statistical basis and interpretation. Other norms could be used. A similar approach, so called curve-fitting, could be used to fit a function F to a given set of points or to approximate another function, possibly introducing constraints on F, for example, requiring bounds on F or its derivatives, without necessarily being supported by the attendant statistical rationale and utilizing a variety of norms.

The problem of finding the minimum eigenvalue and associated eigenvector of a real symmetric $n \times n$ matrix $\boldsymbol{A}$ can be posed as that of determining the optimal value and solution vector, respectively, of the constrained NLP problem, min $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}$ s.t. $\|\boldsymbol{x}\|^2 = 1$, again a minimum-norm-type problem. The problem of finding the shortest distance between one point and another, or a point and a line, or a point and a set or, more generally, between one set S_1 and another set S_2 takes on the rather natural NLP-constrained form, min $\|\boldsymbol{x} - \boldsymbol{y}\|^2$ s.t. $\boldsymbol{x} \in S_1$ and $\boldsymbol{y} \in S_2$. Many extensions and ramifications of this idea can be envisioned.

Practical Applications

Hancock (1960, p. 151) stated that "by means of Gauss's principle all problems of mechanics may be reduced to problems of maxima and minima." Principles in optics, wave mechanics, quantum physics, astronomy, chemistry, biology, etc., can usually be formulated in terms of extremal (i.e., maximum or minimum) principles, for example, a path of least resistance, minimum energy, maximum entropy, etc.

The practical applications of nonlinear programming are incredibly vast. Regression and curve-fitting applications abound in mathematics and physics, the natural and applied sciences, econometrics, and engineering statistics. Generalizations include possible constraints on the parameters and extensions to higher dimensions (surface fitting), with applications in pattern recognition, geography, agriculture and quantum physics, for example (Hobson and Weinkam 1979). As early as 1980, Hillier and Lieberman (1980, Ch. 1) reported that the most widely used operations research techniques were statistical techniques (mainly those involving regression analysis), simulation, and linear programming. They noted that the most important applications of mathematical programming were those in production management (e.g., in allocation of resources to maximize some measure of profit, quality, efficiency, effectiveness, etc.), followed next by financial and investment planning, and they reported that about 25% of all scientific computation on computers was devoted to linear programming and related techniques. It seems clear that these trends have sustained. Winston (1991, p. 51) noted that 85% of the respondents of a survey of Fortune 500 firms report use of LP, and that about 40% of his book is devoted to related optimization techniques.

Practically every research and textbook in NLP discusses important current applications. Fletcher (1987, p. 4) noted important applications in structural design, scheduling, and blending, as well as numerical analysis and differential equations. McCormick (1983) analyzed problems in chemical equilibrium, inventory control, engineering design and water pollution control. Bazaraa and Shetty (1979) discussed problems in discrete and continuous optimal control, mechanical and structural design, electrical networks, and location of facilities.

A methodology for NLP started coming together around 1960. This was largely motivated by applications, for example, to petroleum refinery problems which inspired algorithmic work by Rosen in 1960–61 and a paper-pulp manufacturing process that led to a technique proposed by Carrol in 1959

and 1961. A collection of case studies in such diverse areas as bid evaluation, stratified sampling, launch vehicle design and alkylation process optimization was given by Bracken and McCormick (1968). The algorithms used were based on Carrol's interior barrier function and Courant's exterior quadratic penalty function proposed in 1943, developed and extended by Fiacco and McCormick in 1963, and implemented via the SUMT computer program by McCormick, Mylander and Fiacco in 1965 (Fiacco and McCormick 1968; Fiacco and Ishizuka 1990).

Basic Theory

Problem types can be differentiated in many ways: (i) one dimensional or many dimensional, (ii) finite-dimensional (e.g., in R^n) or infinite-dimensional (e.g., as in variational calculus and optimal control), (iii) a finite number of constraints or an infinite number (as in semi-infinite programming), (iv) unconstrained or constrained, (v) involving real numbers (standard NLP) or integers (integer programming), (vi) convex or nonconvex, (vii) smooth (i.e., differentiable) or nonsmooth (nondifferentiable), and (viii) deterministic or stochastic.

A local minimizer of (P) is a feasible point $\bar{\boldsymbol{x}}$ such that $f(\bar{\boldsymbol{x}}) \leq f(\boldsymbol{x})$ for all $\boldsymbol{x}$ in a feasible neighborhood of $\bar{\boldsymbol{x}}$. If $f(\bar{\boldsymbol{x}}) \leq f(\boldsymbol{x})$ for *all* feasible $\boldsymbol{x}$, then is called a global minimizer. If $\bar{\boldsymbol{x}}$ is a local minimizer and $f(\bar{\boldsymbol{x}}) < f(\boldsymbol{x})$ for all $\bar{\boldsymbol{x}} \neq \boldsymbol{x}$ in a feasible neighborhood of $\boldsymbol{x}$, then $\bar{\boldsymbol{x}}$ is called a strict local minimizer. If $\bar{\boldsymbol{x}}$ is the only local minimizer in some feasible neighborhood of $\bar{\boldsymbol{x}}$, it is called an isolated local minimizer.

A fundamental result of great importance is the fact that a feasible global minimizer of a continuous function f exists in the feasible region R if R is nonempty and compact, a result attributed to Weierstrass. If f is once continuously differentiable and $\bar{\boldsymbol{x}}$ is a local unconstrained minimizer, then the gradient, $\nabla f(\bar{\boldsymbol{x}}) = 0$. If f is twice continuously differentiable, then $\nabla f(\bar{\boldsymbol{x}}) = 0$ and the Hessian (matrix of second partial derivatives) $\nabla f^2(\bar{\boldsymbol{x}}) = 0$ is positive-semi-definite (p.s.d.) at a local minimizer $\bar{\boldsymbol{x}}$ and $\nabla f(\bar{\boldsymbol{x}}) = 0$ is positive definite (p.d.), then $\nabla^2 f(\bar{\boldsymbol{x}}) = 0$ is an isolated (hence, also strict) local minimizer.

The usual Lagrangian of problem P is defined as

$$L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}) = f(\boldsymbol{x}) - \sum_{i=1}^{m} u_i g_i(\boldsymbol{x}) + \sum_{j=1}^{p} w_j h_j(\boldsymbol{x})$$

where the $\{u_i\}$ and $\{w_j\}$ are the Lagrange multipliers. John in 1948, Karush in 1939, and Kuhn and Tucker in 1951 (Fiacco and McCormick 1968; Fiacco and Ishizuka 1990) independently generalized and extended the classical Lagrange multiplier rule (Lagrange 1762) for equalities to include inequalities, arriving at the following first-order conditions called the Karush-Kuhn-Tucker conditions and abbreviated as KKT $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$: there exist $\bar{u}_i \geq 0\,(i = 1, \ldots, m)$ and $\bar{w}_j (j = 1, \ldots, p)$ such that $\nabla_x L(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}}) = 0$ and $\bar{u}_i g_i(\bar{\boldsymbol{x}}) = 0\,(i = 1, \ldots, m)$, for $\bar{x}$ feasible. If a suitable constraint qualification (CQ) holds at a local minimizer $\bar{x}$, then KKT $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$ holds. Though a more general CQ was given in Kuhn and Tucker (1951), it turns out that this holds if the binding constraint gradients are linearly independent, i.e., if $\{\nabla g_i(\bar{\boldsymbol{x}}), i \in B(\bar{\boldsymbol{x}}); \nabla h_j(\bar{\boldsymbol{x}}), j = 1, \ldots, p\}$ are linearly independent, where $B = \{i : g_i(\bar{\boldsymbol{x}}) = 0\}$. Denote this CQ by LI $(\bar{\boldsymbol{x}})$. The KKT $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$ are sufficient for $\bar{x}$ to be a minimizer if (P) is a convex program, i.e., if f is convex, the $\{g_i\}$ concave and the $\{h_j\}$ affine. Convex programs have additional attributes: local solutions are global, they have associated with them a rich duality theory, and they are among the easiest to analyze and solve. Second-order optimality conditions are now also well known and heavily used.

When P is convex, a dual problem is the following:

$$\max_{(x,u,w)} L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}) \text{ s.t. } \quad \nabla_x L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}) = \boldsymbol{0},\ \boldsymbol{u} \geq \boldsymbol{0}, \quad (\mathbf{D})$$

where $\boldsymbol{u} = (u_1, \ldots, u_m)$ and $\boldsymbol{u} \geq \boldsymbol{0}$ means that $u_i \geq 0$ for all $i = 1, \ldots, m$. This simple but remarkably useful formulation was first proposed and developed by Wolfe in 1961 (Fiacco and McCormick 1968; Fiacco and Ishizuka 1990). It turns out that the optimal value of **P** is bounded below by the optimal value of **D**. Further, if $L(\bar{\boldsymbol{x}})$ holds, or one of several other well known CQs, then if $(\bar{\boldsymbol{x}})$ solves **P**, it follows that KKT $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$ holds, $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$ solves the dual **D**, and $f(\bar{\boldsymbol{x}}) = (\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$. Duality has significant computational applications; for example, algorithms that generate dual-feasible points also yield lower bounds on the primal optimal value.

Other duals have been developed, most notably the Fenchel Dual, significantly extended and utilized by Rockafellar (1970) and others. Not surprisingly, a rich and finely tuned duality theory has been developed for LP.

Algorithms

An algorithm is a numerical procedure that starting with given initial conditions, calculates a sequence of steps or iterations until some stopping rule is satisfied. Up until the emergence of interior-point methods, the uncontested winner for LP has been some version of Dantzig's Simplex Method, a technique based on the idea of moving from one vertex of the feasible region to an adjacent vertex while reducing the objective function with each move. The elegance of the mathematics, industrialization and economic planning needs, and the advent of the electronic digital computer in the 1940s, and a host of important practical applications that followed, all resulted in making LP widely accepted and heavily utilized.

The same forces that stimulated the development of LP in the latter 1940s were encouraging research on theory and algorithms for NLP. The 1930s and 1940s saw a flurry of theoretical activity in variational calculus and optimization at the University of Chicago and other mathematical centers by mathematicians like Valentine, Reid, McShane, Karush, Bliss, Graves, Hestenes, Courant, John and others. The early 1950s brought a sharper focus to first-order and second-order optimality conditions for inequality constrained NLP by Kuhn and Tucker in 1951 and Pennisi in 1953, respectively, and others. As early as 1951, for example, a paper by Arrow in 1951 on a gradient method for solving constrained saddle-point problems, there is evidence of serious algorithmic work. Two key results during this period were the conjugate direction method, an iterative procedure for solving a system of linear equations, by Hestenes and Stiefel in 1952, and a variable-metric method (a quasi-Newton algorithm wherein the required Hessian inverse is calculated iteratively) by Davidon in 1959. Such developments significantly enhanced the steepest-descent and Newton tools for solving equations, the heart of solving NLP problems, and were followed by an intense period of activity in the 1960s, with a rapid solidification of the theory and computational and methodological breakthroughs such as cutting plane algorithms by Kelley in 1960, methods of feasible directions by Zoutendijk in 1960, gradient projection methods by Rosen in 1960–61, and SUMT by Fiacco and McCormick in 1963 (Fiacco and McCormick 1968; Fiacco and Ishizuka 1990).

Many algorithms were subsequently developed for both unconstrained and constrained problems. The most popular prototype starts with a merit function (e.g., the objective function) and initial conditions, determines a search direction vector, then calculates a step in the given direction based on a line search, some one-dimensional curve fitting scheme aimed at both reducing the value of the merit function and maintaining feasibility. The process is repeated until some convergence criteria are satisfied. An algorithm for a well-posed problem generally attempts to satisfy first-order necessary optimality conditions for a local minimizer. More sophisticated algorithms satisfy second-order necessary conditions and others even seek out global minimizers, though techniques for the latter continue to be under intense development. Algorithms may be deterministic or stochastic, continuous or discrete-step, accumulate information or not, etc. A host of special-purpose algorithms have been developed for one-dimensional optimization, for example, variations of successive bisection, Newton's method, the secant method, false position, Fibonacci search and golden section (McCormick 1983).

Some of the most effective contemporary algorithms for smooth unconstrained problems are generally some variant or mixture of a quasi-Newton (approximate Newton) or conjugate direction algorithm. The survivors in the competition must fare well overall in meeting several demanding and sometimes opposing criteria: computational effort, speed of convergence, accuracy, robustness, ease of implementation, accessibility, and so on. Developing rigorous computational and theoretical standards for measuring these attributes are important, e.g., a rate of convergence theory is in place that establishes that steepest descent converges at least at a linear rate (as in a geometric series) and Newton's method at a quadratic rate (exponentially, at least quadratic), under rather ideal circumstances. Hybrid methods, conjugate directions and variable metric methods, are thought to perform adequately when they converge superlinearly (more or less, the best linear rate possible, a compromise between linear and quadratic).

Another important criterion is that an unconstrained algorithm be able to calculate the minimizer of a positive definite quadratic form in n variables, in at most n iterations. A key driving principle is exploitation of problem structure.

Some important algorithms for constrained problems are sequential linear programming (SLP), e.g., separating or cutting plane algorithms; sequential quadratic programming (SQP), e.g., constrained Newton approaches; generalized reduced gradient (GRG) methods, essentially, variable elimination simplex-type algorithms; feasible direction (constrained steepest descent) methods; projected gradient methods; and auxiliary function methods, e.g., augmented Lagrangian function (i.e., Lagrangian plus penalty term) techniques, penalty function (objective function plus constraint violation cost) and barrier function (objective function plus feasibility enforcing) methods. Algorithms and software are given in the references.

Additional important topics and suggested references are the following: global optimization, (Kan et al. 1989); parametric programming, sensitivity and stability analysis (Fiacco 1983; Fiacco 1990), discussed next; stochastic programming (Wets 1989); semi-infinite programming (SIP) (Fiacco and Kortanek 1983); multi-objective programming (Sawaragi et al. 1985); multi-level programming (Anandalingam 1992); control theory (Hocking 1991); numerical methods and implementation (Gill et al. 1981); software evaluation and comparison of algorithms (Waren et al. 1987) and (Moré and Wright 1993); parallel and large-scale programming (Rosen 1990); integer programming (Schrijver 1986); basic barrier and penalty function methodology (Fiacco and McCormick 1968; Fiacco and Ishizuka 1990); and nonsmooth optimization (Neittaanmaki 1992).

Intense activity in devising polynomial-complexity interior point methods for LP and NLP was sparked by a theoretical breakthrough with Khachian's ellipsoid method (Khachian 1979) and a theoretical-computational breakthrough with Karmarker's potential method (Karmakar 1984). The reader is referred to the excellent surveys by Gonzaga (1992) and Wright (1992) for a good introduction to this important development and for many good references, and to the books of Megiddo (1989) and Nesterov and Nemirovski (1993) for technical advances.

Software exists to implement variations of all the methods described here. As to computational capability, problems in thousand of variables and constraints can now be solved on a PC. Large problems may require parallel processors. However, a meaningful measure of computational difficulty is elusive in NLP. For example, consider that a high-degree polynomial in one variable may have many local minima and may be much more difficult to solve globally than a large convex program with hundreds of variables and constraints.

Sensitivity Analysis

The question motivating this topic can be raised in connection with almost any method of inquiry that results in a conclusion: How does the answer change when the assumptions change? The assumptions can be any conditions or data that are given and the changes can be qualitative or quantitative, controlled or uncontrolled, deterministic or stochastic, small or large, known or estimated, immediate or staged over time. The issue is inevitable and universal, since there are ever-present errors and ranges in approximation and interpretation, whether it be in carrying on a conversation, steering a car, hitting a tennis ball, or calculating expected return on investment.

In the early days of mathematics and physics, a related issue was apparently frequently raised: When is a problem well posed, i.e., when does a solution change continuously with continuous changes in the problem data? Variations on this theme must have quickly followed: When are the solution changes well behaved in some sense, e.g., when are they finite or bounded or smooth, and when are they not? Can the solution be calculated in closed form as a function of the changes in the data, or can at least bounds on the changes or rate of growth of the changes be calculated? Can any useful properties of perturbed solutions be identified or measured, e.g., whether a unique solution remains unique, whether a solution function or set is convex as a function of the changes, if a solution trajectory is differentiable, whether an assumption persists under given perturbations, etc.? At a slightly more sophisticated level, can some of these properties be calculated from information available at a solution ... without resolving the problem with new data?

A brief summary of a collection of such results is presented in the context of nonlinear programming (NLP) where there are parameters (e.g., data) present that are subject to perturbations. Here, the focus is on characterizations involving small specified changes in the parameters, the study of which is termed sensitivity analysis.

Very simple problems can be stable or unstable, for different parameter values. Consider the linear problem, minimize x_1 s.t. $x_1 \geq -1$, $x_2 \leq \varepsilon x_1$, and $x_2 \geq 0$, where $x \in R^2$ and $\varepsilon \geq 0$. If $\varepsilon > 0$, then the solution is $x(\varepsilon) = (0, 0)$ and does not change with small enough changes in ε. However, if $\varepsilon = 0$, then the solution is $\boldsymbol{x}(0) = (-1, 0)$ and this changes to $x(\varepsilon) = (0, 0)$ for arbitrarily small positive changes of ε, an extremely erratic change. The desire is to understand the causes and implications associated with such stability or instability.

Preliminaries — The parametric NLP is defined as

$$\begin{array}{l}\text{minimize} \quad f(\boldsymbol{x}, \varepsilon) \text{ subject to} \\ \{\boldsymbol{x} \in E^n : g_i(\boldsymbol{x}, \varepsilon) \geq 0,\ i = 1, \ldots, m; \\ h_j(\boldsymbol{x}, \varepsilon) = \boldsymbol{0},\ j = 1, \ldots, p)\}\end{array} \qquad \mathbf{P}(\varepsilon)$$

where $\boldsymbol{x} \in R^n$ and ε is a perturbation parameter in T, a nonempty subset of R^k. If ε is held constant, then problem $P(\varepsilon)$ is simply a realization of a standard NLP problem of the form P that was discussed at the outset. The Lagrangian associated with $P(\varepsilon)$ is defined as $L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}, \varepsilon) = f(\boldsymbol{x}, \varepsilon) - \Sigma_{i=1}^m u_i g_i(\boldsymbol{x}, \varepsilon) + \Sigma_{j=1}^p w_j h_j(\boldsymbol{x}, \varepsilon)$. The optimal-value function f^* and the optimal-solution map S if $P(\varepsilon)$ are defined as

$$f^*(\varepsilon) = \begin{cases} \inf_{R(\varepsilon)} f(\boldsymbol{x}, \varepsilon) & (\text{if } R(\varepsilon) \neq \oslash) \\ +\infty & (\text{if } R(\varepsilon) = \oslash) \end{cases}$$

and $S(\varepsilon) = \{\boldsymbol{x} \in R(\varepsilon) : f(\boldsymbol{x}, \varepsilon) = f^*(\varepsilon)\}$. The set of optimal Lagrange multipliers for a given solution $\boldsymbol{x} \in S(\varepsilon)$ is the set $\{(\boldsymbol{u}, \boldsymbol{w})$: KKT$(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})$ holds$\}$.

The directional derivative of the function f^* at the point ε in the direction z is defined as

$$D_z f^*(\varepsilon) = \lim_{\alpha \to 0^+} \frac{f^*(\varepsilon + \alpha z) - f^*(\varepsilon)}{\alpha}$$

if the limit exits.

The problem $P(\varepsilon)$ is said to be convex in $\boldsymbol{x}$ if f and the $-g_i$ are convex in $\boldsymbol{x}$ and the h_j are affine in $\boldsymbol{x}$ for each fixed $\varepsilon \in T$, and jointly convex if these functions have the respective properties in $(\boldsymbol{x}, \varepsilon)$ and T is a convex set. Assume that the functions defining problem $P(\varepsilon)$ are continuous jointly in $(\boldsymbol{x}, \varepsilon)$ in the sequel.

Some Basic Theoretical Results — The following conditions are used, which may hold at a feasible point x for some parameter value ε. Differentiability is assumed as needed, at $(\boldsymbol{x}, \varepsilon)$.

(a) The Karush-Kuhn-Tucker conditions, as before, designated KKT($\boldsymbol{x}$, $\boldsymbol{u}$, $\boldsymbol{w}$): there exist $u_i \geq 0$ ($i = 1, \ldots, m$) and w_j ($j = 1, \ldots, p$) such that $\nabla_x L(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}, \varepsilon) = 0$, $u_i g_i(\mathbf{x}, \varepsilon) = 0 (i = 1, \ldots, m)$, and $h_j(\boldsymbol{x}, \varepsilon) = 0 (j = 1, \ldots, p)$;

(b) Linear Independence, as before, designated LI(x) : $\nabla_x g_i(\boldsymbol{x}, \varepsilon) (i \in B(x, \varepsilon))$, $\nabla_x h_j(\mathbf{x}, \varepsilon)$ $(j = 1, \ldots, p)$ are linearly independent, where $B(\mathbf{x}, \varepsilon) = \{i : g_j(\boldsymbol{x}, \varepsilon) = 0\}$;

(c) Strict Complementary Slackness, designated SCS$(x) : u_i > 0 (i \in B(\boldsymbol{x}, \varepsilon))$;

(d) The Mangasarian-Fromovitz Constraint Qualification, designated MFCQ(x) : (i) $\nabla_x h_j(\boldsymbol{x}, \varepsilon) (j = 1, \ldots, p)$ are linearly independent and there exists z such that $\nabla_x g_i(\boldsymbol{x}, \varepsilon) z > 0 (i \in B(\boldsymbol{x}, \varepsilon))$ and $\nabla_x h_j(\boldsymbol{x}, \varepsilon) z = 0 (j = 1, \ldots, p)$;

(e) The Second-Order Sufficient Condition, designated SOSC$(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}) : z^{\mathrm{T}} \nabla_{x^2} \mathrm{L}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}, \varepsilon) z > 0$ for all $z \neq 0$ such that $\nabla_x g_i(\boldsymbol{x}, \varepsilon) z \geq 0$ $(i \in B(\boldsymbol{x}, \varepsilon)), \nabla_x g_i(\mathbf{x}, \varepsilon) z = 0 (i \in D(\boldsymbol{x}, \varepsilon) = \{i \in B(\mathbf{x}, \varepsilon) : u_i > 0\})$, and $\nabla_x h_j(\boldsymbol{x}, \varepsilon) z = 0 (j = 1, \ldots, p)$, for some $(\boldsymbol{u}, \boldsymbol{w})$ such that KKT$(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})$ holds;

Known facts relevant to this brief overview are the following. The SOSC $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$ implies that $\bar{\boldsymbol{x}}$ is a strict local minimizer (i.e., the unique global minimizer in some feasible neighborhood of $\bar{\boldsymbol{x}}$) of P with optimal Lagrange multipliers $(\bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$. The condition MFCQ$(\bar{\boldsymbol{x}})$ holds at a local solution $\bar{x}$ if and only if the set of $(\boldsymbol{u}, \boldsymbol{w})$ satisfying KKT $(\bar{\boldsymbol{x}}, \boldsymbol{u}, \boldsymbol{w})$ is nonempty, compact and convex. If $LI(\bar{\boldsymbol{x}})$ holds at a local solution $\bar{\boldsymbol{x}}$, then there exists a unique $(\bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$ satisfying KKT $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$.

Using these and other well-known facts, some of the important results that hold for problem $P(\varepsilon)$:

(i) If all f, g_i, h_j are once differentiable in $(\boldsymbol{x}, \varepsilon)$, $R(\bar{\varepsilon}) \neq \emptyset$, $R(\varepsilon)$ is contained in a compact set for ε

near $\bar{\varepsilon}$, and $MFCQ(\bar{\boldsymbol{x}})$ holds for some $\bar{\boldsymbol{x}} \in S(\bar{\varepsilon})$, then $f^* \in C$ it $\varepsilon = \bar{\varepsilon}$.

(ii) If problem $P(\varepsilon)$ is jointly convex, then f^* is convex on T. If f is concave in ε and R does not depend on ε, then $f^*>$ is concave on T. Global parametric optimal value bounds can readily be calculated at a solution point when f^* is convex or concave and optimal Lagrange multipliers exist and are known. Also, when f^* is convex or concave, it follows from well-known results that f^* is continuous in the interior of T.

(iii) If $R(\varepsilon) \neq \emptyset$ and is compact and independent of ε and f and $\nabla_\varepsilon f$ are jointly continuous in $(\boldsymbol{x}, \varepsilon)$, then $D_z f^*(\varepsilon) = \min \nabla_\varepsilon f(\boldsymbol{x}, \varepsilon) z \,\text{s.t.}\, \boldsymbol{x} \in \mathrm{S}(\varepsilon)$.

(iv) If f, g_i, h_j are twice continuously differentiable in $(\boldsymbol{x}, \varepsilon)$ and KKT, SOSC $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \bar{\boldsymbol{w}})$, $\mathrm{LI}(\boldsymbol{x})$ and $\mathrm{SCS}(\bar{\boldsymbol{x}})$ hold at $\varepsilon = \bar{\varepsilon}$, then $(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})$ is locally unique and once continuously differentiable as a function of ε, such that the assumptions persist to hold at $(\boldsymbol{x}(\varepsilon), \boldsymbol{u}(\varepsilon), \boldsymbol{w}(\varepsilon))$, near $\bar{\varepsilon}, f^*$ is twice continuously differentiable where $f^*(\varepsilon) = f[\boldsymbol{x}(\varepsilon), \varepsilon]$ and $\nabla_\varepsilon f^*(\varepsilon) = \nabla_\varepsilon L[\boldsymbol{x}(\varepsilon), \boldsymbol{u}(\varepsilon), \boldsymbol{w}(\varepsilon), \varepsilon]$ near $\varepsilon = \bar{\varepsilon}$. Thus, $\mathbf{x}(\varepsilon)$ is an isolated (i.e., locally unique) and hence also strict local minimizer and $[\boldsymbol{u}(\varepsilon), \boldsymbol{w}(\varepsilon)]$ is unique. Strengthening SOSC by relaxing the restriction that $\nabla_x g_i(\boldsymbol{x}, \varepsilon) z \geq 0 (i \in B(\boldsymbol{x}, \varepsilon))$ and dropping SCS, $\boldsymbol{x}(\varepsilon)$ is an isolated local minimizer with unique $[\boldsymbol{u}(\varepsilon), \boldsymbol{w}(\varepsilon)]$ since the assumptions again persist near $\varepsilon = \bar{\varepsilon}$ at $[\boldsymbol{x}(\varepsilon), \boldsymbol{u}(\varepsilon), \boldsymbol{w}(\varepsilon)]$ locally unique, but now $(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})$ is not once continuously differentiable in ε but only directionally differentiable, with f^* once continuously differentiable and $\nabla_\varepsilon f^* = \nabla_\varepsilon L$ is before. Relaxing LI to MFCQ and further strengthening SOSC as above and assuming this holds for all $(\boldsymbol{u}, \boldsymbol{w})$ in the set of optimal multipliers, the assumptions again locally persist, although now the Lagrange multipliers are not unique but are known to form a nonempty compact convex set, $\boldsymbol{x}(\varepsilon)$ is a locally isolated minimizer as before and is known to be at least continuous and f^* is only directionally differentiable. It may also interest the reader to know that KKT, SOSC, LI and SCS are satisfied at the (unique vertex) solution of a nondegenerate LP problem.

Extensions and Future Research — With additional problem structure, more analytic results follow. For example, a fairly highly developed post-optimality sensitivity analysis is known and extensively used in linear programming, including parametric expressions for local solution changes and error bounds. Likewise, more can be said about unconstrained minimization, right-hand-side perturbations in the constraints, separable programs, geometric programs, etc. Closed-form formulas or detailed characterizations have been given for optimal value, solution point and Lagrange multiplier parameter derivatives or directional derivatives when these exist, in addition to those noted in the last section.

Extensions of the kind of results indicated have been developed for problems in more general spaces or with less structure, for example, utilizing weaker constraint qualifications, involving an infinite number of variables or constraints such as in control theory and semi-infinite programming, multiobjective optimization, integer programming, and stochastic programming. Further generalization of structure leads to variational inequalities, equilibrium problems and, at a more abstract level, generalized equations, for which a sophisticated parameter perturbation theory exists, specializations of which yield deep results for NLP. Qualitative extensions include significant additional more general convexity and concavity characterizations of the optimal value function for generalized convexity or concavity assumptions on the problem functions, more general optimal value derivative measures such as the Clarke generalized derivative, and other solution continuity concepts such as Holder continuity. Other significant extensions are those involving other (than parametric) classes of perturbations, for example, functional perturbations or abstract set-theoretic perturbations. A considerable literature exists on variations of all these ideas.

Two more research directions must be mentioned: the approximation of sensitivity information from information available as an algorithm makes progress towards a solution; and measurement of the effect of perturbations on the convergence and rate of convergence of solution algorithms. A solid basis for algorithmic approximation, the first topic, has been developed for barrier and penalty methods, but little

else. Some results on the latter topic are known for a few standard algorithms.

Applications — All the results mentioned have significant theoretical and practical applications. Perhaps one of the most obvious is in extrapolating from a solution with given data to a solution with perturbed data. Another is the approximation of the change in the optimal value resulting from perturbations of the constraints, a measure directly related to the associated optimal Lagrange multipliers (shadow prices in linear programming) which in turn are involved in duality relationships. Applications exist in decomposition, min-max problems, bilevel and multilevel programming, semi-infinite programming, implicit function optimization and other areas where optimal value functions of subproblems are encountered and certain variables are viewed as a function of others that are treated as parameters during a given iteration. Sensitivity analysis results provide valuable inputs to parametric programming, where one endeavors to approximate a solution over a finite range or given set of parameter values.

Post-optimality sensitivity analysis for linear programming is a standard option in many commercial packages and is heavily used in practice. The potential applications of NLP sensitivity analysis are even more vast. NLP computational implementations on practical problems have been extremely limited, sporadic and ad hoc, largely experimental, and applied only to a few well-structured models. A number of experiments have been conducted on geometric programs, for example. Some of the other models and parameters for which a variety of sensitivity results have been generated are stream water pollution with maximum allowable dissolved oxygen deficit and on the order of 70 other parameters perturbed, a continuous review multi-item inventory model with several parameters such as item unit cost and the standard deviation of the lead-time demand, the structural design of a vertically corrugated transverse bulkhead of an oil tanker with many design parameters, portfolio analysis with parameters affecting risk and expected return on investments, and a power system energy model requiring the development of a turbine exhaust annulus and condenser system design with objective and constraint function parameter changes. Sensitivity information was calculated by SENSUMT, a computer code developed in 1973 by Fiacco, Armacost and Mylander (see Fiacco 1983), using barrier function approximations. SENSUMT is apparently the first code to offer sensitivity analysis for NLP as a user option.

Notes and Literature — Most of the theoretical results presented here on SA can be found in Fiacco (1983), particularly in the survey given in Chapter 2. Much has been done elsewhere and since 1983, but the focus here has been on a nucleus of early basic results that provides a good profile of the variety of qualitative and quantitative sensitivity measurements. Other directly relevant surveys are Fiacco and Hutzler (1982), Fiacco and Kyparisis (1992), and Fiacco and Ishizuka (1990). For a compendium on the state of the art of sensitivity and stability analysis in variational inequalities, and stochastic, semi-infinite, integer, non-linear, geometric, linear and multi-objective programming with parameters, including results on continuity, differentiability, bounds, algorithmic perturbation results and continuation and parametric methods, the reader is referred to the collection of tutorials edited by Fiacco (1990). Hundreds of references are given to significant current work, including numerous references to other important areas such as generalized equations, curve-following techniques, multi-level programming and other topics mentioned in this article and beyond. Recent books in sensitivity analysis and related topics are those by Jongen et al. (1986) on parametric results; Brosowski (1982) on semi-infinite optimization; Brosowski and Deutsch (1985) on approximation; Guddat et al. (1987) on parametric optimization; Fiacco (1984) on a wide variety of topics; Bank et al. (1982) on continuity results in particular and nonlinear parametric optimization in general; Dontchev and Zolezzi (1993) on well-posed optimization; and Levitin (1993) for a unified general perturbation theory.

See

- ▶ Barrier Functions and their Modifications
- ▶ Convex Optimization
- ▶ Integer and Combinatorial Optimization
- ▶ Interior-Point Methods for Conic-Linear Optimization
- ▶ Linear Programming

▶ Parametric Programming
▶ Regression Analysis
▶ Unconstrained Optimization

References

Anandalingam, G., (Ed.) (1992). Hierarchical Optimization, special issue of Annals of Operations Research 34, J.C. Baltzer, Basel, Switzerland.

Bank, B., Guddat, J., Klatte, D., Kummer, B., & Tammer, K. (1982). *Nonlinear parametric optimization*. Berlin: Akademie-Verlag.

Bazaraa, M. S., & Shetty, C. M. (1979). *Nonlinear programming theory and algorithms*. New York: Wiley.

Bracken, J., & McCormick, G. P. (1968). *Selected applications of nonlinear programming*. New York: Wiley.

Brosowski, B. (1982). *Parametric semi-infinite optimization*. Frankfurt am Main: Verlag Peter Lang.

Brosowski, B., & Deutsch, F. (Eds.), (1985). *Parametric optimization and approximation*, Birkhauser.

Dontchev, A. L., & Zolezzi, T. (1993). *Well-posed optimization problems*. Berlin Heidelberg: Springer.

Fiacco, A. V. (1983). *Introduction to sensitivity and stability analysis in nonlinear programming*. New York: Academic.

Fiacco, A. V. (Ed.). (1984). *Sensitivity, stability and parametric analysis, mathematical programming study 21*. North-Holland: Elsevier Science Publishers B.V.

Fiacco, A. V., (Ed.) (1990). *Optimization with data perturbations*, *special issue of Annals of Operations Research* 27, J.C. Baltzer, Basel, Switzerland.

Fiacco, A. V., & Hutzler, W. P. (1982). Basic results in the development of sensitivity and stability analysis in nonlinear programming. In J. E. Falk & A. V. Fiacco (Eds.), *Mathematical programming with parameters and multi-level constraints* (Computers and operations research, Vol. 9, pp. 9–28). New York: Pergamon.

Fiacco, A. V., & Ishizuka, Y. (1990). Sensitivity and stability analysis for nonlinear programming. *Annals of Operations Research, 27*, 215–235.

Fiacco, A. V., & Kortanek, K. O. (Eds.). (1983). *Semi-Infinite Programming and Applications* (Lecture Notes in Economics and Mathematical Systems, Number 215). Berlin: Springer.

Fiacco, A. V., & Kyparisis, J. (1992). A tutorial on parametric nonlinear programming sensitivity and stability analysis. In F. Y. Phillips & J. J. Rousseau (Eds.), *Systems and management science by extremal methods: Research honoring Abraham Charnes at Age 70* (pp. 205–223). Boston: Kluwer Academic Publishers.

Fiacco, A. V., & McCormick, G. P. (1968). *Nonlinear programming, sequential unconstrained minimization techniques*. New York: Wiley. An unabridged corrected version was published by SIAM in the series Classics in Applied in Applied Mathematics (1990).

Fletcher, R. (1987). *Practical methods of optimization*. New York: Wiley.

Gill, P. E., Murray, W., & Wright, M. H. (1981). *Practical optimization*. London: Academic.

Gonzaga, C. C. (1992). Path following methods for linear programming. *SIAM Review, 34*, 167–224.

Guddat, J., Jongen, H. T., Kummer, B., & Nozicka, F. (Eds.). (1987). *Parametric optimization and related topics*. Berlin: Akademie-Verlag.

Hancock, H. (1960). *Theory of maxima and minima*. New York: Dover.

Hillier, F. S., & Lieberman, G. J. (1980). *Introduction to operations research*. Oakland, CA: Holden-Day.

Hobson, R. F., & Weinkam, J. J. (1979). Curve Fitting. In A. G. Holzman (Ed.), *Operations research support methodology* (pp. 335–362). New York: Marcel-Dekker.

Hocking, L. M. (1991). *Optimal control: An introduction to the theory with applications*. New York: Oxford University Press.

Jongen, H. T., Jonker, P., & Twilt, F. (1986). *Non-linear optimization in Rn: II. Transversality, flows, parametric aspects*. New York: Verlag Peter Lang.

Kan, A. H., Rinnooy, G., & Timmer, G. T. (1989). Global Optimization, Ch. IX. In G. L. Nemhauser et al. (Eds.), *Handbooks in OR & MS* (Vol. 1). Amsterdam: North-Holland.

Karmakar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatoria, 4*, 373–395.

Khachian, L. G. (1979). A polynomial algorithm in linear programming. *Soviet Mathematics Doklady, 20*, 191–94.

Kuhn, H. W. (1991). Nonlinear programming: A historical note. In J. K. Lenstra, A. H. G. Rinnooy Kan, & A. Schrijver (Eds.), *History of mathematical programming* (pp. 82–96). Amsterdam: North-Holland.

Kuhn, H. W., & Tucker, A. W. (1951). Nonlinear programming. In J. Neyman (Ed.), *Proceedings of the second Berkeley symposium on mathematical statistics and probability* (pp. 481–493). Berkeley: University of California Press.

Lagrange, J. L. (1762). Essai sur une Nouvelle Methode pour Determiner les Maxima et Minima des Formules Integrales Indefinies. *Miscellanea Taurinensia, II*, 173–195.

Levitin, E. S. (1993). *Perturbation theory in mathematical programming*. New York: Wiley-Interscience.

McCormick, G. P. (1983). *Nonlinear programming: Theory algorithms and applications*. New York: Wiley.

Megiddo, N. (Ed.). (1989). *Progress in mathematical programming — Interior point and related methods*. New York: Springer.

Moré, J. J., & Wright, S. J. (1993). *Optimization software guide, frontiers in applied mathematics 14*. Philadelphia: SIAM.

Neittaanmaki, M. (1992). *Nonsmooth optimization*. London: World Scientific Publishing.

Nesterov, Y., & Nemirovski, A. (1993). *Interior-point polynomial algorithms in convex programming, studies in applied mathematics 13*. Philadelphia: SIAM.

Rockafellar, R. T. (1970). *Convex analysis*. Princeton, NJ: Princeton University Press.

Rosen, J. B. (Ed.). (1990). *Supercomputers and large-scale optimization: Algorithms, software, applications, special issue of Annals of Operations Research 22(1)–(4)*. Basel, Switzerland: J.C Baltzer.

Sawaragi, Y., Nakayama, H., & Tanino, T. (1985). *Theory of multiobjective optimization*. New York: Academic.

Schrijver, A. (1986). *Theory of linear and integer programming*. New York: Wiley.

Waren, A. D., Hung, M. S., & Lasdon, L. S. (1987). The status of nonlinear programming software: An update. *Operations Research, 35*, 489–503.
Wets, R. J.-B. (1989). Stochastic programming. In *Handbooks in OR & MS* (Vol. 1). Amsterdam: Elsevier/North-Holland.
Winston, W. L. (1991). *Operations research: Applications and algorithms*. Boston: PWI-Kent Publishing.
Wright, M. H. (1992). Interior methods for constrained optimization. In A. Iserles (Ed.), *Acta Numerica* (Vol. 1, pp. 341–407). New York: Cambridge University Press.

Nonnegative Solution

A solution to a problem in which all variables $x_j \geq 0$.

Nonnegativity Conditions

A restriction that limits a variable or a set of variables to be either zero or positive. The set of conditions $x_j \geq 0$ ($j = 1,\ldots, n$) are the usual nonnegativity conditions that apply to the variables of a linear-programming problem.

Nonsingular Matrix

A square matrix that has an inverse. A nonsingular matrix has a nonzero value for its determinant.

See

► Matrices and Matrix Algebra

Nontrivial Solution

For the set of homogeneous linear equations $\boldsymbol{Ax} = \boldsymbol{0}$, a solution $\boldsymbol{x} \neq \boldsymbol{0}$.

See

► Null Space
► Trivial Solution

Nonzero-Sum Game

A game in which the payoffs p_i to the players do not sum to zero. Here the payoff to player i is positive if it is a win and negative if it is a loss.

See

► Game Theory
► Payoff Matrix
► Zero-Sum Game

Non-Archimedean Number

A number that does not satisfy the Archimedean axiom. Such numbers arise in setting preemptive (lexicographic) priorities in goal programming, the "Big M" for finding a feasible basis to a linear-programming problem, and in selecting an infintesimal in data envelopment analysis.

See

► Archimedean Axiom
► Big M Method
► Data Envelopment Analysis
► Goal Programming

Non-Compensatory Choice Strategies

Not employing trade-offs between the dimensions of choice alternatives but, using thresholds (or cutoffs) that need to be achieved for choice of an alternative.

See

► Choice Theory

Non-Preemptive

Concept having to do with how priorities are treated. In queueing models, it refers to a queue discipline that does not allow a customer who has already started

service to be interrupted (preempted) when a customer with higher priority arrives. In goal programming, it has to do with a priority ranking that orders the systematic optimization of the deviation variables.

See

- ▶ Goal Programming
- ▶ Queueing Theory

Non-uniform Random Variates

- ▶ Random Number Generators
- ▶ Random Variates
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Variance Reduction Techniques in Monte Carlo Methods

Normative Model

A model that attempts to describe standards of behavior of a man/machine system; the "what ought to be." Normative models identify feasible and desirable configurations of the system to serve as goals or norms. For a decision problem, such a model specifies logically consistent decision procedures that indicate how an individual should decide. Normative models are often based on an axiomatic foundation.

See

- ▶ Decision Problem
- ▶ Descriptive Model
- ▶ Mathematical Model
- ▶ Prescriptive Model

Northwest-Corner Solution

A procedure for finding a basic feasible solution to a transportation problem. For a problem with m origins and n destinations, the approach is to form a matrix with m rows and n columns, where a cell (i, j) of the matrix represents the shipment of goods from origin i to destination j. The algorithm starts with all shipments zero and first assigns the maximum shipment possible to the most northwest cell $(i = 1, j = 1)$. Each time an allocation is made, either a row or column of the matrix is crossed out. The algorithm continues to make the maximum possible shipments in the northwest corners of the reduced matrices, until the shipment is made in cell $i = m$ and $j = n$. The resulting shipments form a basic feasible solution to the underlying linear-programming problem. A degeneracy-avoiding procedure may have to be used in determining whether a row or column is to be crossed out in the intermediate steps.

See

- ▶ Transportation Problem

NP, *NP*-Complete, *NP*-Hard

- ▶ Computational Complexity

Null Matrix

A matrix with all entries equal to zero.

See

- ▶ Matrices and Matrix Algebra

Null Space

The set of solutions to the equations $\boldsymbol{Ax} = \boldsymbol{0}$ is called the null space of $\boldsymbol{A}$.

See

- ▶ Trivial Solution

Numerical Analysis

Stephen G. Nash
George Mason University, Fairfax, VA, USA

Introduction

Numerical analysis uses computation as a tool to investigate mathematical models. At its most basic, this might mean computing an answer, such as the optimal value of a linear program. Beyond this, one might want error estimates (how accurate is the optimal value computed by the algorithm) or sensitivity information (how sensitive is the optimal value to changes in the data). It might be desirable to visualize the results of the computation as a static image or – in the case of a dynamic model – as an animation. One might even wish to analyze the effects of randomness in the data. Numerical analysis can also be used as an experimental tool to reveal properties of models that may be inaccessible by analytic means.

The techniques of numerical analysis have been widely adopted. It is rare for someone to solve a linear program by hand — except perhaps in a classroom. Large-scale simulations would be all but impossible without the aid of a computer. For many people, numerical techniques have superseded analytic techniques as a tool for solving mathematical problems. There are many cases (such as when optimizing nonlinear models or solving differential equations) where no closed form analytic solution exists, but where a numerical solution is reasonable to compute. There are also cases where, even when an analytic solution is available, it is preferable to use a numerical method because it can compute the solution more efficiently and more accurately. In many areas of application, numerical analysis offers a routine, reliable, and often automated way of solving mathematical problems.

It is possible to solve many problems on standard computers, but the most challenging computational problems require high-performance computing. In moving to such highly parallel machines, it is typically necessary to use specially-adapted algorithms and software. This effort may be worthwhile in cases where no other approach is feasible.

The Impact of Computers

It makes sense to speak of numerical analysis together with the computer. Numerical analysis only developed as a separate discipline after the invention of the computer. Although computation was an important subject at earlier times, it is only with the invention of the computer that the full range of numerical analysis techniques becomes necessary. Pencil and paper calculations tend to be small scale, and are carefully supervised. There is less opportunity for accumulation of error. In addition, the precision of the calculations can be adjusted during a calculation, if that becomes necessary. On a computer, however, it is easy to perform a sequence of millions of calculations. These calculations will normally be performed at a fixed precision, without supervision. Further, algorithms that are satisfactory for small problems may not scale well to larger problems. Automatic computation carries with it both opportunities and risks. The techniques of numerical analysis attempt to exploit these opportunities while understanding and minimizing the risks involved.

There are some central questions in the study of numerical analysis. Is there an efficient algorithm to solve the given mathematical problem? How sensitive is the solution of the problem to errors in the data? How accurate is the computed solution? Can the algorithm provide an error estimate?

The most important and immediate question is whether there exists any algorithm to solve a particular problem. Currently a wide variety of numerical software is available, so for many classes of problems good methods are available. (Some sources are listed in the references.) These methods are capable of solving a great many problems that lack closed-form solutions. Even when closed-form solutions exist, the methods used in the software may be unrelated, for reasons of efficiency and accuracy. For example, the eigenvalues of a matrix will generally be calculated without forming the characteristic polynomial.

Good numerical methods, together with powerful modern computers, have made possible the routine solution of many large and difficult computational problems. Linear programs with thousands of variables pose no great challenge, for example. Although there still exist problems that strain the

most powerful computers (such as optimization models constrained by complicated partial differential equations), off-the-shelf software and desk-top computers are capable of serving many people's needs.

As software and computers have improved, the expectations of numerical software have expanded. Models have become more elaborate, and visualization of results is expected. More and more, modelers would like to incorporate uncertainty in their models. Doing this requires more sophisticated software and more powerful computers. It is now routine for computers to have multiple processors, and if software is to exploit the full capabilities of the computer, some degree of parallelism must be incorporated. In the realm of high-performance computing, computers may have more than 100,000 processors. There are significant research challenges in using these powerful machines, but they make it possible to simulate complex real-world phenomena with high fidelity.

Linear Equations

The ideas of numerical analysis are perhaps most clearly expressed in the context of solving systems of linear equations. Such a system can be written as $\boldsymbol{Ax} = \boldsymbol{b}$, where $\boldsymbol{A}$ is an $n \times n$ invertible matrix and $\boldsymbol{b}$ is the vector of right-hand side coefficients.

Methods for solving linear equations are central to numerical analysis. In particular, they form an essential component of algorithms for linear programming and other optimization problems.

The most commonly used technique for solving linear equations is Gaussian elimination. Gaussian elimination requires about n^3 arithmetic operations to solve a linear system, where n is the number of variables. On many current computers, a linear system with a few thousand variables can be solved in several seconds, and high-performance computers can solve problems with a million variables in well under a second. If the number of variables doubles, the number of arithmetic operations increases by a factor of eight.

Gaussian elimination does not compute $\boldsymbol{A}^{-1}$, and in fact there are a number of reasons why computing $\boldsymbol{A}^{-1}$ is undesirable in many circumstances (Golub and Van Loan 1996). This is especially true for large sparse problems, problems where many of the entries in the matrix $\boldsymbol{A}$ are 0. Gaussian elimination can take advantage of the presence of these zeros. Often the number of arithmetic operations required to solve such a system will be proportional to the number of nonzeros in the matrix, which in turn will often be proportional to the number of variables n. In contrast, $\boldsymbol{A}^{-1}$ may have virtually no zero entries, even when $\boldsymbol{A}$ is sparse, and computing and applying the inverse will require between $O(n^2)$ and $O(n^3)$ operations. This is one case, among many, where the mathematical solution $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ and the computer solution are calculated in different ways.

Gaussian elimination is not the only algorithm available for solving linear equations. There exist algorithms with costs proportional to n^{α} with $\alpha < 3$, but these are not widely used. There are also techniques called iterative methods that are especially effective on large sparse problems (Golub and Van Loan 1996).

Error Analysis

Suppose that one of these algorithms is applied to a system of linear equations. How accurately can the solution $\boldsymbol{x}$ be computed? It is useful to phrase this question in another way: How sensitive is the solution $\boldsymbol{x}$ to errors in the data $\boldsymbol{A}$ and $\boldsymbol{b}$? It is worrisome if small errors in the data are magnified into large errors in the solution. Such magnification can occur for two reasons. It may be because of a "bad" problem (the solution is poorly determined by the data) or because of a "bad" algorithm (an algorithm that magnifies errors in the data). If the problem is bad, it is called ill conditioned; if the algorithm is bad, it is called unstable.

If the data – either the matrix $\boldsymbol{A}$ or the right-hand side $\boldsymbol{b}$ – are subject to errors of order ε, then the relative errors in the solution $\boldsymbol{x}$ will in general be proportional to $cond\,(\boldsymbol{A})\,\varepsilon$, where $cond\,(\boldsymbol{A})$, the condition number of $\boldsymbol{A}$, is a measure of how close $\boldsymbol{A}$ is to being singular. These errors in the solution are due solely to the errors in the data; for now it is assumed that the system of equations is solved exactly. If $\boldsymbol{A}$ is singular, then $cond\,(\boldsymbol{A}) = \infty$; otherwise,

$$cond(\boldsymbol{A}) = \|\boldsymbol{A}\| \cdot \left\|\boldsymbol{A}^{-1}\right\|$$

in terms of some matrix norm $\|\cdot\|$. If, say, the Euclidean norm is used, then $cond(\boldsymbol{A}) \geq 1$ for all matrices $\boldsymbol{A}$. To illustrate this result, suppose that the data were accurate to $\varepsilon = 10^{-6}$ and that $cond\,(\boldsymbol{A}) = 10^4$; then it is expected that the relative errors in $\boldsymbol{x}$ to be proportional to 10^{-2}, that is, $\boldsymbol{x}$ would be accurate to two decimal digits. Thus, the condition number can be used as a quantitative tool to predict the accuracy of the solution to a linear system.

Even though $\boldsymbol{A}^{-1}$ is not computed by Gaussian elimination, it is still possible to estimate $cond\,(\boldsymbol{A})$ as a byproduct of the algorithm. Some additional calculations are required (about n^2 arithmetic operations), but this is much less than the n^3 operations required to solve the linear system. Thus, not only can the solution be computed, but an error estimate can be provided as well.

For most computational problems it will be possible to determine the sensitivity of the solution to errors in the data. This sensitivity can be considered as a "condition number" for that problem. If this condition number is large, it can be expected that the errors in the solution will be large, regardless of what algorithm is used to solve the problem. Of course, it will be desirable to use an algorithm that does not further magnify errors.

Computers only store numbers to a finite number of digits, often using binary arithmetic, so that just storing numbers in a computer can introduce errors in the data. For example, $1/3 = 0.333\ldots$ cannot be represented exactly with a finite number of binary digits. The precision of computer arithmetic – referred to as machine epsilon $\varepsilon_{\text{mach}}$ or unit round-off – limits the accuracy of computer calculations. Even if the data in a linear system are otherwise known exactly, when they are stored in the computer and computer arithmetic is use, the solution of the linear system can be expected to have errors proportional to $\varepsilon_{\text{mach}}$ times $cond\,(\boldsymbol{A})$. If $cond\,(\boldsymbol{A}) \approx 1/\varepsilon_{\text{mach}}$ then, from the point of view of computer arithmetic, the matrix might as well be singular.

Mathematically, a matrix is either singular or non-singular, and there are sharp differences between the two cases. Computationally it makes more sense to refer to the condition number of a matrix, and use this to measure how close a matrix is to being singular. Whether a matrix is sufficiently nonsingular to be useful will depend on the accuracy of the data and the desired accuracy of the solution. For many sorts of computational problems, the meaning of singularity or degeneracy will be blurred, with the accuracy of the solution deteriorating as the problem becomes closer to being degenerate.

So far, only errors arising from the data in the problem have been considered. The algorithm used to solve the linear system will also introduce errors. Assume that Gaussian elimination is used to solve the linear system. Gaussian elimination is unstable in its raw form, and can fail even when $\boldsymbol{A}$ is nonsingular. With minor modifications (such as the use of partial pivoting) it becomes a stable algorithm that can be applied to any nonsingular system. It can be proved that Gaussian elimination with partial pivoting computes the exact solution to a perturbed system of the form $(\boldsymbol{A} + \boldsymbol{E})\boldsymbol{x} = \boldsymbol{b}$, where $\|\boldsymbol{E}\|$ is proportional to machine epsilon times $\|\boldsymbol{A}\|$. Thus $(\boldsymbol{A} + \boldsymbol{E})$ can be interpreted as a perturbation of $\boldsymbol{A}$ where the relative errors are proportional to machine epsilon. As has been mentioned, just storing $\boldsymbol{A}$ in the computer can introduce relative errors of this magnitude. Thus the errors introduced by Gaussian elimination are comparable to the errors introduced by storing the problem on the computer. Thus Gaussian elimination is considered to be a benign algorithm.

Saying that the computed solution from Gaussian elimination is the exact solution to a perturbed problem $(\boldsymbol{A} + \boldsymbol{E})\boldsymbol{x} = \boldsymbol{b}$ represents the adoption of a distinctive point of view. For many, it will be more common to ask about the error in the computed solution. Instead the concern is how much Gaussian elimination distorts the original problem. This is a property of the algorithm. The error in the solution (or the amount by which this distortion is magnified in the solution) is a property of the data, and depends on the condition number of the matrix. This point of view isolates the effect of the algorithm on the accuracy of the solution. In this case, it shows that Gaussian elimination computes the exact solution to a "nearby" problem. This point of view is referred to as a backward error analysis.

The error analysis for linear systems is particularly elegant. For other computational problems the error analysis may not be so favorable (the computed solution may exactly solve a perturbed problem where the perturbations are large), or a backward error analysis may not be possible. In the latter case, other techniques must be used to assess the stability of an algorithm.

Concluding Remarks

The above comments illustrate some of the major questions of numerical analysis. In some settings additional questions arise. For example, the linear system might be obtained by discretizing a differential equation, i.e., by approximating a continuous function using its values at finitely many points. Then it is natural to ask how accurately the solution of the linear system approximates the solution of the original continuous problem. In addition, it is desirable that the discrete solution converge to the continuous solution as the size of the finite-dimensional problem increases.

When implementing an algorithm in software, the ultimate goal is to try to produce software that can efficiently compute a solution to full accuracy whenever the data and the solution consist of numbers that can be stored on the computer, and to design the software so that it works reliably on as large a collection of computers as possible. This goal can be difficult to achieve. Even seemingly innocuous tasks, such as computing the Euclidean norm of a vector, can require great care when the components of the vector are pathologically large or small, near the limits of computer arithmetic.

Numerical analysts continually try to solve ever larger and more difficult computational problems. This has often meant turning to parallel computers, computers capable of carrying on multiple computations simultaneously. This has led to further questions. Can an efficient parallel algorithm be found to solve the problem? Is the algorithm scalable, i.e., does it continue to perform well as the problem size and the number of processors increase? How does the parallel algorithm compare to the best scalar, or non-parallel, algorithm? Because of the variety of parallel computers available, the answers to these questions can vary from machine to machine, making it ever more difficult to design effective algorithms and software.

There is a vast literature on numerical analysis. General introductions to the topic can be found in Heath (2002); O'Leary (2009); Press et al. (2007), and Sauer (2006). An extensive discussion of numerical linear algebra is given in Golub and Van Loan (1996). A large online collection of software is described in Grosse (1994), the repository is available online at the Netliib Web site. Reviews of software for operations research are regularly published in *OR/MS Today*; the latest reviews are available online at the INFORMS Web site. The issues involved in developing software for linear algebra computations are mentioned in Anderson et al. (1999). An extensive discussion of parallel computing can be found in Dongarra et al. (2003). Examples of projects that use high-performance computing can be found, for example, at the Web site for the National Center for Supercomputer Applications.

See

- ▶ Gaussian Elimination
- ▶ Linear Programming
- ▶ Matrices and Matrix Algebra

References

Anderson, E., et al. (1999). *LAPACK users' guide* (3rd ed.). Philadelphia: SIAM.

Dongarra, J., et al. (2003). *The sourcebook of parallel computing*. San Francisco: Morgan Kaufmann.

Golub, G. H., & Van Loan, C. F. (1996). *Matrix computations* (3rd ed.). Baltimore: The Johns Hopkins University Press.

Grosse, E. (1994). Netlib joins the world wide web. *SIAM News, 27*(5), 1–3.

Heath, M. T. (2002). *Scientific computing: An introductory survey* (2nd ed.). New York: McGraw-Hill.

O'Leary, D. P. (2009). *Scientific computing with case studies*. Philadelphia: SIAM.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). *Numerical recipes: The art of scientific computing* (3rd ed.). England: Cambridge University Press.

Sauer, T. (2006). *Numerical analysis*. Boston: Pearson Addison Wesley.

O

O, o Notation

O means "order of" and o means "of lower order than." If $\{u_n\}$ and $\{v_n\}$ are two sequences such that $|u_n/v_n| < K$ for sufficiently large n, where K is a constant independent of n, then $u_n = O(v_n)$; for example, $(2n - 1)/(n^2 + 1) = O(1/n)$. The symbol O (colloquially called "big O") also extends to the case of functions of a continuous variable; for example, $(x + 1) = O(x)$. $O(1)$ denotes any function that is defined for all values of x sufficiently large, and which either has a finite limit as x tends to infinity, or at least for all sufficiently large values of x remains less in absolute value than some fixed bound; for example, $\sin x = O(1)$.

If $\lim_{n->\infty} u_n/v_n = 0$, then $u_n = o(v_n)$ (colloquially called "little o"); for example, $\log n = o(n)$, where again the notation extends to functions of a continuous variable; for example, $\sin x = o(x)$. Furthermore, $u_n = o(1)$ means that u_n tends to 0 as n tends to infinity; for example, $(\log n)/n = o(1)$. In probability modeling (e.g., Markov chains and queueing theory), it is common to see $o(\Delta t)$ used to represent functions going to 0 faster than a small increment of time Δt, i.e., $\lim_{\Delta t->0}[o(\Delta t)/\Delta t] = 0$.

Objective Function

The mathematical expression that is to be optimized (maximized or minimized) in an optimization problem.

See

- ▶ Measure of Effectiveness (MOE)
- ▶ Optimality Criteria

Object-Oriented Database

- ▶ Information Systems and Database Design in OR/MS

OEG

Operations Evaluation Group.

See

- ▶ Center for Naval Analyses

Offered Load

The ratio of mean service time to mean interarrival time; the rate at which work is brought to a queueing system.

See

- ▶ Erlang
- ▶ Queueing Theory

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Open Network

A queueing network in which all customers enter and eventually leave the network, i.e., the routing process contains no closed subsets of states for any type of customer.

See

- ▶ Closed Network
- ▶ Mixed Network
- ▶ Networks of Queues
- ▶ Queueing Theory

Open-Source Software and the Computational Infrastructure for Operations Research (COIN-OR)

Matthew J. Saltzman
Clemson University, Clemson, SC, USA
The COIN–OR Foundation, Inc., Towson, MD, USA

Introduction

Algorithms are methods developed by mathematicians, scientists, and engineers for manipulating data to provide insight and solutions to problems in theoretical and applied fields. Computer software is a vehicle for realizing algorithmic ideas. Open-source software is a vehicle for sharing those ideas that complements archival journal publications and other means of knowledge transfer.

This article explains the ideas and principles behind open-source software, how it works in practice, and its benefits and costs. It also describes a number of open-source tools and resources available for operations research and management science researchers and practitioners. The premier publisher of open-source software for operations research is the COIN-OR initiative. This article describes the COIN-OR initiative, its history, and the impact of open-source software and COIN-OR on the field of operations research.

Section "Open Source: What and Why" describes the concept of open-source software, how it started, and how its impact has grown. It also discusses the relationship between open-source software and academic research. Section "How Open Source Works" describes open source licenses and the legal framework that supports them. The broad classes of open source licenses are described along with features of licenses that are common in operations research software. Section "Open Source in Operations Research" lists a wide variety of open-source tools available for operations researchers. The largest collection is the Computational Infrastructure for Operations Research (COIN-OR). The COIN-OR initiative is described and the available projects are enumerated. In addition, several other open-source resources for OR are listed.

A note on references. Several of the documents and resources referred to in this article are available only on the World Wide Web. The associated URLs are not cited in the *Encyclopedia of ORMS* because they are subject to uncontrolled change. The URLs can be located via most Internet search engines by searching on relevant terms, such as author, title, organization, or keywords.

Open Source: What and Why

Computers and software are indispensable tools for operations researchers, whether in academia, government, the military, or industry. Aside from standard software such as operating systems, office tools, and business management tools, OR practitioners need software to run simulations, solve optimization problems, and manage and analyze data. The technology inside those tools is developed by those same practitioners, by industrial software houses, and by academic researchers. Those developers need tools to create and manage their software libraries.

Proprietary software to fill these needs may be of high quality, but it is often costly to industrial customers and may come with restrictions on its use and redistribution that can be problematic. For researchers and developers, software created in the course of research may be abandoned after the project is completed. Researchers trying to follow up on computational work by others may have to redo the

earlier work from scratch, based only on sketchy descriptions of implementation details in articles or monographs. A principal tenet of scientific research—reproducibility of results—is undermined and extension of results of computational research is impeded by lack of access to the original source code.

The concepts of free and open-source software bear on these concerns. Free and open-source software leverages intellectual property law to support:

- software source code as a publication medium for algorithmic ideas,
- an environment for collaborative research and development of software, and
- a cooperative community of developers and users of software.

Users and developers of open-source software are guaranteed access to the source code. They are granted the right to read the code, modify it, and distribute their modifications. As a result, user and developer communities often grow around open-source packages, improving and extending them to better meet the communities' needs.

In the Beginning

The advent of commercially available digital computers in the 1950s began a fundamental transformation of the way research and business are done. Data collection and manipulation that was previously impossible to accomplish by hand became routine. As a result, the sophistication of simulations, data analyses, and decision making increased markedly. Computing power has increased and cost decreased exponentially over the intervening decades.

The availability of these machines sparked a strong interest in the development of algorithms that could efficiently carry out the needed analyses and software systems (including programming languages) that could express those algorithms precisely and translate them into instructions that the machines could carry out. Significant parts of computer science and operations research are devoted principally to developing new, more efficient, more powerful algorithms that take advantage of high-powered computing systems.

It is a part of folk history that, in the early days of computing, the knowledge being developed was widely and freely shared, both in literature and in code. Sharing helped to advance rapidly the limits of knowledge in the field. But as the commercial computing industry grew, vendors saw value in keeping their software proprietary and began selling it as add-ons to the computer systems they were marketing. An open letter to members of the microcomputer programming community (then primarily the province of hobbyists) by Microsoft founder Bill Gates 1976 famously criticized the culture of sharing that had dominated the computing community up to that point. Over time, the industry transitioned to the model familiar to those who lived and worked through the early days of the personal computer, where operating systems, compilers, office tools, and other software were primarily available only at a (sometimes significant) cost. Vendors of large systems and peripheral devices adopted the same proprietary conventions.

In the meantime, the academic research culture in operations research and computer science evolved into the familiar culture of today, where the primary means of disseminating knowledge are archival journal articles, peer-reviewed conference proceedings, and research monographs.

GNU and Linux

In the mid-1980s, Richard Stallman, a researcher at MIT's Lincoln Laboratories, became frustrated with one vendor's lack of response to a bug in a printer driver and the vendor's restrictions on access to the software. His response was to launch an effort to create software that could be freely shared and rules for distributing the software that would ensure that recipients of the shared code could not lock up the code in proprietary systems. The result of Stallman's efforts was the Free Software Foundation and the GNU (GNU's Not Unix) project to create a freely sharable version of the Unix operating system and accompanying software development toolchain. Stallman referred to code distributed under these rules as "free software." He was careful, however, to distinguish the idea of "freely sharable" from "free of charge." (Stallman describes the distinction as, "free as in 'free speech,' not as in 'free beer' (Stallman and Gay 2002).") The distinction between "free" and "open source" is addressed in Section "Open Source: What and Why."

The GNU project's efforts through the 1980s resulted in a software development toolchain including the Emacs editor, the GCC compiler suite, and most of the utilities associated with the Unix

operating system, all of which were distributed under rules that kept them open and sharable. In 1991, Linus Torvalds began the Linux project to implement an operating system kernel (the layer of the OS that communicates with the hardware and manages the scheduling of software tasks) for personal computers using the Intel 80386 CPU. The Linux kernel was distributed under the same rules as the GNU software.

The Spread of Free and Open Source Software

An important consequence of the GNU and Linux efforts was that they attracted large numbers of independent programmers to work on them. The developer communities came to include hobbyists, academics, and professionals, distributed over many countries around the world. Contributors to the Linux kernel now number in the tens of thousands. Individuals and academic, government, and commercial organizations devote significant resources to open source development. Currently, one can find multiple open-source operating systems, office tools, Web browsers, database systems, games, and all manner of specialized tools and systems.

Several factors are key to the spread of open source, including:

- *Participatory communities*. The open source model encourages the evolution of participatory communities of users and developers. The Linux kernel effort boasts contributions by tens of thousands of programmers distributed all over the world. Other projects have smaller but similarly diverse communities. The contrast between this development model and the common, centralized development model of proprietary software is outlined by Raymond (2001a).
- *Rapid evolution*. A consequence of this "community development" model and a "release early, release often" strategy common in open source projects is that bugs are found and fixed and features are added quickly. Users are encouraged to interact with developers to report bugs and test fixes, and users with programming skills can fix the bugs they find and submit their patches back to the authors for incorporation into the official releases. "Given enough eyeballs, all bugs are shallow (Raymond 2001a)."
- *Low cost*. Open-source software is often available at little or no cost. Because recipients are free to redistribute copies without paying royalties, the price of copies tends to fall toward the marginal cost of distributing them.
- *Associated business models*. Perhaps counterintuitively, there are viable business models built around open-source software (see, for example, Raymond 2001b; Young and Rohm 1999). Device manufacturers distribute open-source drivers or use open-source operating systems. Experts train or consult with users of systems and packages. Companies sell systems built around open-source tools or support contracts for those tools. Companies distribute code under dual licenses, selling a proprietary license for incorporation in proprietary products and giving away a version that cannot be made proprietary.

Open Source and Academic Research

The essential mission of the university is to create and disseminate knowledge. Traditional vehicles for dissemination are journal articles and research monographs, as well as integration of knowledge into courses and textbooks. But articles and monographs have significant drawbacks as outlets for computational research, because they do not generally include the code that the authors used to generate their results. The verbal descriptions of algorithms in articles cannot include all the details necessary for another developer to exactly reproduce the original implementation. As a result (Lougee-Hiemer 2003):

- *Results are irreproducible*. Without access to the original code, other researchers cannot reproduce or verify reported results of computational experiments.
- *Comparisons are unfair*. When investigators engaged in follow-on research attempt to compare their results to earlier work, they must re-implement the earlier work. Their re-implementations can (however unintentionally) be biased in favor of the new work.
- *Models and implementations are lost*. As researchers move on to new projects, they may neglect or even discard the codes they developed for their earlier work.
- *Evolution is stunted*. Important and useful implementation techniques are unavailable to subsequent developers without access to the original code.

- *Wheels are reinvented.* Subsequent developers must rediscover important implementation techniques on their own.
- *Knowledge transfer is limited.* Implementation techniques hidden in unpublished code cannot be transferred to other problem domains.
- *Collaboration is inhibited.* Unavailability of reference implementations mean that there is no final arbiter of interpretations of standards. For example, problem instance data files can be interpreted differently by different readers. Also, commercial solver library interfaces all differ, creating vendor lock-in for embedded solvers and inhibiting sharing of code that calls those libraries.

Open-source software is a highly effective vehicle for technology transfer because it reduces to practice every detail of computational research results. It addresses the above problems by providing a path to publishing source code that mirrors the open literature for theory in important ways, such as providing a vehicle for peer review. It also addresses publication issues that are peculiar to software—particularly the need for a living document that can be updated and incorporated into new research that builds on the original code.

Software publishing as a scholarly activity is still not widely recognized in academic circles because publication venues are not well established and the mechanisms for peer review are different from those of archival journals, conference proceedings, and monographs. Hafer and Kirkpatrick (2009) propose ways to integrate software publication into the academic promotion and tenure review process. If reviewers would heed their recommendations, that would encourage scholars to engage in this important endeavor.

How Open Source Works

Companies often find working with open source challenging due to the vast array of licenses that accompany open-source software and the unfamiliarity of users and corporate legal departments with the principles of open source and the details of the various licenses. This section examines those principles and discusses some issues related to some particular popular licenses. *The information presented here does not constitute legal advice. The final arbiters of legal issues are the courts, and readers with legal concerns relating to open source should consult their legal advisers.*

Intellectual Property

Intellectual property (IP) is protected by two bodies of law: copyright law and patent law. These two legal concepts protect different kinds of intellectual property and provide different kinds of recourse to IP holders.

Copyright law protects creative expressions, such as writings, visual arts, musical compositions, and recorded performances. The protections include reserving to the creator or the owner of a copyrighted work the right to make and distribute copies and to create derivative works. A computer program is an expression of an algorithm in a particular computer language, authored by a particular programmer (or more than one) and owned by the author or the author's employer. Such creations are automatically protected with no action necessary on the author's part, although filing a claim can support legal actions for violation. Copyright protection is the primary legal underpinning of open source licenses.

Patent law protects ideas. The protections include reserving to the inventor or patent holder the right to prevent others from using the idea or derivative ideas in products or other inventions. In recent decades, courts have ruled that software can be patented. Software patents are controversial—particularly within the open source community—but they are a side issue with respect to the legal structure of open source licenses. Some open source licenses attach patent licenses for patented ideas that appear in the code or derived works, sometimes with conditions on licensees' enforcement of their own patent rights. The penalties for violating the restrictions are generally limited to revocation of the license to the code.

Open Source and Copyright

Open-source software licenses work legally the same way licenses for any software work. The authors or owners of the intellectual property have rights reserved to them under copyright law to specify the conditions under which the software is distributed to users.

Proprietary licenses generally specify that users must pay license fees, refrain from redistributing the software, limit the number of copies they make, refrain from reverse engineering, restrict usage to certain computers or individuals, and meet other restrictive conditions in order to receive permission to use the software.

Open source licenses use copyright law in the same way: the authors or owners reserve their rights under copyright law to determine the conditions under which users can obtain the software. However, with open source, the conditions explicitly grant users the right to use the software, to obtain the source code in human-readable and -modifiable form, to create derived works, and to redistribute those works. Redistributors and distributors of derived works may also be required by the license to distribute the works under the same license terms and to make source code available.

Open source is not the same as the public domain. Works placed in the public domain (either intentionally by the creator or owner or due to expiration of copyright or patent protection) are free to be used in any fashion by recipients. Open source is protected by copyright law and recipients are required to abide by the terms of the license under which they received the software. There are subtle legal issues associated with placing intellectual property in the public domain and using public-domain works, so if the intention is to make a software package available with no restrictions, it is best to distribute it with an explicit declaration or under a license that specifies that criterion explicitly [see, for example, Lindberg 2008, pp. 299–300].

A key to understanding open source licenses is the notion of a derived work. According to US copyright law [as cited by Rosen 2005], a derived work is "[a] work based upon one or more preexisting works, such as a translation…or any other form in which a work may be recast, transformed, or adapted (17 U.S.C. §101)." In software, derived works are usually created by making changes to the source code to fix bugs, add features, improve interoperability, etc. However, interpretation of the term is critical to the understanding of how different licenses work. As with legal concepts in general, the exact interpretation is a matter of case law. Open source licenses have, to date, faced limited scrutiny in court, so the concepts are not yet well settled.

Open Source Licenses

There are a wide variety of open source licenses. They vary in overall structure and in details. This section explains the basic principles of open source licensing, describes classes of licenses, and examines some of the ones that are common in the operations research community.

The Open Source Initiative

In the mid-1990s, some members of the open source community took on the challenge of advocating the use of open source in the business community. The result was the Open Source Initiative (OSI), which their Web site describes as "a non-profit corporation formed to educate about and advocate for the benefits of open source and to build bridges among different constituencies in the open-source community."

The Initiative published the Open Source Definition (OSD), a set of voluntary standards that licenses must comply with in order to receive OSI approval [reprinted in summary form in Rosen (2005), annotated version available online from the OSI]. The wording of some of the ten points in the OSD is somewhat unclear, but Rosen (2005) summarizes the main ideas in the standards as follows:

- Licensees are free to use open-source software for any purpose whatsoever.
- Licensees are free to make copies of open-source software and to distribute them without payment of royalties to the licensor.
- Licensees are free to create derivative works of open-source software and to distribute them without payment of royalties to the licensor.
- Licensees are free to access and use the source code of open-source software.
- Licensees are free to combine open-source and other software.

The licensor can require that copies and derived works be distributed under the same license that they were received under. Licenses with this provision are called "reciprocal licenses" and are discussed in Section "Reciprocal Licenses." Licenses without this requirement are "academic licenses" and are discussed in Section "Academic Licenses."

Broad Classes of Licenses

The variety of open source licenses can be bewildering. The OSI lists 67 approved licenses (as of this writing), although many are one-off

licenses created for particular individual products. The bulk of open source projects use a relatively small selection of these licenses. Unfortunately, even those licenses can have significantly different conditions, which may conflict if users try to redistribute code combined from multiple sources.

Notwithstanding the proliferation of licenses, they can be mapped to a few categories with broadly similar properties. Rosen (2005) identifies academic licenses and reciprocal licenses as the main categories of open source licenses applying to software. These categories are described in more detail below. He also identifies standards licenses—intended to enforce openness of standard protocols and reference implementations, and content licenses—which apply to non-software creations such as music, literature, and video.

The text of licenses not accompanied by citations in the discussion below can be found at the Web site of the Open Source Initiative.

Academic Licenses

"[S]uch licenses were originally created by academic institutions to distribute their software to the public, allow the software to be used for any purpose whatsoever with no obligation on the part of the licensee to distribute the source code of derivative works... Academic licenses create a public commons of free software, and anyone can take such software for any purpose—including for creating proprietary collective and derivative works—without having to add anything back to that commons (Rosen 2005)."

The best known example of this type of license is the Berkeley Software Distribution (BSD) license [reprinted in Rosen 2005, pp. 316–317], which allows "[r]edistribution and use in source and binary forms, with or without modification," subject to the requirement that the copyright notice be preserved and that the owner's name not be used without permission in an endorsement or promotion of a derived work. The license also includes disclaimers of warranty and liability.

Licenses in this class are easy to use. They are generally compatible with each other and with other licenses, and place few conditions on redistribution. (No open source license restricts use of the code if it is not redistributed.) While the BSD License is probably the most widely used in this class, other academic licenses in use for widely deployed software include the MIT License [reprinted in Rosen 2005, p. 319] and the Apache Licenses [reprinted in Rosen 2005, pp. 320–323].

Reciprocal Licenses

Reciprocal licenses "also allow software to be used for any purpose whatsoever, but they require the distributors of derivative works to distribute those works under the same license, including the requirement that the source code of those derivative works be published... Anyone who creates and distributes a derivative work of a work licensed under a reciprocal license must, in turn, license that derivative work under the same license. Reciprocal licenses, like academic licenses, contribute software into a public commons of free software, but they mandate that derivative works also be placed in that same commons (Rosen 2005)."

Reciprocal licenses are the major legal innovation of the free and open-source software movement. While they are effective at extending the body of work in the public commons, they are also controversial and the interactions of different licenses are complicated.

The GNU General Public Licenses. The GNU General Public License (GPL) is the original "free software" license, developed by Richard Stallman. Versions of the GPL (there are two in wide use, plus some variations) are the most common reciprocal licenses for free and open-source software.

The GPL version 2.0 (GPLv2) [reprinted in Lindberg 2008, pp. 333–340] dates from 1991. The most recent version, version 3.0 (GPLv3) [reprinted in Lindberg 2008, pp. 341–354] was released in 2007. The GNU Library (or "Lesser") General Public License (LGPL) version 2.1 (LGPLv2.1) [reprinted in Lindberg 2008, pp. 319–328] dates from 1999, and version 3.0 (LGPLv3) [reprinted in Lindberg 2008, pp. 329–332] is also from 2007. The GNU Affero Public License version 3 dates from 2007 as well. The distinctions are addressed below.

The most controversial aspect of the GPL is the scope of the reciprocity requirement. The GPL states that its terms apply to a covered Program (a "program or other work") and to "work[s] based on the Program." The definition of a "work based on the Program" broadens the definition of "derived work" in copyright law to include "a work containing the Program or a portion of it, either verbatim or with modifications." The common understanding of this principle is that it is intended to extend the coverage

of the GPL from a library of subroutines and functions covered by the GPL to programs that call subroutines or functions in the library. Thus, except under certain conditions, if a program incorporates a GPL library in such a manner, the program is considered a work based on the library and must in turn be distributed under the GPL. The legality of this interpretation has not been settled in case law and remains a point of controversy.

The LGPL differs from the GPL by explicitly exempting the calling program of a GPL library from the requirement that it also be distributed under the GPL. Version 3.0 of both licenses differ from their predecessors in their treatment of embedded programs that are incorporated into hardware devices and in their treatment of patents. The Affero license applies to programs that are accessed over a network, as contrasted with libraries that run on the same computer as their calling programs.

The Mozilla Public License. The Mozilla Public License [reprinted in Rosen 2005, pp. 351–367] is the license under which the Mozilla and Firefox Web browsers and associated software are distributed. It is a popular reciprocal license and Rosen considers it to be well thought out and constructed. While its history may be familiar to those who remember the "browser wars" of the late 1990s, it does not seem to be a common choice in the operations research community.

The IBM, Common, and Eclipse Public Licenses. Due to the influence of IBM and the COIN-OR initiative on operations research software, this family of licenses is fairly common in the OR community. The IBM Public License (IPL) was the first license in this family. The Common Public License (CPL) [reprinted in Rosen 2005, pp. 358–376] was created by IBM from the IPL so that developers other than IBM could release software under its terms.

The Eclipse Public License (EPL) was adapted from the CPL by the Eclipse Foundation, a consortium of companies involved in development of the Eclipse programmer's development environment. It is essentially the same as the CPL, except for the elimination of a "patent retaliation clause," which revoked license of any patents in the code under license for any user who sues a contributor for infringement of a software patent not related to that code. Recently, the CPL and EPL license stewards agreed to replace the CPL with the EPL, so developers and contributors to CPL projects are encouraged by the stewards of the two licenses (IBM and the Eclipse Foundation) to upgrade them to the EPL.

Open Source in Operations Research

The Computational Infrastructure for Operations Research (COIN-OR)

The COIN-OR initiative supports the development and publication of open-source software for the benefit of the operations research community. The project began as the Common Optimization INterface for Operations Research at IBM Research in 2000 (Lougee-Hiemer 2003; Saltzman 2002), a collection of four related projects (the Open Solver Interface, the Volume Algorithm, the Cut Generator Library, and the Branch-Cut-Price Framework) and two independent projects (Derivative-Free Optimization and Open Tabu Search). In 2004, the initiative was turned over to the COIN-OR Foundation, an organization created for the purpose of supporting the project independent of IBM, and the name was changed to the COmputational INfrastructure for Operations Research to better reflect its expanded mission (while keeping the acronym the same).

Since its inception, the COIN-OR repository has grown to house over 50 separate projects in a variety of areas of computational operations research, including LP solvers, branch-and-cut frameworks and mixed-integer solvers, continuous and discrete nonlinear solvers, algorithmic differentiation tools, modeling tools, and others. The current projects are mostly optimization related, although the COIN-OR Foundation welcomes contributions in other areas (such as stochastic processes, statistics, and simulation) as well.

Development and Dissemination

The COIN-OR initiative provides an infrastructure to support the distributed community development model common to many open source projects and to provide a central forum for publishing software in the operations research field.

The developer infrastructure includes a Web server, a source-code version control system, a mailing list server, a bug tracker and Wiki, and related tools. In addition, Foundation volunteers support a collection of tools to ease the process of compiling and installing

projects that conform to the tool's configuration requirements.

There are several open source infrastructure projects—the largest being SourceForge—which provide similar tools to COIN-OR. Any such centralized service provides advantages over self management: developers need not provide their own repositories or support tools; and Users know where to look for packages and know that the packages and support services work the same way for all projects in the repository.

COIN-OR offers some additional advantages over these generic hosting services for operations researchers:

- *One-stop resource*. Because the initiative is focused on operations research, it serves as a one-stop resource for potential users and developers in the field. SourceForge, by contrast, hosts thousands of open-source projects and does not even list a category for science or engineering applications.
- *OR-focused support*. Because the volunteers on the support staff are operations researchers, they understand the needs of OR users and developers, and they can provide support appropriate to that environment. Developers whose projects meet the requirements of the BuildTools project, for example, don't need to be experts in building and packaging software in order to package and distribute their programs.

A goal of the COIN-OR Foundation is to serve as a peer-reviewed publication venue for operations research software. While the practice of assessing the quality of a journal article is well established, it is an open question how best to evaluate the quality of published source code. Informal, indirect assessment of quality may be based on reputation, penetration, support from the developer community, and other measures of community acceptance (Hafer and Kirkpatrick 2009). The COIN-OR Foundation does review contributed projects for certain features, including:

- *Legal provenance*. The project must satisfy reporting requirements regarding legal documentation. At a minimum, contributors must certify that they understand and have met basic legal requirements to ensure that they have the right to license their contributions. Beyond that minimum, contributors can provide documentation from the owner of the intellectual property certifying that they own the code and license it. The level of certification of each contribution is documented in the project's index entry and is the responsibility of the project manager.
- *Documentation*. The project must include certain documentation, including acknowledgment of contributors, a copy of the license, and instructions for building and installing the software.
- *Functionality*. The projects are also required to "work." That is, an independent reviewer, following the instructions, must be able to successfully build and install the software and run a unit test provided by the contributor.

Providing support for more sophisticated peer review is a long term goal of the foundation.

Using COIN-OR Software

Martin (2010) gives a tutorial on using COIN-OR software. He describes applications in industry, education, and research, and describes how to obtain and use COIN-OR tools.

COIN-OR users fall into two broad categories: users who need a prepackaged tool to solve a particular problem and developers interested in incorporating COIN-OR code into programs that they write themselves. The former group can obtain precompiled binary packages of COIN-OR libraries and programs and use them just as they would other self-contained software products. Developers can check out the source code to the latest versions of COIN-OR programs and either incorporate the code as callable libraries or integrate the source code directly into their own program source. Martin explains both uses.

COIN-OR Projects

The complete collection of COIN-OR projects as of this writing is listed here, by category. The current list is maintained at the COIN-OR Web site. Project URLs can be found by entering the project name and keywords from the description in your favorite WWW search engine. Some projects appear in multiple categories.

Developer Tools

BuildTools COIN-OR Unix developer tools and documentation, tools for managing configuration and compilation of various COIN-OR projects under Linux, Unix, and Cygwin.

CoinBazaar The COIN-OR Bazaar, small examples and extensions of COIN-OR projects.

CoinBinary COIN-OR Binary Distributions, pre-compiled binary distributions of COIN-OR projects.

CoinWeb COIN-OR Web Services, COIN-OR Web pages, Subversion, Trac, etc.

Coopr A COmmon Optimization Python Repository, the Coopr software project integrates a variety of optimization-related Python packages. Coopr supports a diverse set of optimization capabilities that can be used formulate and analyze optimization applications.

TestTools TestTools, Python scripts to automatically download, configure, build, test, and install COIN-OR projects.

Documentation

CoinEasy New user information and support.

Graphs

CGC COIN-OR Graph Classes, a collection of network representations and algorithms.

LEMON Library of Efficient Models and Optimization in Networks, a C++ template library aimed at combinatorial optimization tasks, especially those working with graphs and networks.

Interfaces

AIMMSlinks AIMMS/COIN-OR Links, links between the modeling language AIMMS and solvers that are hosted at COIN-OR.

CMPL Coliop/CoinMathematical Programming Language, a mathematical programming language and a system for modeling.

CoinMP CoinMP, a lightweight API and DLL for CLP, CBC, and CGL.

GAMSlinks GAMS/COIN-OR Links, links between GAMS (General Algebraic Modeling System) and solvers that are hosted at COIN-OR.

NLPAPI Nonlinear Programming API, a subroutine interface for defining and solving nonlinear programming problems.

OS Optimization Services, standards for representing optimization instances, results, solver options, and communication between clients and solvers in a distributed environment using Web Services.

OSI Open Solver Interface, a uniform API for calling embedded linear and mixed-integer programming solvers.

PuLP Python library for modeling linear and integer programs.

SMI Stochastic Modeling Interface, for optimization under uncertainty.

Metaheuristics

Djinni A templatized C++ framework with Python bindings for heuristic search.

METSlib An object oriented metaheuristics optimization framework and toolkit in C++.

OTS Open Tabu Search, a framework for constructing tabu search algorithms.

Modeling Systems

AIMMSlinks AIMMS/COIN-OR Links, links between the modeling language AIMMS and solvers that are hosted at COIN-OR.

CMPL Coliop/CoinMathematical Programming Language, a mathematical programming language and a system for modeling.

Coopr A COmmon Optimization Python Repository, the Coopr software project integrates a variety of Python. optimization-related packages. Coopr supports a diverse set of optimization capabilities that can be used formulate and analyze optimization applications.

FLOPC++ An algebraic modeling language embedded in C++.

GAMSlinks GAMS/COIN-OR Links, links between GAMS (General Algebraic Modeling System) and solvers that are hosted at COIN-OR.

OS Optimization Services, standards for representing optimization instances, results, solver options, and communication between clients and solvers in a distributed environment using Web Services.

PuLP Python library for modeling linear and integer programs, Python library for modeling linear and integer programs.

ROSE Reformulation-Optimization Software Engine, software for performing symbolic reformulations to Mathematical Programs (MP).

Optimization Convex Non-differentiable

OBOE Oracle Based Optimization Engine, optimization of convex problems with

user-supplied methods delivering key first order information (like support to the feasible set, support to the objective function).

Optimization Deterministic Linear Continuous

CLP COIN-OR LP, a simplex solver.

CoinMP CoinMP, a lightweight API and DLL for CLP, CBC, and CGL.

DyLP Dynamic LP, an implementation of the dynamic simplex method.

FLOPC++ An algebraic modeling language embedded in C++.

OSI Open Solver Interface, a uniform API for calling embedded linear and mixed-integer programming solvers.

VOL Volume Algorithm, a subgradient algorithm that also computes approximate primal solutions.

Optimization Deterministic Linear Discrete

ABACUS A Branch-And-CUt System, an LP-based branch-and-cut framework.

BCP Branch-Cut-Price Framework, a framework for constructing parallel branch-cut-price algorithms for mixed-integer linear programs.

CBC COIN-OR Branch and Cut, an LP-based branch-and-cut library.

CGL Cut Generator Library, a library of cutting-plane generators.

CHiPPS COIN-OR High Performance Parallel Search Framework, a framework for constructing parallel tree search algorithms (includes an LP-based branch-cut-price implementation).

DIP Decomposition in Integer Programming, a framework for implementing a variety of decomposition-based branch-and-bound algorithms for solving mixed integer linear programs.

KSP *K* Shortest Paths, algorithms for *K* shortest paths.

SYMPHONY Single- or Multi-Process Optimization over Networks, a callable library for solving mixed-integer linear programs.

VRPH Vehicle Routing Problem Heuristics, a library of heuristics for generating solutions to variants of the vehicle routing problem.

Optimization Deterministic Nonlinear

DFO Derivative-Free Optimization, a package for solving general nonlinear optimization problems when derivatives are unavailable.

filterSD Subroutines for nonlinear optimization, a library for nonlinear optimization written in Fortran.

Ipopt Interior-Point Optimizer, for general large-scale nonlinear optimization.

MOCHA Matroid Optimization. Combinatorial Heuristics and Algorithms, heuristics and algorithms for multicriteria matroid optimization.

NLPAPI Nonlinear Programming API, a subroutine interface for defining and solving nonlinear programming problems.

OptiML Optimization for Machine Learning, interior point, active set method and parametric solvers for support vector machines, solver for the sparse inverse covariance problem.

Optimization Deterministic Nonlinear Discrete

Bonmin Basic Open-source Nonlinear Mixed INteger programming, an experimental open-source C++ code for solving general MINLP (Mixed Integer NonLinear Programming) problems.

Couenne Convex Over and Under ENvelopes for Nonlinear Estimation, a branch-and-bound algorithm for mixed integer nonlinear programming problems.

LaGO Lagrangian Global Optimizer, for the global optimization of nonconvex mixed-integer nonlinear programs.

Optimization Deterministic Semidefinite Continuous

CSDP C Library for Semidefinite Programming, an interior-point method for semidefinite programming.

Optimization Stochastic

Coopr A COmmon Optimization Python Repository, the Coopr software project integrates a variety of Python. optimization-related packages. Coopr supports a diverse set of optimization capabilities that can be used formulate and analyze optimization applications.

SMI Stochastic Modeling Interface, for optimization under uncertainty.

Optimization Utility

ADOL-C Package for the automatic differentiation of C and C++ programs.

CHiPPS COIN-OR High Performance Parallel Search Framework, a framework for constructing parallel

tree search algorithms (includes an LP-based branch-cut-price implementation).

CoinBazaar The COIN-OR Bazaar, small examples and extensions of COIN-OR projects.

CoinUtils COIN-OR utilities, utilities, data structures, and linear algebra methods for COIN-OR projects.

Coopr A COmmon Optimization Python Repository, the Coopr software project integrates a variety of Python optimization-related packages. Coopr supports a diverse set of optimization capabilities that can be used formulate and analyze optimization applications.

CppAD CppAD, a tool for differentiation of C++ functions.

LEMON Library of Efficient Models and Optimization in Networks, a C++ template library aimed at combinatorial optimization tasks, especially those working with graphs and networks.

OS Optimization Services, standards for representing optimization instances, results, solver options, and communication between clients and solvers in a distributed environment using Web Services.

PFunc Parallel Functions, a lightweight and portable library that provides C and C++ APIs to express task parallelism.

Web Services

OS Optimization Services, standards for representing optimization instances, results, solver options, and communication between clients and solvers in a distributed environment using Web Services.

Other Open-Source OR Software

COIN-OR is not the only source of open-source code for OR. Many authors do offer their experimental codes or even production-quality codes individually. Some may be available at SourceForge, GitHub, the GNU project, or other general-purpose repositories; many are offered only on the authors' Web pages. But there is no other centralized location where an operations researcher can go to find those codes.

Below is a list of several OR-related codes that are not part of COIN-OR, of which this author is aware. The list appears on the COIN-OR Web site. URLs can be located through Internet search engines. If you know of other relevant projects, please send this author a link.

cdd The double description method for constructing representations of polyhedra.

Cliquer For finding cliques in graphs.

GLPK The GNU Linear Programming Kit.

Gnumeric Spreadsheet with solvers.

GSL The GNU Scientific Library: C library for mathematical functions, including random variables, statistics, linear algebra, and lots more.

GOBLIN Graph optimization library.

lrs The reverse search algorithm for finding vertices of polyhedra.

Maxima Computer algebra, similar to Mathematica or Maple.

MCFClass Interface for Minimum Cost Flow problems (mix of open-source and other software).

mexclp A MATLAB interface to the COIN-OR LP solver (CLP).

MINLP CMU-IBM Cyber-Infrastructure for MINLP.

MUMPS A MUltifrontal Massively Parallel sparse direct linear system solver.

Octave Matrix based mathematics, similar to and mostly compatible with Matlab.

OpenOffice open-source office suite with spreadsheet optimization using COIN-OR solvers.

OpenSolver Excel plugin for optimization.

OpenForecast The name says it all.

QtsPlus4Calc A collection of OpenOffice spreadsheets that solve a variety of problems related to queuing theory.

R Statistics, graphics, and more. Similar to S-plus (both are based on the language S).

Sage Flexible, extensible system for symbolic and numerical mathematics.

Shogun A large-scale machine learning toolbox.

SolverStudio Cloud-based optimization plugin for Excel.

swIMP SWIG-based interfaces for Mathematical Programming.

Zimpl Translate LP/MIP models into MPS or LP formats.

Concluding Remarks

Open-source software is an important resource for researchers and practitioners of computational operations research. This article reviewed the benefits of open source for academic researchers and practitioners and outlined the legal mechanics of working with open source. The COIN-OR Foundation provides to computational operations researchers an

infrastructure for supporting open source development and a publication forum for open-source software. COIN-OR is the largest repository of open-source software for operations research, comprising more than 50 projects and growing. Other OR-related open-source tools are available, but they are not available from any central source.

References

Gates, W. (1976). An open letter to hobbyists. *Homebrew Computer Club Newsletter, 2*(1), 2.

Hafer, L., & Kirkpatrick, A. E. (2009). Assessing open source software as a scholarly contribution. *Communications of the ACM, 52*(12), 126–129.

Lindberg, V. (2008). *Intellectual property and open source*. Sebastopol, CA: O'Reilly Media.

Lougee-Hiemer, R. (2003). The common optimization INterface for operations research: Promoting open-source software in the operations research community. *IBM Journal of Research and Development, 47*(1), 57–65.

Martin, K. (2010). Tutorial: COIN-OR: Software for the OR community. *Interfaces, 40*(6), 465–476.

Raymond, E. S. (2001a). *The cathedral and the bazaar*. O'Reilly Media.

Raymond, E. S. (2001b). *The magic cauldron*. Raymond.

Rosen, L. (2005). *Open source licensing: Software freedom and intellectual property law*. Prentice Hall. Available online under the Academic Free License version 3.0.

Saltzman, M. J. (2002). COIN-OR: An open-source library for optimization. In S. S. Nielsen (Ed.), *Programming languages and systems in computational economics and finance* (pp. 3–32). Boston: Kluwer. chapter 1.

Stallman, R. M., & Gay, J. (2002). *Free software, free society: Selected essays of Richard M Stallman*. Boston: Free Software Foundation.

Young, R., & Rohm, W. G. (1999). *Under the radar: How red hat changed the software business and took Microsoft by surprise*. Scottsdale: Coriolis Group.

Operational Research Society (ORS)

The United Kingdom's Operational Research Society (ORS) had its origins in an informal OR Club, founded in 1948, for "people who are working in or are concerned with problems associated with Operational Research" (Anon 1950b). Membership was limited to one member per industry or organization. In 1950, the Club established a specialist journal, the *Operational Research Quarterly*, that would "assemble in one place as much as possible of the information that operational research workers now find (or fail to find) scattered widely over the very large body of scientific and technical literature" (Anon 1950a). The journal was renamed the *Journal of the Operational Research Society* in 1978.

The Club was reconstituted as a Society in 1953, with no numerical limit on membership. The aims were defined in the Constitution as "the advancement of education through the provision of training in and the promotion and adoption of operational research." Cooperation with the American and French societies, following the first international conference at Oxford in 1957, led to the creation of the International Federation of Operational Research Societies (IFORS) in 1959. Another outcome of the conference in Oxford was the first U.K. national conference in 1958.

See

► International Federation of Operational Research Societies (IFORS)

References

Anon. (1950a). Editorial notes. *Operational Research Quarterly, 1*, 1–2.

Anon. (1950b). Operational research club. *Operational Research Quarterly, 1*, 36.

Operations Evaluation Group (OEG)

► Center for Naval Analyses

Operations Management

Mark A. Vonderembse[1], William G. Marchal[1] and David Dobrzykowski[2]

[1]The University of Toledo, Toledo, OH, USA

[2]Eastern Michigan University, Ypsilanti, MI, USA

Introduction

Organizations exist to meet the needs of society that people working alone cannot. Operations are part of an organization and they are responsible for producing the tremendous array of products in the quantities

consumed each day. Operations are the processes which transform inputs (labor, capital, materials, and energy) into outputs (services and goods) consumed by the public. Operations employ people, build facilities, purchase equipment in order to change materials into finished goods such as computer hardware and/or to provide services such as computer software development.

Services are intangible products and goods are physical products. According to the classification scheme used by the U.S. Department of Commerce and Labor, services include transportation, utilities, lodging, entertainment, health care, legal services, education, communications, wholesale and retail trade, banking and finance, public administration, insurance, real estate, and other miscellaneous services. Goods are described as articles of trade, merchandise, or wares. Manufacturing is a specific term referring to the production of goods. Here, the term product will be used to refer to both goods and services.

Whether an organization is producing services or goods as part of the for-profit private sector or the not-for-profit public sector, the output from operations should be worth more to customers than the total cost of its inputs. As a result, an organization creates wealth for society through decisions and actions made in managing operations.

Operations management is a multi-disciplinary sub-field of management with particular focus on the production or operations function of the firm. Its scope includes decision making involving the design, planning, and management of the many factors that affect operations. Decisions include: what products to produce, how large a facility to build, how many people to hire, and what methods to use to increase product quality. Operations managers apply ideas and technologies to reduce leadtime dramatically, increase productivity and reduce costs, improve flexibility to meet rapidly changing customer needs, enhance quality, and improve customer service.

Organizations can use operations as an important way to gain an advantage on the competition. Synergy results when operations are linked to the overall strategy of the organization (including engineering, financial, marketing, and information system planning). Operations become a positive factor when facilities, equipment, and employee training are viewed as a means to achieve organizational rather than suboptimal departmental goals.

Increasing demand for product variety and shorter product life cycles are forcing operations to respond more frequently and more quickly to customer needs. Competition is no longer based only on price or price and quality. Competition is becoming time-based, with customers expecting high-quality, low-cost and innovative products that are designed and produced quickly to meet specific customer requirements. When flexibility is designed into operations, an organization is able to rapidly and inexpensively respond to changing customer needs. Organizations can use computers and information technology to become more flexible. Improvements in productivity and product quality provide the basis for competing in global markets.

To be successful, an organization should consider issues related to: designing a system that will be capable of producing the appropriate services and goods in the needed quantities; planning how to use the system effectively; and managing key elements of the operations. Each of these topics are described briefly in the following sections.

Designing the System

Designing the system includes all the decisions necessary to determine the characteristics and features of the goods and services to be produced. It also establishes the facilities and information systems required to produce them. When designing a system which is capable of producing services and/or goods several questions arise.

- What products will the organization produce (product development and design)?
- What equipment and/or methods will be used (process design)?
- How much capacity will an organization acquire?
- Where will the facility be located?
- How will the facility be laid out?
- How will individual jobs and tasks be designed?

Product Development is a process for (1) assessing customer needs, (2) describing how products (both services and goods) can be designed to meet those needs, (3) determining how processes can be designed to make quality products efficiently and reliably, and (4) developing marketing, financial, and operating plans to successfully launch those products. Product development is a cross-functional decision

process that requires teamwork. It is a key factor for success because it shapes how the organization competes.

Product Design is the determination of the characteristics and features of the product, i.e., how the product functions. Product design determines a product's cost and quality as well as its features and performance, and these are the primary criteria on which customers make the decision to purchase. Techniques such as Design for Manufacturing and Assembly (DFMA) are being implemented with very successful results. The objective is to improve product quality and lower product costs by focusing on manufacturing issues during product design. DFMA is implemented through computer software that points designers towards designs that would be easy to build by focusing on the economic and quality implications of design decisions. This is often critical because even though design may be a small part of the overall cost of a product, the design decision may fix 70-90% of the manufacturing costs. Quality Functional Deployment is also being used. It is a set of planning and communication routines that focuses and coordinates actions. The foundation is the belief that a product should be designed to reflect customers' desires and tastes.

Process Design describes how the product will be made. The process design decision has two major components; a technical or engineering component and a scale economy or business component. The technical side requires that decisions be made regarding the technology to be used. For example, a fast food restaurant should decide whether its hamburgers will be flame broiled or fried. Decisions must also be made about the sequence of operations. For example, should a car rental agency inspect a car that has been returned by the customer first or send it to be cleaned and washed? Decisions need to be made regarding the type of equipment to be used in making the good or providing the service. In addition, the methods and procedures used in performing the operations must also be determined.

The scale economy or business component involves applying the proper amount of mechanization (tools and equipment) to leverage the organization's work force to make it more productive. This involves determining whether the demand for a product is large enough to justify mass production; if there is sufficient variety in customer demand so that flexible production systems are required; or if demand is so small that it cannot support a dedicated production facility.

Capacity is a measure of an organization's ability to provide the demanded services or goods in the amount requested and in a timely manner. More specifically, capacity is the maximum rate of production that can be sustained over a long period of time. Capacity planning involves estimating demand, determining the capacity of facilities, and deciding how to change the organization's capacity to respond to demand.

Facility Location is the placement of a facility with respect to its customers, suppliers, and other facilities with which it interacts. Normally, facility location is a strategic decision because it is a long-term commitment of resources which cannot easily or inexpensively be changed. When evaluating a location, management should consider: customer convenience, initial investment for land and facilities, operating costs, transportation costs, and government incentives. In addition, qualitative factors, such as availability of financial service, cultural activities for employees, and university research programs that relate to the needs of the firm, should be considered. As world economies become closely linked, the location decision takes on global dimensions.

Facility Layout is the arrangement of the work space within a facility. At it highest level, it considers which departments or work areas should be adjacent so that the flow of product, information, and people can move quickly and efficiently through the production system. Next, within the department or work area, where should people be located with respect to equipment and storage? How large should the department be? Finally, how should each work area within a department be arranged?

Job Design specifies the tasks, responsibilities, and methods used in performing a job. For example, a job design for x-ray technicians would describe what equipment they would use and explain the standard operating procedures including the safety requirement that should be followed.

Planning the System

A plan is a list of actions management expects to take to deal with opportunities and problems present in the environment. Production planning is how management expects to utilize the resource base created when the

production system was designed. One of the outcomes may be to change the design such as increasing or decreasing capacity and rearranging layout to enhance efficiency.

Production planning decisions depend upon the planning time horizon. Long-range decisions include how many facilities to add to match capacity with forecasted demand and how technological change might affect techniques used to manufacture goods and provide services. The time horizon for long-term planning varies with the industry and depends on how long it would take an organization to build new facilities. For example, in electric power generation it often takes ten or more years to build a new plant. So electrical utilities must plan at least that far into the future.

In medium-range production planning, which is normally about one year, organizations find it difficult to make substantial changes in facilities. In this case, production planning involves determining work force size and developing training programs, working with suppliers to improve product quality and improve delivery, and determining how much material to order on an aggregate basis.

Scheduling has the shortest planning horizon. As production planning proceeds from long-range planning to short-range scheduling, the decisions become more detailed. In scheduling, management must decide what product or products will be made; who will do the work; what equipment will be used; which materials will be consumed; when the work will begin; and what will happen to the product when the work is complete. All aspects of production come together to make the product a reality.

Some techniques used in production planning require special mention. Aggregate planning, material requirements planning, just-in-time, and the critical path method are important techniques that can be useful in production planning.

Managing the System

The impact of people, information, materials, and quality on operations is growing. As a result, managing these areas is a key factor for organizational success. Participative management and teamwork are becoming an essential part of successful operations. Motivation, leadership, and training are receiving new impetus.

Information systems are mechanisms for gathering, classifying, organizing, storing, analyzing, and disseminating information. Information requirements in some operations are extensive. From product development through job design and from long-range planning to scheduling, timely information is required to make better decisions.

Material management includes decisions regarding the procurement, control, handling, storage, and distribution of materials. Materials and material management are becoming more and more important because in many operations purchased material costs are over 50% of the total product cost. How much material should be ordered, when should it be ordered, and which supplier should it be ordered from are some of the important questions.

Producing high-quality products is a minimum requirement for a customer to consider buying an organization's product. Quality is increasingly becoming customer-driven with emphasis put on obtaining a product design that builds quality into the product. Then, the process is designed to transform the product design into a quality product and the employees are trained to execute it. The role of inspection is not to enhance quality but to determine if the designs are effective.

Over time, operations management has grown in scope. It has elements that are strategic; it relies on behavioral and engineering concepts; and it utilizes management science/operations research tools and techniques for systematic decision making and problem solving. It also interacts with other functional specialties such as research and development, marketing, engineering, and finance to develop integrated answers to complex interdisciplinary problems.

See

- ▶ Facility Location
- ▶ Flexible Manufacturing Systems
- ▶ Information Systems and Database Design in OR/MS
- ▶ Inventory Modeling
- ▶ Job Shop Scheduling
- ▶ Production Management
- ▶ Quality Control
- ▶ Scheduling and Sequencing
- ▶ Total Quality Management

References

Blackburn, J. (1991). *Time-based competition, business One*. Homewood: Irwin.
Blackstone, J. H. (1989). *Capacity management*. Cincinnati: South-Western Publishing.
Chase, R. B., & Garvin, D. A. (1989). The service factory. *Harvard Business Review, 67*(4), 61–69.
Clark, K. B., & Fujimoto, T. (1991). *Product development performance*. Boston: Harvard Business School Press.
Day, R. G. (1993). *Quality function deployment: Linking a company with its customers*. Milwaukee: ASQC Quality Press.
Doll, W. J., & Vonderembse, M. A. (1991). The evolution of manufacturing systems: Towards the post-industrial enterprise. *OMEGA International Journal of Management Science, 19*, 401–411.
Flaherty, M. T. (1996). *Global operations management*. New York: McGraw-Hill.
Garvin, D. A. (1988). *Managing quality*. New York: The Free Press.
Skinner, W. (1969). Manufacturing-missing link in corporate strategy. *Harvard Business Review, 52*(3), 136–145.
Sule, D. R. (1988). *Manufacturing facilities: Location, planning, and design*. Boston: PWS-KENT Publishing Company.
Umble, M. M., & Srikanth, M. L. (1990). *Synchronous manufacturing*. Cincinnati: South-Western Publishing.
Utterback, J. M., & Abernathy, W. J. (1975). A dynamic model of process and product innovation. *OMEGA International Journal of Management Science, 3*, 639–656.
Vonderembse, M. A., & White, G. P. (1996). *Operations management: Concepts, methods, strategies*. St. Paul: West Publishing.

Operations Research Office and Research Analysis Corporation

Eugene P. Visco[1] and Carl M. Harris[2]
[1]Silver Spring, MD, USA
[2]George Mason University, Fairfax, VA, USA

Introduction

In discussing the early days of operations research in the United Kingdom, Blackett (1962) notes:

> The Armed Services have for many decades made use of civilian scientists for the production of new weapons and vehicles of war, whereas the tactical and strategical use of these weapons and vehicles has been until recently almost exclusively a matter for the uniformed Service personnel. During the first years of the Second World War circumstances arose in which it was found that civilian scientists could sometimes play an important role in the study of tactics and strategy. The essential feature of these new circumstances was the very rapid introduction of new weapons and devices, preeminently radar, into the Services at a time both of great military difficulty and of such rapid expansion that the specialist officers of the Armed Services, who in less strenuous times can and do adequately compete with the problems raised, found themselves often quite unable to do so. I will attempt to describe … how it was that civilian scientists, with initially little or no detailed knowledge of tactics or strategy, came to play a sometimes vital role in these affairs, and how there grew up a virtually new branch of military science – later to be dignified in the United Kingdom by the name 'Operational Research,' or 'Operations Analysis' in the United States. By the end of the war, all three Services had operational research groups of mainly civilian scientists either at headquarters or attached to the major independent Commands. These groups were, in varying degrees, in close touch with all the main activities of the Service operational staffs and were thus in a position to study the facts of operations in progress, to analyze them scientifically, and, when opportunities arose, to advise the staffs on how to improve the operational direction of the war....

While British military operational analysis was in place in all three uniformed services during World War II, U.S. military operations research during the war was carried out primarily in the Army Air Corps (later the Army Air Force) and the Navy. There was no single U.S. Army group comparable to the British Army Operational Research Group. There was a scattering of small groups doing operational analyses in various parts of the Army. The Signal Corps set up an Operational Research Division to prepare instruction manuals for radio communications by using operational experience data. The Office of Field Service, a major subdivision of the Office of Scientific Research and Development, provided civilian scientists, initially to conduct operational analyses, to Army units in the Pacific Theater. However, the scientists were often called upon to carry out work other than operational analysis. Only the Navy and Army Air Force groups were dedicated to operational analyses. By war's end, the U.S. Army Air Force had 26 Operations Analysis sections assigned to the numbered Air Forces, Commands, Areas, Wings, Boards, and Schools. Approximately 250 analysts served in those sections. A wide range of professions were involved: typically 50 engineers, 40 educators and trainers, 35 mathematicians, 25 lawyers, and 21 physicists. Other professions represented included architects, meteorologists, physiologists, a historian, agriculturists,

investment analysts and stock brokers, an astronomer, biologists, and many others – true adherence to the mixed team concept introduced by the British founders. Some analyses conducted at the Army's Aberdeen Proving Ground in its recently formed Ballistic Research Laboratory (BRL) can certainly be considered Army operations analysis, even though those words were not recognized there. A variety of survivability and vulnerability studies, particularly on Army aircraft, were carried out at BRL, as were many weapons effectiveness and bombing pattern analyses. However, there was no central overall Army operations research group, so identified, other than those working with the Air Forces, during World War II.

Post-World War II Activities

After the war, the British and United States wartime groups, in one form or another, continued to conduct operations research and analysis for their respective services. In the U.S., it was clear, due to the demonstrated relevance and importance of military operations research, that operations analysis organizations were needed in all the services.

Thus, within a few years after the end of World War II, the Navy's Operations Evaluation Group (OEG) [later to become the Center for Naval Analyses (CNA)], the Air Force's Operations Analysis Division and Project RAND, and the Army's Operations Research Office (ORO) were formed. Each has played an important role in the history and development of OR/MS. The Operations Research Office of The Johns Hopkins University (JHU) was founded in 1948 (by the U.S. Army) to serve as the Army's civilian run organization for operations research analysis and studies, with offices in the Washington, DC area. ORO was managed by the trustees of JHU and had offices in the Washington, DC area. ORO had the major goal of providing independent, objective, and scientifically sound studies of national security and defense issues.

The ORO Director

The history of ORO is one with the history of Ellis A. Johnson, its founder and only Director. After earning his M.S. and D.Sc. degrees at the Massachusetts Institute of Technology, Johnson went to Washington in 1934 to work on magnetic instruments at the U.S. Coast and Geodetic Survey. In 1935, he joined the Department of Terrestrial Magnetism, Carnegie Institution as a geophysicist. Early in 1940 he moved to the Naval Ordnance Laboratory (NOL), first as a consultant, then as Associate Director of Research, where he worked on degaussing as a countermine tactic, among other things. He quickly became interested in the operational offensive use of mines and countermeasures to mines. Even during the early days of the minecountermine analysis, Johnson believed that analysts and researchers had to maintain a close association with those who had the ultimate responsibility for military operations – the very essence of operations research. Thus, from the outset, ORO reflected Johnson's wartime experiences and the philosophy of analysis. Much of what follows here, particularly in relation to Johnson, draws heavily on the tribute to him published by the Operations Research Society of America following his death (Page et al. 1974). With respect to the start-up of ORO, Page et al. (1974) noted:

> Thus, as ORO began its work, there was a working assumption that something called operations research was in being, and the Army anticipated its value enough to be willing to try to use it. But for the Army, this did not mean that it was clearly defined. Ground warfare was recognized as a more difficult field for operations research than air and sea warfare; on the one hand, ground warfare could not be affected so much by one new technical factor as air warfare was by radar, while, on the other, the analysis of the convenient geometry of the open space in the air or on the sea was quite inapplicable for troop movement on terrain. So, if OR was to play a significant role in support of Army planning, it would have to learn how to structure the problems, identify the elements amenable to analysis, and find methods of analysis by adaptation or invention. There were almost no direct precedents as to what could be expected....

ORO Activities and Projects

The organizational principles that quickly evolved included: a wide breadth of study topics; control and management of analysis in the hands of the researchers conducting the analyses; and close involvement with the operational elements of the Army, including access to real and often raw operational data representing performance of organizations and systems. Research leaders at ORO were also expected to conduct research

themselves, to maintain a connection with the reality of research management.

The first 2 years of ORO included assignments from the Army covering a major study of military aid to other nations, a study of the causation of artillery firing errors, and armored force operations. During this time the staff was brought to the level of about 40 full-time analysts, and a pool of more than 100 consultants was established, with ORO linkages to a number of research and analytic companies to provide additional, on-call support. Arrangements were concluded with the Army to establish a broad program of continuing research on nuclear weapons, tactics, logistics, military costing, psychological warfare, guerrilla warfare, and air defense. A core set of 15 projects was authorized and funded, thus providing a formative and formidable base from which to proceed.

When the Korean War broke out in June 1950, ORO was a functioning institution with a developing reputation for sound and practical analysis on behalf of the operational Army. Johnson quickly recognized a need and an opportunity in the war. He made an early visit to Korea to establish a modus operandus for field analysis teams in the theater of operations. By the fall, ORO had 40 analysts in the field (as many as the full staff only a few months earlier). At the end of the war, over 50% of the professional staff had spent time in the combat theater. Many hundreds of reports were written, with considerable impact on military operations. ORO's influence was felt in the UK and Canada, and operations analysts from those two UN participating countries joined their respective countries' military units operating in Korea.

A small ORO field team was organized at the Continental Army Command Headquarters, Fortress (now Fort) Monroe, Virginia. At that time, CONARC, as the Command was known, was responsible for the development of operational doctrine for the Army and for training related to that doctrine. It was Johnson's view that operations research could make important contributions to the development of doctrine, particularly considering the need for combat formations to adapt to the new considerations of ground combat under conditions of the potential use of atomic (later nuclear) weapons on the battlefield. ORO help design formations, assisting in the structure and doctrine for the Pentomic Division and the Pentagonal (for five combat commands) Division. Other studies looked at the vulnerability of armored formations to tactical nuclear weapons and at the potential for the offensive use of low-yield nuclear weapons. Much attention was paid to tactical operations and logistics in the early days. Later, there were studies related to strategic matters, the most demanding and significant of which was a large study devoted to defense of the U.S. mainland from manned bomber attack involving nuclear weapons.

A field office of the ORO was also established at the headquarters of the U.S. Army Europe in Heidelberg, Federal Republic of Germany, where major contributions to the defense of Europe and NATO operations were made. Heavy use was made of war gaming and exercises for European operations at the Heidelberg office.

ORO was a continuing and positive force in advancing military OR and OR in general. It conducted a series of conferences designed to evaluate the Army's proposed research and development budget to help the Army understand the potential effects of R&D investments and improve the allocation of funds to the many R&D projects competing for support. The PISGAH (named for the mountain from which Moses saw the promised land) conferences brought uniformed officers, operations analysts, industrial scientists, and academics together to examine the Army's future needs. Seminars and colloquia were regular weekly events; the former related to planned or on-going research or outside speakers of note, while the latter focused on more abstruse mathematical analysis topics. ORO conducted experiments to test the capability of bright high-school students to conduct (relatively) independent analyses, under the guidance of senior ORO analysts. Through the years, studies by student teams were done on a wide range of topics, including, for example, the characteristics of effective air raid warning systems for civilians, and deep-thrust armored operations in difficult terrain. During the 5 years of the program, 75 students spent at least one summer at ORO, and a number joined the regular staff after completing college.

During the 13 years of ORO activity, a full range of Army study topics was addressed: air operations and air defense; guerrilla, urban and unconventional warfare; tactical, intra-theater and strategic mobility and logistics; weapons systems; civil defense; intelligence, psychological warfare and civil affairs; and, overall, Army readiness for operations in a complex national security world

(Operations Research Office 1961). Two examples are cited next to demonstrate the wide-ranging impact of ORO studies.

In the arena of tactical operations, ORO examined ways to improve the casualty producing capability of small arms fire. Two unique ideas were introduced and assessed. One was a salvo concept, developed from a patent taken out in the nineteenth century by a serving Army officer. The concept consisted of a system of two projectiles of rifle ammunition, one nested behind the other, with a single cartridge casing and propellant. With one trigger pull, the two rounds came out of the weapon in tandem. ORO analysts predicted, using probability theory, that the natural spread of the two projectiles would greatly increase the hit probability on a man-sized target at operational combat ranges. An ORO analyst, returning to an earlier principle of operations analysis as an experimental science, cast a few bullets in the salvo mode, loaded them with his hand-loading equipment, and fired them on his backyard range. The crude experimental results confirmed the statistical analysis. The Army accepted the results and standardized a two-round salvo projectile for the M14.30 caliber rifle.

The second concept concerning improved effectiveness of small arms fire focused on infantry rifle training. The ORO analysts developed and tested a simulated infantry battlefield target array as an alternative to the known-distance range traditionally used to train infantry. Sets of man-sized targets were scattered over the battlefield and linked with electronic controls that caused the targets to pop-up to vertical positions simulating enemy shooters. The concept was adopted by the Army as the TRAIN-FIRE system.

The second example is a study that ORO did on the use of black soldiers in Korea and the extension of the study to the broader issue of full integration of black troops throughout the Army (Hausrath 1954). From the Revolutionary War to the Korean War, it was traditional for the U.S. military services not to integrate its forces. President Truman's 1948 Presidential Executive Orders, directing equal opportunity in the Executive Branch and the Armed Forces, plus the growing post-World War II economy and major demographic changes, gave impetus for the study requested of ORO by the Army. The study used a wide range of tools: demographic analysis, opinion and attitude surveys, content analysis, critical incident technique, statistical analysis, and community surveys.

The project's summary from Hausrath (1954) notes the following:

> . . . this study provided policy-makers in the U.S. Army with objective evidence in support of integrated units of Negro and white soldiers. This evidence indicated: first, that integrated units allow more effective use of the manpower available through a more even distribution of aptitudes than is possible in segregated units; second, that performance of integrated units is satisfactory; and, third, that the resistance to integration is greatly reduced as experience is gained. The limit, if any, on the level of integration was shown to be above 20% Negroes, and difficulties in extending integration to all parts of the Army were identified and arranged in a sequential order so that a program leading to Army-wide integration could be formulated.

The ORO findings, conclusions and recommendations supported the Army process and success in integration during the 1950s.

End of ORO

In 1961, The Johns Hopkins University, following a disagreement with the Army over management issues, withdrew from the contractual relationship. At midnight on August 31, 1961, the Johns Hopkins University Operations Research Office ceased to exist. Its activities were transferred to the newly formed Research Analysis Corporation (RAC), a Federal Contract Research Center (FCRC).

This brief history of ORO is closed with a statement from the late Ellis A. Johnson, written in the summer of 1961 (Operations Research Office 1961):

> During the last 13 years ORO's accomplishments have indeed been noteworthy. ORO published 648 studies containing thousands of conclusions and recommendations. A majority of these have been adopted and acted on. This survey was written to summarize ORO accomplishments so that these could be considered in perspective and with satisfaction by those responsible for the accomplishments – the entire ORO staff: research staff, support staff, and administrative staff.
>
> We can all be proud of this record.

Transition to RAC

Although the great bulk of RAC's work would be done for defense agencies, RAC sought to diversify its capabilities and its clients. As a result, RAC's activities were expanded to include the White House

and the National Security Council; the Department of Defense; nine other governmental agencies with national security interests; some 40 other governmental agencies of all levels; and private foundations whose primary interests lie outside the field of national security. This range of clients, and their varying interests and requirements for research support, broadened RAC's resources of data and knowledge, analytic capability and interpretative skills. Work done on foreign aid and development further enriched RAC's capabilities to deal with the Army's problems, as did its work for urban clients on problems of crime and delinquency.

Throughout its existence, RAC viewed its major mission to be service to the public interest, chiefly by providing the Army with services, studies, research, and counsel based on operations research and systems analysis. RAC's Army work concentrated on force structure analysis and planning, logistics, military manpower, resource analysis, cost studies, and military gaming. These studies encompassed problems of operations, planning, intelligence, and research and development. In addition, RAC made studies of the nature and purposes of insurgency, counterinsurgency, and operations undertaken to stabilize societies under threat. RAC examined the politico-military aspects of these regions where U.S. land forces were already operating or providing advice and training, or might be called on to do so. These studies considered both current and projected environments. As ORO did, RAC set up field offices to study problems on the scene and to provide analytic support in direct conjunction with local tests and local operations.

RAC's Project Portfolio

Some of the largest projects RAC undertook involved very complex, worldwide logistical and transportation systems. RAC's studies of air mobility for army forces (in both wartime and peacetime) led to specific evaluations later on, after such systems were built, deployed, and put to work, thus creating bodies of hard data suitable for operations analysis.

There was also the continuation of ORO's emphasis on the assessment of weapons requirements and of the comparative effectiveness of competing weapons systems. RAC engineers and scientists also sought ways to improve the management of military research and development, seeking more efficient ways of allocating uncommitted resources to research and development projects. RAC analysts also dealt with communications, proposing new ways to allocate radio frequencies to military users and to improve the dependability of communication nets, early forerunners of today's highly sophisticated command, control and communications systems.

RAC inherited an especially strong program from ORO of basic research into quantitative methods for analyzing a wide variety of OR problems, particularly in the areas of mathematical programming and decision analysis. The term "think tank" was most appropriately applied to RAC at the time.

RAC's work also included economic, political and social science studies of problems arising outside formal military institutions. Most prominent were studies of public safety problems, the administration of justice and control of crime and delinquency, and economic and social development at home and abroad. These grew logically out of RAC's work for defense clients. Also prominent in RAC's work on military subjects were manpower and personnel. Problems in these areas took on new dimensions in the 1960s, under the impact of Vietnam, the draft, and the later shift to an all-volunteer army.

ORO had pioneered the study of military costs and cost analysis, war gaming and simulations, and strategic and limited war. RAC continued these efforts, improving methods and exploring further the possibilities for applying more sophisticated and powerful analytic procedures to the unfolding problems of the 1960s. RAC also conducted studies in arms control and disarmament. It inaugurated a broad range of politico-military analyses relevant to the needs not only of military planners, but also of those concerned with broader questions of national security.

When RAC took over from ORO, a program of advanced research studies was well under way and maintained momentum throughout most of RAC's existence. This continuation was not without conflict, both within the Army and within RAC, since there were serious differences of opinion about how much effort (if any) should be devoted to basic research, as opposed to applications of existing techniques that would be directly useful to the Army in the short run. At the outset, top Army officials decided that such a program was needed and suggested that it form about 10% of the total effort under the RAC-Army

contract. They felt it was necessary to explore ways in which advances in basic methodology might be used to deal with short-run problems. RAC's Advanced Research Group devoted considerable attention to applications, while it continued to make advances primarily in mathematical programming and decision analysis. Fiacco and McCormick's 1968 seminal book on nonlinear programming was a product of the Advanced Research Group.

RAC also conducted a number of studies in the areas of decision, utility, and cognitive theory. In total, these studies were aimed at a comprehensive understanding and theory of decision making at its various levels. In addition to client research applications, this work provided guidance for the establishment of a problem solving rationale within RAC.

In 1972, the General Research Corporation (GRC), a for-profit organization, bought RAC and partly took over its staff, physical assets, and contract relationships with the Army and other RAC clients.

Concluding Remarks

ORO and RAC were important elements of OR/MS history. It would probably be fair to say that their combined contributions played a major role in establishing operations research as a paradigm for rational decision making.

See

- ▶ Air Force Operations Analysis
- ▶ Battle Modeling
- ▶ Center for Naval Analyses
- ▶ Military Operations Research
- ▶ RAND Corporation

References

Blackett, P. M. S. (1962). *Studies of war, nuclear and conventional*. New York: Hill and Wang.

Fiacco, A. V., & McCormick, G. P. (1968). *Nonlinear programming: Sequential unconstrained minimization techniques*. New York: Wiley.

Harris, C. M. (Ed.). (1993). Roots of OR, I: The research analysis corporation, The World War II years. *OR/MS Today*, *20*(6), 30–36.

Harris, C. M. (Ed.). (1994). Roots of OR, II: The history of the research analysis corporation continues with the Vietnam Era. *OR/MS Today*, *21*(3), 46–49.

Hausrath, A. H. (1954). Utilization of Negro manpower in the army. *Operations Research Society of America, 2*, 17–30.

Moore, L. (1968). 20th Anniversary of ORO/RAC. *The Raconteur*, *4*(15).

Operations Research Office. (1961). *A survey of ORO accomplishments*. Baltimore: The Johns Hopkins University.

Page, T., Pettee, G. S., & Wallace, W. A. (1974). Ellis A. Johnson, 1906–1973. *Operations Research Society of America, 22*, 1141–1155.

Parker, F. A. (1967). *An introduction to the research analysis corporation*. McLean, VA: RAC.

Operations Research Society of America (ORSA)

Founded in 1952, the Operations Research Society of America was the major U.S. society for operations researchers. It was merged with The Institute of Management Sciences (TIMS) into the Institute for Operations Research and the Management Sciences (INFORMS) effective January 1, 1995. The purposes of ORSA were (1) the advancement of operations research through the exchange of information, (2) the establishment and maintenance of professional standards of competence for work known as operations research, (3) the improvement of the methods and techniques of operations research, (4) the encouragement and development of students of operations research, and (5) the useful applications of operations research. During the period of its independent existence, ORSA published the journal *Operations Research* (in 42 volumes), as well as other journals (some jointly with TIMS). In addition, ORSA sponsored national meetings (jointly with TIMS), and other meetings organized by its technical sections and geographic chapters. It was the U.S. representative to International Federation of Operational Research Societies (IFORS).

See

- ▶ Institute for Operations Research and the Management Sciences (INFORMS)
- ▶ The Institute of Management Sciences (TIMS)

Opportunity Cost

The cost associated with forgoing an opportunity; the money or other value sacrificed by choosing a nonoptimal course of action. In linear programming, the opportunity cost is the reduced cost of a variable not in the optimal basic solution. If a unit of a nonbasic variable is introduced into the solution, the optimal value of the objective function would decrease by an amount equal to the associated reduced cost.

See

- ▶ Linear Programming
- ▶ Prices
- ▶ Simplex Method (Algorithm)

Optimal Computing Budget Allocation

A statistical ranking and selection framework for choosing the best system design among a finite set of alternatives whose performance must be estimated, usually via simulation, with the objective of maximizing the probability of correct selection.

See

- ▶ Statistical Ranking and Selection

References

Chen, C.-H., & Lee, L. H. (2010). *Stochastic simulation optimization: An optimal computing budget allocation*. Singapore: World Scientific.

Optimal Control

Branch of engineering and applied mathematics dealing with optimization of a dynamical system in continuous time. Similar to dynamic programming, the (optimal) value function satisfies an optimality condition, the Hamilton-Jacobi-Bellman equation. For the special case of a linear time-invariant dynamical system with quadratic cost, an explicit solution for the optimal feedback control policy can be found by solving the Riccati equation.

See

- ▶ Control Theory
- ▶ Dynamic Programming
- ▶ Hamilton-Jacobi-Bellman Equation

References

Bryson, A. E., & Ho, Y. C. (1975). *Applied optimal control*. Washington, DC: Hemisphere.
Sethi, S. P., & Thompson, G. L. (2000). *Optimal control theory: Applications to management science and economics* (2nd ed.). New York: Springer.

Optimal Feasible Solution

For an optimization problem, an optimal feasible solution is a solution that satisfies all the constraints of the problem and optimizes the objective function.

Optimal Solution

- ▶ Optimal Feasible Solution

Optimal Stopping

Sequential decision-making problem under uncertainty in which a decision maker must decide when to stop observing a stochastic process, with the usual objective being to maximize a terminal reward (or minimize cost). Many practical OR/MS applications can be formulated as optimal stopping problems. A well-known example is the so-called secretary problem, in which an employer interviews potential candidates in succession and must decide when to stop the interviewing process and select one

to hire. In finance, the pricing of American options is a well-known class of optimal stopping problems. In discrete time, optimal stopping problems can be formulated as Markov decision problems, in principle solvable by dynamic programming.

Optimal Value

The best value that can be realized or attained; for a mathematical programming problem, the minimum or maximum value of the objective function over the feasible region.

Optimal Value Function

The optimal value of a mathematical programming problem as a function of problem parameters, such as objective function coefficients. Also the name given to the function satisfying the Bellman optimality equation in a Markov decision process or dynamic program, especially in a revenue/profit maximization problem; otherwise sometimes known as the optimal cost-to-go function for a cost minimization problem.

See

- ▶ Approximate Dynamic Programming
- ▶ Bellman Optimality Equation
- ▶ Markov Decision Processes
- ▶ Optimal Value

Optimality Criteria

Mathematical conditions used to test whether or not a given feasible solution is optimal in an optimization problem. Examples include the Karush-Kuhn-Tucker conditions for some nonlinear-programming problems; the simplex algorithm test applied to the reduced costs of the nonbasic variables for linear-programming problems; the Bellman optimality equation for dynamic programming, and the Hamilton-Jacobi-Bellman equation for optimal control.

See

- ▶ Bellman Optimality Equation
- ▶ Hamilton-Jacobi-Bellman Equation
- ▶ Karush-Kuhn-Tucker (KKT) Conditions
- ▶ Linear Programming
- ▶ Nonlinear Programming
- ▶ Simplex Method (Algorithm)

Optimization

The process of searching for the best value that can be realized or attained. In mathematical programming, this is the minimum or maximum value of the objective over the feasible region. Optimization without constraints is called unconstrained optimization.

See

- ▶ Mathematical Programming
- ▶ Nonlinear Programming
- ▶ Unconstrained Optimization

Optimization of Queues

The process of determining the optimal setting of a particular queueing system parameter. The optimization refers to the minimization or maximization of a cost function where the parameter (or parameters) of interest appear as variables.

See

- ▶ Queueing Theory

Option Pricing

In finance, finding the value of an option, a type of financial derivative.

See

- ▶ Financial Engineering

OR

Operations research or operational research.

OR/MS

Operations research and management science.

Ordinal Optimization

In the simulation optimization setting, an approach that exploits the property that it is easier to select the correct order among noisy measurements than to obtain precise estimates, i.e., ordering converges faster (exponentially) than estimation.

See

- ► Large Deviations
- ► Simulation Optimization

References

Ho, Y. C., Sreenevas, R., & Vakili, P. (1992). Ordinal optimization of DEDS. *Discrete-Event Dynamic Systems: Theory and Applications, 2*, 61–88.

Organization

Richard M. Burton[1] and Børge Obel[2]
[1]Duke University, Durham, NC, USA
[2]Aarhus University, Aarhus, Denmark

Introduction

Organization studies encompass two areas: organization theory as a positive science to explain and understand the structure, behavior, and effectiveness of an organization; and organizational design as a normative science to recommend better designs for increased effectiveness and efficiency. Organization theory attempts to understand and explain; organizational design creates and constructs an organization.

Organizing behavior is evident in history from the earliest of recorded time. Ancient China was a highly organized society, a meritocracy with labor specialization. The Roman Empire, and in particular the Roman army, was efficiently designed. The modern organization is part and parcel to civilization, and its understanding fundamental to modern life. Not only is organization both timely and timeless, its study is basic in management science, political science, economics, sociology, business, and military science, to name a few. Organization study is interdisciplinary and central to all of social science. Scott and Davis's book (2006) is an extraordinarily comprehensive and lucid integrating review of the sociological approach to organization theory. It is a positive science review and considers organizational design only implicitly.

The great insight that management science brought to understanding organization is that the basic work of organization is information processing. The information processing perspective permits one to move easily from the positive view of organization to a normative view of what should be; that is, the way the organization deals with information can be modified. In a rough analogy, the nerve system, which channels information even more than the blood (energy carrier) or the skeleton (structure), provides the fundamental basis for understanding organization in modern life. Despite centuries of organization study, the organization as an information processor is a new insight of the twentieth century — even the latter half of the twentieth century. To study organization without information is analogous to studying the human body but ignoring the nervous system; it can be done, but much is lost, or ignored.

In this brief essay, the focus lies on the contribution of management science and operations research to the study of organization. A more formal description and definition of an organization will be given. Then, a number of management science theories, models, and methods are considered one. Finally, some alternative approaches to organization are briefly mentioned and what the future holds. Throughout, the management science literature is referenced which will provide a beginning point to pursue the issue in greater detail.

Organization as Information Processing

What is an organization? Definitions abound. All have certain elements in common. An organization is a created social entity which is composed of individuals (and machines) who must be coordinated to achieve its purpose. March and Simon (1958), in an early and perhaps the most influential book in modern organization studies, wrote (p. 4):

> A biological analogy is apt here, if we do not take it literally or too seriously. Organizations are assemblages of interacting human beings and they are the largest assemblages in our society that have anything resembling a central coordinative system. Let us grant that these coordinative systems are not developed nearly to the extent of the central nervous system in higher biological organism — that organizations are more earthworm than ape. Nevertheless, the high specificity of structure and coordination within organizations — as contrasted with the diffuse and variable rations among organizations and among unorganized individuals — marks off the individual organization as a sociological unit comparable in significance to the individual organism in biology.

March and Simon's organization examined attitudes, values, and goals and developed propositions about decision makers and problem solvers — a new organizational vocabulary to replace authority, responsibility, and span of control as organizing principles. They provided a new basis for thinking about organizing — the principle of bounded rationality (pp. 140–141):

> Most human decision-making whether individual or organizational, is concerned with the discovery and selection of satisfactory alternatives; only in exceptional cases it is concerned with the discovery and selection of optimal alternatives.

This is in contrast to the rational economic man whomakes optimal decisions in well-defined environments. The information processing model is a powerful metaphor of an organization which processes information to obtain coordination:

- Reads information, or observes the world,
- Stores information, or remembers facts and programs,
- Transmits information, or communicates among the members,
- Transposes information, or makes decisions.

These are the work tasks of the organization. Information processing includes choosing, decision making, and problem solving. At a very basic level, the work of an organization is symbol manipulation. Whether human or machine, the organization is rational only in a bounded sense, reaching less than optimal decisions with the less than perfect information available to it.

Coordination of decisions and their implementation is the fundamental problem. Team theory models (Marschak and Radner 1972) of organization were explicit mathematical models of multi-person organizations who had to make multiple decisions in the face of uncertainty — both uncertainty about the true state of nature and about the information and decisions of other team members. Better prediction, communications, and decision rules reduce the level of uncertainty and obtain more nearly optimal decisions. The team theory models explicitly incorporated information: reading, storing, communicating, and calculating. The best information scheme, or organizational design, balances the returns from nearly optimal coordinated actions and the costs of organizing.

The ship builder's problem is deceptively simple, yet fundamental (p. 132):

> Let a firm have two sales managers, each specializing in a different market for its product. Let it have two production facilities, one producing at low cost and another, more costly, to be used as a standby. This second facility can be visualized as a separate plant or as the use of the same plant at "overtime" periods, which involves higher wages. A conveniently simple case is offered by a shipyard firm with two docks (a new one and an old, less efficient one) and two markets ("East" and "West"). Each sales manager is offered a price for a ship to be delivered in his market. The prices offered in each of the two markets are the two state variables. (That is, the market prices have a priori known probabilities of high or low.) There are two decision variables, each of them taking one of two values: either accept or reject the order.

There are nine possible organizational designs about reading and communicating. For each case, there are decision rules which maximize the expected returns. Here are four of the possible designs:

1. No market information is gathered or communicated;
2. Both market prices are observed and communicated to a central headquarters;

3. Each market price is observed and a decision is made to accept the offer, or not; and
4. The market price is obtained in one market and sent to a central headquarters, but not in the other.

The best design depends upon the returns from better information and cost of observing, communicating, and choosing. The costlier the observations, the fewer the observations. The costlier communications are, the more decentralized the organization. The decision rules depend upon the information available.

Information processing is a core issue. March and Simon begin with the boundedly rational individual and build implications for the organization. Marschak and Radner take a number of perfectly rational individuals who are bounded rationally as a team because of the limited and costly information available, that is, imperfect information at the right time for the right person.

Focusing directly on organizational design, Galbraith (1974) assertively conjectured that the principal managerial task is to reduce uncertainty by processing information: "A basic proposition is that the greater the uncertainty of the task, the greater the amount of information that has to be processed between decision makers during the execution of the task" (p. 28).

Following March and Simon, Galbraith (p. 29) offered three mechanisms to obtain greater coordination among the decision makers:

1. If coordination by rules or programs: operational contingent rules can be stated in "if-then" terms; for example, if the inventory stock is less than four, then reorder ten items. Programs are compositions of large numbers of rules;
2. Hierarchy: with greater uncertainty and no rules, exceptions and new situations are referred up the hierarchy for resolution. (This is a rule itself; *if* there is great uncertainty and no rule about what to do, *then* refer the issue up the hierarchy.); and
3. If coordination by targets or goals: here the rules may be largely unspecified but the desired ends or goals can be stated. Subgoals are developed to obtain coordination among the units.

Adding to these organizational design alternatives, Galbraith (p. 30) then offered four information processing strategies. The first two reduce the need for information processing by creating slack resources (for example, excess personnel to complete a task) and self-contained units; that is, small quasi-independent units. Alternatively, increased information processing capacity can be obtained by investing in vertical information systems, for example, MIS; or by creating lateral relations; that is, the genesis of the matrix organization (Galbraith 1995). Each organizational design alternative is developed and its appropriateness rationalized on the need for information to coordinate activities in the face of uncertainty.

Building upon these ideas, Burton and Obel (1980, 1984) formally modeled a hierarchical, decentralized organization using a Dantzig-Wolfe decomposed linear program – a model of who does what based upon what information. It is an explicit multi-agent information processing model for observing, storing, transferring, and decision making – based upon bounded rationality and localized information among the agents. Divisional units pass up local planning information to the headquarters unit, which evaluates these plans and sends revised guidance on limited resource costs to the units. To replicate actual planning systems, only a very few iterations are permitted prior to implementation. Research questions focus on which organizational design – that is, information and decision making system in the spirit of team theory – would yield the best performance in the face of uncertainty. The empirical results verify Williamson's M-form hypothesis that a divisional organization yields better performance than a functional organization – without invoking opportunism or information misrepresentaion by the divisional agents. The power of this approach is to test alternative organizational designs and the way information is handled in an organization to assess which alternative is more efficient. In a second experiment, these mathematical computer simulations were modified for laboratory experiments (Burton and Obel 1988) to investigate the importance of opportunism; that is, whether individuals give misleading information to better their own situation at the expense of others and the organization as a whole. Indeed, some will behave opportunistically, but the M-form suffered less due to opportunism than the functional U-form. These computational and laboratory studies are

controlled experiments to investigate basic hypotheses in organizational design. The models are explicitly information processing where the individuals in the models face an uncertain environment, are bounded rationally, have limited information, and can behave opportunistically. The first model confirmed the M-form hypothesis without invoking opportunism; the second laboratory study again confirmed the hypothesis invoking opportunism.

More recently, Mihm et al. (2010) modeled hierarchy using an NK approach (where N is the number of variables and K is the degree of dependency among the variables), finding that: assigning a lead function speeds up the problem solving; local search should be delegated to the lowest level; and structure matters less in the middle than the front line, which should be kept small.

These later studies provide a transition to computational modeling of organizations – their decision-making and information processes.

Computational Organization Theory

Computational approaches or simulation offers a complementary approach to the study of organization. In launching a new journal, *Computational and Mathematical Organization Theory*, Carley (1995) outlined how computational methods can be applied to better understand organizations – to develop theory and improve practice. Computational models are frequently complicated but mathematically ill-formulated, and hence do not lend themselves to analytical closed-form solutions. As such, simulation is a very powerful approach to investigate complex phenomenon without the need for inappropriately simplifying assumptions. Computational models are usually explicit in their modeling of organization as an information processing task. On the other hand, computational models can be made too complicated, beyond the purpose or the question at hand. Burton and Obel (1995) argue that a good computational model is a parsimonious balance of the purpose, the model, and the experiment. They outline the role of computational modeling in theory development, suggesting and testing simple mechanisms, testing limits and boundary conditions, developing alternative explanations, and more generally exploring with what-might-be models to develop insight and develop a better understanding of organizations and organizing. In the discussion below, a number of computational models are commented on which are parsimonious and balance complexity and simply of purpose. These applications are varied in nature – ranging from the behavioral theory of the firm to organizational design in NASA.

Computational organization models were pioneered by Cyert and March (1963) and related studies which helped in the development of the behavioral theory of the firm. The store-buyer model and oligopoly model were early applications which confirmed the idea that simulation models could be used both for real-world application and theory development. In short, computational models are laboratories (Burton and Obel 2011) for studying organizations and how they work.

Cohen et al. (1972) developed a "garbage can model" of organizational choice. They, too, began with observing how organizations (here, educational institutions) choose, or make decisions. Their discovery was a process in marked contrast to the normal scientific method which gathers data, defines the problem, lists the alternatives, chooses the best one, and then implements it. Rather, the organization was a garbage can of an unordered set of choices looking for problems; issues and feelings seeking forums for airing; solutions looking for issues; and decision makers looking for work. They translated these observations into an explicit computer simulation model and were able to verify and explain a number of observations. Can such an organization or super-bounded rationality ever accomplish anything? Perhaps surprisingly, yes! One conclusion is that "important problems are more likely to be solved than unimportant ones" (p. 10). This much is reassuring – indeed, such organizations can and do function and can be quite effective. Information is used in very complex ways. Their simulation model was devised to explain and understand these very complex organizational processes; nonetheless, the model itself is parsimonious.

Individuals or organizations learn how to update these routines through experience, including sampling experience, which can be biased. March (1991) introduced two contrasting learning strategies: explore and exploit. He demonstrates that exploitation is more likely to be beneficial in the short run, but self-destructive in the long run. Exploration is not as

beneficial in the short run, but can yield strategies that are viable for the longer run. Each strategy has benefits and risks for the learner. The concept of a slow learner (one that adapts gradually to the code) and the result that a slow learner can yield better outcomes for exploration is perhaps counterintuitive. The fast learner adapts too quickly to realize exploratory behavior and thus puts the organization at risk. These notions have inspired extensive research in organizational learning and strategy – both related computational models and empirical studies. With these computational models, the theory of organizational learning is advanced.

Carley and Prietula (1993) have utilized computational organization theory as a complement to deductive analytical modeling and field-based empirical study, again an information processing perspective. Carley et al. (1992) generalized the SOAR model. SOAR models simulate goal-driven search behavior through problem spaces. SOAR acts through a series of decision cycles which include working memory, permanent memory of if-then production rules, and a preference memory. The task is the retrieval of requested warehouse items involving multiple agents. Carley and Lin (1995) investigated information distortion effects in the SOAR environment. More recently, Carley and Prietula (1998) introduced the concept of a WebBot – a critter which does information work, but can also display more human characteristics, such as trust.

Burton and Obel (2004) and Baligh et al. (1996) have devised the Organizational Consultant. It is a knowledge-based expert system which uses what is known from organization theory and executive experience. The Organizational Consultant then asks questions about the specific organization and, utilizing the knowledge base, diagnoses the organization and offers design recommendations. The Organizational Consultant is a computational model in that data or situational facts are analyzed in order to make specific recommendations or a solution. It is not a numerical calculation; the model is a set of some 300 if-then statements which examine the data in order to develop the design recommendation. A broad range of organizations and case studies have been devised to validate the approach. It is an explicit normative model of organizational design. In the spirit of computational modeling, Baligh (2005) developed a process of design that is systematic; it uses the algebra of decision rules. He defines an organization structure as a set of people connected by decision rules. These are mappings of which an element is of the form (If A, choose (do) one of the elements of B), where A represents a circumstance and B a set of possible decisions. Each rule has a set of makers and a set of users. Different rules and different sets of rules describe many different structures, including all those mentioned above. Decision rules identify their own information needs, and design decisions must consider the returns to decision rules and the costs of the information they require.

Levitt et al. (1994) built a multi-agent network model (SimVision) of a project organization. One purpose is to predict the duration of the project. A second more important purpose is to bring understanding and insight into the management process itself. Managerial bottlenecks become evident in the model before they are realized on the job. The model is an information-processing model of an organization: tasks, agents, communications, tools, and structure. The agents are boundedly rational. Each agent manages in an information-rich environment. The computer processes the set of tasks for a network project – a 3-year petroleum refinery design project. Their experiment compares the decentralized and centralized organizations, and voice mail and no voice mail. Decentralization reduces the total work-days, and voice mail reduces the total effort. The more general purpose and application is to predict managerial problems for the organization and then to devise means to prevent the difficulties rather than create and realize difficult and costly situations for the organization. Jin and Levitt (1996) elaborated on the model's relation to contingency theory and the model's validation.

In a related real-world application, NASA began with a clear purpose to design a high-level multi-location project organization (Carroll et al. 2006). The goal was to determine what should be the project design, with the stipulation that it must be closely related to the current practice and an examination of other what-might-be alternatives. They utilized three different simulation tools to help them develop alternative organizational designs and assess their projected performance. In their triangulation approach, the NASA design team began with DSM (design structure matrix) to map the

information processing or communications and coordination structure of existing and proposed organizations to gain insights about the issues and challenges. Then, applying the OrgCon, an expert system of design rules, and using a top-down approach, the design team examined a large number of what-might-be alternative design tradeoffs and their projected consequences. Finally, using a micro bottoms-up approach for the project and the requirements, they developed a what-should-be SimVision model of the proposed organization. The triangulation of the three approaches served multiple practical purposes for NASA. They had multiple views of the problem, which gave them greater understanding. They also had greater confidence in the recommended design, as it was embedded in current practice and the examination of several alternatives and their implications.

Using an NK agent-based model, Siggelkow and Levinthal (2003) examine performance when the competitive landscape shifts. A temporary decentralization – a form not found in the literature – performs best. As interactions across and within divisions increase, the optimal length of decentralized exploration tends to grow. It is a deeper examination of dynamic relations of a what-might-be situation in order to generate a more nuanced theory. Siggelkow and Rivkin (2005) demonstrate that ample processing power at the bottom of the firm can slow down improvement and narrow overall search by the firm – initially a counterintuitive result. The results are stated as hypotheses that can be tested in other settings. Ethiraj and Levinthal (2009) found that incomplete guides to action prove more effective at directing and coordinating behavior than more complete representations. Fewer goals provide clarity and focus for boundedly rational actors. In these what-might-be studies, insights and some counterintuitive results are found that extend understanding.

The Future

The management science information perspective described above remains a base for the future – after all, information processing is fundamental to what an organization does. Daft and Lewin (1990), in launching a new journal, *Organization Science*, began with a provocative question: "Is the field of organization studies irrelevant?" Where is the audience in business and government? What does it mean to be "relevant?" Perhaps relevancy is developing a better understanding of the future, but not necessarily predicting what will happen. What-might-be studies which explore new ideas and possibilities, investigate mechanisms, alternative plausible explanations, boundaries, limits of explanations are the future. Knowledge from diverse sources and multiple perspectives should be encouraged, given the complexity of organization. No one method or approach can reveal complete understanding – it takes multiple views. Computational modeling can be a very important approach. Organizational design includes the organization processes of culture, decision making, information processing, CEO values, and style, at least. They conclude with a challenge for a new era in organization studies which upholds the rigor of scientific inquiry and embrace multiple perspectives and approaches. In a very short life, *Organization Science* has answered the challenge.

The focus here has been on the information processing view of organization studies, i.e., the nerve system and brain functions of organization. Other perspectives can be found in the journals *Management Science, Organization Science*, and *Computational & Mathematical Organization Theory*, among others.

See

- ▶ Computational Organization Theory
- ▶ Decision Making and Decision Analysis
- ▶ Economics and Operations Research

References

Argote, L., Beckman, S. L., & Epple, D. (1990). The persistence and transfer of learning in industrial settings. *Management Science, 36*, 140–154.

Arrow, K. J. (1974). *The limits of organization*. New York: W.W. Norton.

Baligh, H. H. (2005). *Organization structures: Theory and design, analysis and prescription*. New York: Springer.

Baligh, H. H., Burton, R. M., & Obel, B. (1996). Organizational consultant: Creating a useable theory for organizational design. *Management Science, 42*, 1648–1662.

Burton, R. M., & Obel, B. (1980). A computer simulation test of the M-form hypothesis. *Administrative Science Quarterly, 25*, 457–466.

Burton, R. M., & Obel, B. (1984). *Designing efficient organizations: Modelling and experimentation*. Amsterdam: North-Holland.
Burton, R. M., & Obel, B. (1988). Opportunism, incentives and the M-form hypothesis. *Journal of Economic Behavior and Organization, 10*, 99–119.
Burton, R. M., & Obel, B. (1995). The validity of computational models in organization science: From model realism to purpose of the model. *Computational and Mathematical Organization Theory, 1*(1), 57–71.
Burton, R. M., & Obel, B. (1998, 1995, 2004). *Strategic organizational diagnosis and design: The dynamics of fit*. Boston: Kluwer Academic Publishers.
Burton, R. M., & Obel, B. (2011). Computation modeling for what-is, what-might-be, what-should-be studies – And triangulation. *Organization Science*, *22*(5), 1195–1202.
Carley, K. (1995). Computational and mathematical organization theory: Perspectives and directions. *Computational and Mathematical Organization Theory, 1*, 39–56.
Carley, K., Kjaer-Hasen, J., Newell, A., & Prietula, M. (1992). Plural-soar: A proglegomenon to artificial agents and organization behavior. In M. Masuch & M. Waglien (Eds.), *Artificial intelligence in organization and management theory*. New York: North Holland.
Carley, K., & Lin, Z. (1995). A theoretical study of organizational performance under information distortion. *Management Science, 43*, 976–997.
Carley, K. M., & Prietula, M. J. (Eds.). (1993). *Computational organization theory*. Hinsdale: Lawrence Erlbaum.
Carley, K., & Prietula, M. J. (1998). WebBots, trust, and organizational science. In M. J. Prietula, K. M. Carley, & L. Gasser (Eds.), *Simulating organizations: Computational models of institutions and groups*. Menlo Park/Cambridge: AAAI Press/The MIT Press.
Carroll, T. N., Gormley, T. J., Bilardo, V. J., Burton, R. M., & Woodman, K. L. (2006). Designing a new organization at NASA: An organization design process using simulation. *Organization Science, 17*(2), 202–214.
Cohen, M. D., March, J. G., & Olsen, P. J. (1972). A garbage can model of organizational choice. *Administrative Science Quarterly, 17*, 1–25.
Cyert, R. M., & March, J. G. (1963). *A behavioral theory of the firm*. Englewood Cliffs: Prentice-Hall.
Daft, R. L., & Lewin, A. Y. (1990). Can organization studies begin to break out of the normal science straitjacket? An editorial essay. *Organization Science, 1*, 1–9.
Daft, R. L., & Lewin, A. Y. (1993). Where are the theories for the 'new' organizational forms? An editorial essay. *Organization Science*, *4*.
Eisenhardt, K. M. (1989a). Agency theory: An assessment and review. *Academy Management Review, 14*(1), 57–74.
Eisenhardt, K. M. (1989b). Making fast strategic decisions in high-velocity environments. *Academy of Management Journal, 32*, 543–576.
Ethiraj, S. K., & Levinthal, D. A. (2009). Hoping for A to Z while rewarding only A: Complex organization and multiple goals. *Organization Science, 20*(1), 4–21.
Galbraith, J. R. (1974). Organizational design: An information processing view. *Interfaces, 4*, 28–36.
Galbraith, J. R. (1995). *Designing organizations*. San Francisco: Jossey-Boss.
Jin, Y., & Levitt, R. E. (1996). The virtual design team: A computational model of project organization. *Computational and Mathematical Organization Theory, 2*, 171–196.
Koza, M. P., & Lewin, A. Y. (1998). The co-evolution of strategic alliances. *Organization Science, 9*, 1–10.
Levitt, R. E., Cohen, G. P., Kunz, J. C., Nass, C. I., Christiansen, T., & Jin, Y. (1994). The virtual design team: Simulating how organization structure and information processing tools affect team performance. In K. Carley & M. Prietula (Eds.), *Computational organization theory*. Hillsdale: Lawrence Erlbaum.
MacKenzie, K. D. (1991). *The organizational hologram: The effective management of organizational change*. Norwell: Kluwer Academic Publishers.
March, J. G. (1991). Exploration and exploitation in organizational learning. *Organization Science, 2*(1), 71–87.
March, J. G., & Simon, H. A. (1958). *Organizations*. New York: Wiley.
Marschak, J., & Radner, R. (1972). *Economic theory of teams*. New Haven: Yale University Press.
Mihm, J., Loch, C., Wilkinson, D., & Huberman, B. (2010). Hierarchical structure and search in complex organizations. *Management Science, 56*, 831–848.
Milgrom, P., & Roberts, J. (1992). *Economics, organization and management*. Englewood Cliffs: Prentice-Hall.
Prietula, M. J., Carley, K. M., & Gasser, L. (Eds.). (1998). *Simulating organizations: Computational models of institutions and groups*. Menlo Park/Cambridge: AAAI Press/The MIT Press.
Scott, W. R., & Davis, G. F. (2006). *Organizations and organizing: Rational, natural, and open systems perspectives*. Englewood Cliffs: Prentice-Hall.
Siggelkow, N., & Levinthal, D. A. (2003). Temporarily divide to conquer: Centralized, decentralized, and reintegrated organizational approaches to exploration and adaptation. *Organization Science, 14*(6), 650–669.
Siggelkow, N., & Rivkin, J. W. (2005). Speed and search: Designing organizations for turbulence and complexity. *Organization Science, 16*(2), 101–122.

Origin Node

A node in a network through which goods can enter the network. It is sometimes useful to define a special origin node through which all goods enter the network.

ORO

► Operations Research Office and Research Analysis Corporation

ORS

► Operational Research Society (ORS)

ORSA

► Operations Research Society of America (ORSA)

Out-of-Kilter Algorithm

A special primal-dual algorithm for solving minimum-cost network-flow problem.

See

► Minimum-Cost Network-Flow Problem

Output Process

The stochastic point or marked point process whereby the marks represent some aspect of the queueing customers or the state of the service stage or node and the points represent the times of customers leaving the server. This is contrasted with the departure process, which requires that the customers leave the entire queueing system for good. For example, in queues with feedback, the output process includes both the departure and feedback processes.

See

► Departure Process
► Networks of Queues
► Queueing Theory

Outside Observer Distribution

The probability distribution of the state of a queueing system at an arbitrarily chosen point in time, as opposed to what it would be at arrival or service-completion epochs. For queueing systems with a Poisson arrival process and exponentially distributed service times, all these steady-state distributions are the same.

See

► PASTA
► Queueing Theory

Overachievement Variable

A nonnegative variable in a goal-programming problem constraint that measures how much the left-hand side of the constraint is greater than the right-hand side.

See

► Goal Programming

Overflow Process

The stochastic marked point or point process of customers arriving to a queueing service center or node but not receiving service there. For example, the arrival process is composed of two stochastic processes, those gaining access to the server (i.e., the input process) and the overflow process of those not gaining access to the server. These distinctions are needed to model finite capacity-nodes.

See

► Arrival Process
► Input Process
► Queueing Theory

Overtaking

In queueing networks with alternate paths, the ability of customers leaving a node to use arcs to move ahead of customers not taking those arcs so as to arrive at a subsequent node ahead of customers they were previously behind. This can also occur for customers on the same path if a visited node has multiple servers, since a customer who started service later than other customers there could be served more quickly and thus pass some of those other customers.

See

► Networks of Queues

P

P_4

Partitioned preassigned pivot procedure. A procedure for arranging the basis matrix of a linear-programming problem into as near a lower triangular form as possible. Such an arrangement helps in maintaining a sparse inverse, given that the original data set for the associated linear-programming problem is sparse.

See

- ▶ Linear Programming
- ▶ Revised Simplex Method

Packing Problem

The integer-programming problem defined as follows:

$$\begin{aligned} &\text{Maximize} \quad \boldsymbol{c}^T\boldsymbol{x} \\ &\text{subject to} \quad \boldsymbol{E}\boldsymbol{x} \leq \boldsymbol{e} \end{aligned}$$

where the components of $\boldsymbol{E}$ are either 1 or 0, the components of the column vector $\boldsymbol{e}$ are all ones, and the variables are restricted to be either 0 or 1. The idea of the problem is to choose among items or combinations of items that can be packed into a container and to do so in the most effective way.

See

- ▶ Bin-Packing
- ▶ Set-covering Problem
- ▶ Set-partitioning Problem

Palm Measure

- ▶ Markovian Arrival Process (MAP)

Parallel Computing

Jonathan Eckstein
Rutgers, The State University of New Jersey, Livingston Campus, New Burnswick, NJ, USA

Introduction

Parallel computing is the use of a computer system that contains multiple, replicated arithmetic-logical units (ALUs), programmable to cooperate concurrently on a single task. Between 2000 and 2010, parallel computing underwent a sea change. Prior to this decade, the speed of single-processor computers advanced steadily, and parallel computing was generally employed only for applications requiring more computing power than a standard PC processor chip could deliver. Taking advantage of Moore's Law (Moore 1965), which predicts the steady increase in the number of transistors that can be packed into a given chip area, microprocessor manufacturers built processors that could execute a single stream of calculations at steadily increasing speeds. In the 2000–2010 decade, Moore's law continued to hold, but the way that chip builders used the ever-increasing number of transistors began to change. Applying ever-larger number of transistors to a single sequential stream of instructions began to encounter diminishing returns, and while smaller transistors enabled increasing clock

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

speeds, clock speeds are limited by energy consumption and heat dissipation issues. To use the ever-increasing number of available transistors, processor designers began placing multiple processor cores, essentially multiple processors, on each CPU chip. In the laptop and desktop markets, processors with four cores are now common, and CPU chips with only a single processing core are now rare. Thus, parallel processing is no longer only an effort to advance over the power available from mainstream computing platforms such as desktop and laptop computers; it has now become an integral part of such mainstream platforms.

Kinds of Parallel Computers

The taxonomy of Flynn (1972) classifies parallel computers as either SIMD (Single Instruction, Multiple Data) or MIMD (Multiple Instruction, Multiple Data). In SIMD architectures, a single instruction stream controls all the ALUs in a synchronous manner. In MIMD architectures, each ALU has its own instruction stream and its own instruction decoding hardware. The two approaches are not mutually exclusive: an approach sometimes called MSIMD (Multiple SIMD) combines multiple blocks of SIMD processors, with each block having its own instruction stream. There was active competition between SIMD and MIMD through the 1980s, but MIMD emerged as the clear winner in the 1990s. SIMD, however, has been staging a quiet resurgence in the form of GPUs (Graphics Processing Units), which typically have an MSIMD organization, as discussed below. Some confusion surrounds the term SIMD, as processor manufacturers also apply it to certain graphics-oriented special machine instructions that process blocks of data. These instructions are not necessarily completely parallel in the classic sense, but instead may simply take advantage of pipelining techniques to achieve higher utilization of ALU hardware than for standard scalar-operand instructions.

Another important distinction is between local and shared memory. In pure local-memory architectures, each processor has its own memory bank, and information may be moved between different processors only by messages passed through a communication network. On the other end of the spectrum are pure shared-memory designs, also called SMPs (Symmetric MultiProcessors), in which there is a single global memory bank that is equally accessible to all processors. Such designs provide performance and ease of programming for small numbers of processors, and are currently the most common, since they are used in desktop- and laptop-level multicore processor chips. In a more powerful server or workstation, two or more processor chips, each with four to six processor cores, share a single global memory. As with MIMD and SIMD, it is also possible to blend global and local memory approaches. For example, a system might be composed of dozens or hundreds of processing nodes, each node consisting of two to twelve processor cores sharing a single memory bank.

In large-scale systems without global memory, it is not generally practical to provide a dedicated connection between every pair of processors. Popular interconnection patterns include rings, grids, meshes, toroids, butterflies, and hypercubes. In academic circles, there has been an extensive debate on the merits of various interconnection topologies. However, the details of the interconnection pattern may not be critical for the kinds of parallel computers that currently exist, which generally range in size from a few processing nodes to thousands of nodes. At such scales, the critical considerations are the speed of the interconnection links, the overhead and latency associated with communication, and elementary non-interference properties. Non-interference means that sending a message from processor *A* to processor *B* should generally not interfere with processor *C* sending to processor *D*.

One way to construct a parallel computing system is simply to combine standard desktop or workstation computers, an approach known as a cluster or CoW (Cluster of Workstations). However, the local-area networks that usually connect such systems may significantly limit performance for some applications. Faster, special-purpose communication networks such as Myrinet or Infiniband may be used to improve the performance of dedicated cluster systems. Cooling and energy consumption can become significant limiting factors in constructing large CoW systems, and are also important design considerations in building higher-performance parallel supercomputers.

Another approach is to assemble ad hoc parallel systems from the background or off-hour capacity of collections of desktop computers, an approach known

as grid computing, a term meant to invoke an infrastructure of computing resources resembling the electric power grid. This approach requires no special hardware, but does need specialized software such as the Condor scheduling system (Litzkow et al. 1988). Communication between instruction streams can be particularly slow in such environments, however, and algorithms must be fault tolerant, i.e., resilient to processors unpredictably disappearing from the available pool, possibly in mid-computation.

A nascent trend is GPU computing (Owens et al. 2008). The demands of ever-more sophisticated animation, driven mainly from the personal computer gaming industry, have led graphics adaptors to evolve into special-purpose parallel computing engines, often far more powerful in terms of floating point operations per second (flops) than their host processors. Modern graphic processors typically have an MSIMD structure, consisting of independent blocks of SIMD processing units. Consumer level GPUs typically contain hundreds of ALUs, at a cost of a few dollars each. In GPU computing, one uses GPU hardware for other purposes than graphics processing. Graphics processors typically have a global memory with a high bandwidth connection to their processors, but this memory is often distinct from main CPU memory.

Programming Models

The primary distinction among styles of parallel computer programming is between data-parallel and control-parallel specification of concurrency. In the data-parallel model, also called SPMD (Single Program Multiple Data), the program essentially specifies a single thread of control, but individual statements may manipulate large arrays of data in an implicitly parallel way. For example, if A, B, and C are arrays of the same size of shape, the statement $A = B + C$ might replace each element of A by the sum of the corresponding elements of B and C. Responsibility for portions of each array is typically partitioned between multiple processors, so they divide the work and perform it concurrently. Communication in data-parallel programs is typically invoked through certain standard intrinsic functions. For instance, the expression $SUM(A)$ might represent the sum, across all processors, of all A's elements, computed by whatever algorithm is optimal for the current hardware.

Data-parallel languages were originally developed for SIMD architectures, but data-parallel and SIMD are not synonymous. MIMD systems may be programmed in a data-parallel manner when it suits the application at hand. Currently, the most prevalent data-parallel programming language is High Performance FORTRAN, or HPF (Koelbel et al. 1993). HPF has its roots in FORTRAN 90 (Metcalf and Reid 1990).

In control-parallel programming, the programmer specifies a distinct thread of control for each processing unit capable of one. Often, each processing unit has the same program, but takes a completely different path through it. If shared memory is available, threads may communicate via memory, using mechanisms called locks or critical sections to prevent simultaneous or inconsistent writes to the same location. Otherwise, threads must communicate by sending and receiving messages, a style called message passing. Note that shared-memory systems may also be programmed in a message-passing style, allowing for relatively straightforward migration to larger, non-shared-memory systems. Control parallel programs are typically written in standard sequential programming languages such as C, C++, or FORTRAN, handling messages and memory interlocks via special subroutine libraries. For message passing, the principle standardized, portable subroutine libraries are based on the MPI standard (Snir et al. 1996). At least three open-source implementations of MPI are available, and system manufacturers and integrators often provide their own optimized implementations.

For shared-memory programming, common standards include Posix threads (Butenhof 1997), in which a process spawns new threads by calling special operating system routines, and OpenMP (Dagum and Menon 1998), in which parallelism is specified by special compiler directives intermixed with standard code from the underlying C, C++, or FORTRAN language. Another alternative is Cilk (Blumofe et al. 1995; Leiserson 2009), which extends the standard C and C++ languages with new parallelism-specifying syntax.

It is generally accepted that control-parallel programs are harder to analyze, understand, develop, and debug than data-parallel programs, due to complicated race and deadlock conditions that can easily develop between threads. On the other hand,

the data-parallel programmer must sacrifice significant flexibility. Data parallelism is most readily applied to problems that require large, extremely regular array data structures. Irregular, sparse data structures are more the norm in operations research, and hence most of the field's successful applications of parallel computing have employed control parallelism.

Control-parallel programs can also exhibit nondeterminism: run twice on the same data, they may obtain different solutions or exhibit very different run times. Such effects occur because small differences in the timing of events may cause control-parallel programs to take complete different execution paths (serial programs that base branching decisions on measurements of clocks or timers may exhibit similar behavior). Such nondeterminism can typically be controlled and essentially eliminated, but sometimes at significant cost in performance.

Speedup, Efficiency and Scalability

If T_p is the time to solve a give problem using p processors, and T_1 is the time to solve the same problem with a single processor (using the best sequential algorithm, if it can be defined), then a key concept is speedup, defined to be $S_p = T_1/T_p$. Efficiency is then defined to be S_p/p, or, roughly speaking, the effectively used fraction of the raw computing power available. The main goal of parallel algorithm designers is to obtain *linear* speedups that grow roughly linearly with p, or, equivalently, efficiencies that do not approach 0 as p increases. In principle, speedups cannot be above linear and efficiencies cannot exceed 1; in practice, such effects can sometimes occur for specific problem instance because the "best" sequential algorithm for a particular problem is not always easily defined. In a search problem, for example, a run of a parallel algorithm might explore early in its history a portion of the search space that a standard serial implementation might not encounter until the later portions of its execution. If this portion of the search space contains the problem solution, an apparently superlinear speedup may result.

A key motivation for using parallel computing is to solve ever-larger problems. Thus, rather than concerning oneself with obtaining very large speedups for a fixed-size problem, it may be more important to study the effect on total solution time as the problem data and number of processors grow in some proportional or related way. This concept is called scalability (Kumar and Gupta 1994).

Applications in Operations Research

Parallel computing is taking an increasing role in operations research, but it has not had nearly the effect on the practice of the field as it has, for example, in computational fluid dynamics. This phenomenon is due largely to the lack of efficient parallel methods for factoring and related operations on irregularly structured sparse matrices. Such operations are essential to the sparse active set and Newton methods that form the core of operations research's numerical optimization algorithms. However, successes have been reported for specially structured problems amenable to decomposition methods, including stochastic programming — see for example Gondzio and Grothey (2007) — and on dense problems. Parallelism has also proved very useful in branch-and-bound and related search algorithms, and in a variety of randomized algorithms.

Currently, the leading vendors of linear/integer-programming software all offer some form of parallel branch-and-cut implementation for solving mixed integer programs; such implementations are typically for shared-memory systems; some are deterministic, others nondeterministic, and some offer the option of either a deterministic or nondeterministic mode. Some software vendors also offer parallel interior point linear-programming software, although speedups in pure linear programming are less dependable than for branch and bound.

Parallel open-source software for operations research operations research is becoming increasingly available. Several projects in the COIN-OR collection (Lougee-Heimer 2003) are aimed at parallel computing (typically through MPI), and several others offer the option of parallel execution.

Simulation applications with many independent trials or scenarios are also natural applications for parallel computing. A general principle seems to be that one should take advantage of problem structure to localize troublesome operations, most typically sparse matrix factorization, onto individual processors.

Another approach is to try radically new algorithms that avoid such operations completely, and are highly parallelizable. One should remember, however, that parallelism is not a panacea that can easily make inappropriate or "brute force" methods competitive.

Early references on the relationships between parallel computing and OR/MS include Barr and Hickman (1993) and Eckstein (1993).

See

- ▶ Integer and Combinatorial Optimization
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Stochastic Programming

References

Barr, R. S., & Hickman, B. L. (1993). Reporting computational experiments with parallel algorithms: Issues, measures and experts = opinions. *ORSA Journal of Computing, 5*, 2–18.

Bertsekas, D. P., & Tsitsiklis, J. (1989). *Parallel and distributed computation: Numerical methods*. Englewood Cliffs, NJ: Prentice-Hall.

Blumofe, R. D., Joerg, C. F., Kuszmaul, B. C., Leiserson, C. E., Randall, K. H., Zhou, Y. (1995). Cilk: An efficient multithreaded runtime system. *Proceedings of the Fifth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, Santa Barbara, California, 207–216.

Butenhof, D. R. (1997). *Programming with Posix threads*. Boston, MA: Addison-Wesley.

Dagum, L., & Menon, R. (1998). OpenMP: An industry standard API for shared-memory programming. *IEEE Computational Science and Engineering, 5*, 46–55.

Eckstein, J. (1993). Large-scale parallel computing, optimization, and operations research: A survey. *ORSA Computer Science Technical Section Newsletter*, 14(2), 1, 8–12.

Flynn, M. J. (1972). Some computer organizations and their effectiveness. *IEEE Transactions on Computers, C-21*, 948–960.

Gondzio, J., & Grothey, A. (2007). Parallel interior-point solver for structured quadratic programs: Application to financial planning problems. *Annals of Operations Research, 152*, 319–339.

Kindervater, G. A. P., & Lenstra, J. K. (1988). Parallel computing in combinatorial optimization. *Annals of Operations Research, 14*, 245–289.

Koelbel, C. H., Loveman, D. B., Schreiber, R. S., Steele, G. L., Zosel, M. E. (1993). *The high performance Fortran handbook*. Cambridge, MA: MIT Press.

Kumar, V., & Gupta, A. (1994). Analyzing scalability of parallel algorithms and architectures. *Journal of Parallel and Distributed Computing, 22*, 379–391.

Leighton, F. T. (1991). *Introduction to parallel algorithms and architectures: Arrays, trees, and hypercubes*. San Mateo, CA: Morgan Kaufmann.

Leiserson, C. E. (2009). The CILK++ concurrency platform. *Proceedings of the 46th Annual Design Automation Conference*, ACM, San Francisco, California, 522–527.

Litzkow, M. J., Livny, M., & Mutka, M. W. (1988). Condor-a hunter of idle workstations. *Proceedings of the 8th International Conference on Distributed Computing Systems*, IEEE, San Jose, California, 104–111.

Lougee-Heimer, R. (2003). The common optimization interface for operations research: Promoting open-source software in the operations research community. *IBM Journal of Research and Development, 47*, 57–66.

Metcalf, M., & Reid, J. (1990). *Fortran 90 explained*. Oxford, UK: Oxford University Press.

Moore, G. (1965). Cramming more components onto integrated circuits. *Electronics, 38*(8), 114–117.

Owens, J. D., Houston, M., Luebke, D., Green, S., Stone, J. E., Phillips, J. C. (2008). GPU computing. *Proceedings IEEE*, 96, 879–899.

Snir, M., Otto, S. W., Huss-Lederman, S., Dongarra, J., Kowalik, J. S. (1996). *MPI: The complete reference*. Cambridge, MA: MIT Press.

Zenios, S. A. (1994). Parallel and supercomputing in the practice of management science. *Interfaces, 24*, 122–140.

Parameter

A quantity appearing in a mathematical model that is subject to controls beyond those affecting the decision variables.

Parameter-Homogeneous Stochastic Process

A stochastic process in which distribution properties between the two index parameter points t_1 and t_2, $t_1 \leq t_2$, depend only on the difference $t_2 - t_1$, and not on the specific values of t_1 and t_2. In the many applications where the parameter set is time, whether discrete or continuous, it is called a time-homogeneous stochastic process.

Parametric Bound

An optimal value function or solution point bound as a function of problem parameters.

Parametric Linear Programming

In the general linear-programming problem of

$$\begin{aligned} \text{Minimize} \quad & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq 0 \end{aligned}$$

it is often appropriate to study how the optimal solution changes when some of the data are functions of a single parameter λ. Most mathematical programming systems allow parametric analysis of the cost coefficients (PAROBJ), the right-hand-side elements (PARARHS), joint analysis of the objective function and right-hand-side elements (PARARIM), and the parametric analysis of the data in a row (PARAROW).

Parametric Programming

Tomas Gal
Fern Universität in Hagen, Hagen, Germany

Introduction

The meaning of a parameter as used here is best explained by a simple example. Recall that a parabola can be expressed as follows: $y = ax^2$, $a \neq 0$. Setting $a = 1$, a parabola is obtained that has a different shape from the parabola when setting, for example, $a = 5$. In both cases, however, there are parabolas that obey specific relationships; only the shapes are different. Hence, the parabola $y = ax^2$ describes a family of parabolas and the parameter a specifies the shape.

Consider the general mathematical-programming problem:

$$\text{Max } z = f(\boldsymbol{x}) \tag{1}$$

$$\text{subject to } g(\boldsymbol{x}) \leq 0 \tag{2}$$

Introducing one or more parameters into f or g, the model stays the same, but for each value of the parameter(s) one obtains a specific problem.

In setting up a mathematical optimization model, one of the first tasks is to collect data. The collected data might, however, be inaccurate, be of a stochastic character, be uncertain or be deficient in other ways. Therefore, it is appropriate to introduce parameters that enable to analyze the influence of specific data elements on the optimal solution. This can be done by:

1. Introducing the parameter(s) at the beginning when setting up the model, or
2. Introducing the parameter(s) after an optimal solution has been found.

The latter case is called postoptimal analysis (POA) and is applied much more frequently than the first case.

Postoptimal analysis is a very important tool that should be used in the framework of a good report generator (Gal 1993). The corresponding decision maker (DM) would then have information with which the DM can select a firm optimum. POA consists of several analyses, the most important of which is sensitivity analysis (SA). A sort of extended SA is parametric programming (PP). In nonlinear programming, SA corresponds to perturbation analysis, in which, after having found an optimal solution, some of the initial data are perturbed and the influence of the perturbation on the outcome is analyzed (Drud and Lasdon 1997).

Historical Sketch

Advanced methods for SA and PP for linear programming have been developed. In the 1950s, Orchard-Hays (in his master's thesis), Manne (1953), Saaty and Gass (1954), Gass and Saaty (1955) published the first works on parametric programming. By the end of the 1960s, the first monograph on parametric programming appeared (Dinkelbach 1969), followed by the monograph and book by Gal (1973, 1979). In 1979, the first Symposium on Data Perturbation and Parametric Programming was organized by A.V. Fiacco in Washington, D.C., with such a symposium being held every year since. (From 1999, Adi Ben Israel has been the organizer). Several monographs (Bank et al. 1982; Guddat et al. 1991) and special journal issues have been published in the 1970s and 1980s. More details on the history of PP are given in Gal (1980, 1983). A bibliography with over 1,000 items is given in Gal (1994b); see also Gal and Greenberg (1997).

Postoptimal Analysis

Assume that the mathematical optimization model under consideration is a linear program of the form:

$$\text{Max } z = \boldsymbol{c}^T \mathbf{x} \tag{3}$$

$$\text{subject to } \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \ \ \boldsymbol{x} \geq \boldsymbol{0} \tag{4}$$

where $\boldsymbol{c}$ is an n-vector of objective function coefficients (OFC) c_j, x is an n-vector of the decision variables x_j, $\boldsymbol{A}$ is an $m \times n$ matrix of the technological coefficients a_{ij}, $m < n$, $\boldsymbol{b}$ is an m vector of the right-hand-side (RHS) elements b_i. All vectors are column vectors.

Suppose that the problem defined by (3) and (4) has an optimal basic feasible solution $x_{\boldsymbol{B}} = \boldsymbol{B}^{-1}\boldsymbol{b}$, where $\boldsymbol{B}^{-1}$ is the inverse of the $m \times m$ basic matrix $\boldsymbol{B}$ (the basis) consisting of m linearly independent columns of $\boldsymbol{A}$. Here, $x_{\boldsymbol{B}}$ is an m-dimensional solution vector. This means that the following solution elements and simplex method elements are determined:

1. The maximal value of the objective function (OF), $z_{\max}$,
2. The values of the basic variables $x_i, i = 1,\ldots, m$, and
3. The reduced costs $d_j = z_j - c_j, j = 1,\ldots, n$.

In the framework of POA, an evaluation of the above solution elements is to be performed. This means that the DM is provided with information about the meaning of the values of the basic variables, the DM is told which resources are used and are critical (values of slack variables), and interpret the values of the opportunity costs and shadow prices. It is also possible to carry out a suboptimal analysis, that is, show the DM what happens if one or several nonbasic variables were introduced into the solution at a positive level.

Sensitivity Analysis

The POA would continue by performing a SA with respect to the OF and the RHS. This analysis is usually a part of the solution output for just about all linear-programming software. It is called OFC-ranging and RHS-ranging, respectively. Behind such analyses is the introduction of a scalar parameter, t or λ, in the form

$$c_j(t) = c_j + t, \ j \text{ fixed} \tag{5}$$

or

$$b_i(\lambda) = b_i + \lambda, \ \ i \text{ fixed} \tag{6}$$

SA finds a critical interval T_j or Λ_i, such that for all $t \in T_j$ or $\lambda \in \Lambda_i$, respectively, the (found) optimal basis $\boldsymbol{B}$ remains the same (so called optimal basis invariancy. For other kinds of invariancies see, e.g., Hladik 2010; Hadigheh et al. 2007). The critical values, that is, the upper and lower bounds of the critical interval can be easily determined by certain formulas (Gass 1985). A change in a RHS element b_i causes, in general, the values of the basic variables and the value of $z_{\max}$ to change, while a change in an OFC c_j causes, in general, the values of the reduced costs and the value of $z_{\max}$ to change. Such information is of great value to the DM. An assumption of this type of SA is that to investigate how the optimal solution would vary with respect to a change in one data element, while holding all other data fixed. Analysis of multiple changes can be done in a limited manner by the techniques of the hundred percent rule (Bradley et al. 1977) and tolerance analysis (Ashram 2007; Filippi 2005; Hladik 2008a, b; Wendell 1985, 2004).

Parametric Analysis

For an element b_i of the RHS, the question is asked: for what range of values of the parameter λ in (6) does there exist an optimal solution to (3) and (4)? Given such values, one can move from the original optimal basis and generate a sequence of optimal bases, with each basis associated with a critical interval of the parameter. Such an analysis provides the DM with a full range of possible solutions from which a subset of optimal solutions appropriate for the given problem can be selected. The DM then chooses a certain value of the parameter and, thus, a corresponding optimal solution for the parametric range of b_i (λ).

Note that a similar analysis can be performed with respect to the parametric OFC, as given by (5). Moreover, taking into account the possibility that a parameter introduced in the RHS may influence some (or several) OFC or vice versa, it is possible to perform a RIM parametric analysis, that is, find a sequence of optimal bases to each of which a critical interval for the RHS-and for the OFC-parameters are

associated simultaneously. Standard RHS, OFC and RIM parametric analysis procedures are usually included in linear-programming software.

It is also possible to perform a sensitivity or parametric analysis with respect to the elements a_{ij} of the matrix $\boldsymbol{A}$. The corresponding procedures are, unfortunately, not incorporated into linear-programming software as the underlying formulas are a bit too complex. However, some software enables one to compute a series of linear programs in each of which slightly changed values of the $\{a_{ij}\}$ are chosen.

Up to now, the simplest parametric case having one parameter with a coefficient equal to 1 has been discussed. The above cases can, however, also be carried out when:

(i) A scalar parameter is introduced into several elements of the RHS and/or OFC with coefficients which differ from 1, and

(ii) A parameter-vector (vector of parameters) is introduced into several elements of the RHS and/or OFC with their respective coefficients different from 1.

As far as case (i) is concerned, to each optimal basis a critical interval is associated. In case (ii), each optimal basis is associated with a higher dimensional convex polyhedral set of parameters. In the RIM case, each optimal basis is associated with a higher dimensional interval, a box, provided that the parameters in the RHS and OFC are independent from each other. The larger the number of parameters in the parameter-vector, the more difficult it is to interpret the results and for the DM to find an appropriate optimal basis. In such cases, an interactive approach is recommended in which the parametric specialist helps the DM to select an appropriate solution.

Applications

There are two kinds of uses of PP:

1. Introducing parameters into various classes of mathematical-programming problems for solving these problems via parameterization; and
2. Practical applications.

As to (1), the introduction of parameters helps to solve problems from the areas of nonconcave mathematical programming, decomposition, approximation, and integer programming. Also, note that by replacing the OFC in (3) and (4) with a matrix $\boldsymbol{C}$ times a parameter-vector $\boldsymbol{t}$ the following problem is obtained

$$\begin{aligned} &\text{Max } z = (\boldsymbol{C}^T\boldsymbol{t})\boldsymbol{x}, \\ &\text{subject to } \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

which is a scalarized version of a linear multiobjective-programming problem (Steuer 1986). Methods for solving the corresponding homogeneous multi-parameter-programming problem provide a procedure to determine the set of all efficient solutions of the corresponding multiobjective problem (Gal 1994b).

As to (2), SA and/or PP has been used in the pipeline industry, in capital budgeting, for farm decision making, refinery operations, for return maximization in an enterprise, and a number of other applications (Gal 1994b).

SA and PP in Other Fields

Theoretical and methodological works have been published about SA and/or PP in linear and nonlinear complementarity problems, control of dynamic systems, fractional programming, geometric programming, integer and quadratic programming problems, transportation problems. A more detailed survey with corresponding references is given in Gal (1994b) (1988), see also, e.g., Ravi and Wendell (1988), Hladik (2008b), Dawande and Hooker (2000), Faisca et al. (2009), Kheirfam (2010).

Degeneracy

Recall that a basic feasible solution to a linear-programming problem is called primal degenerate when at least one element of this solution equals zero. The corresponding extreme point of the feasible set, that is, of the convex polyhedron, is then also called degenerate. Degeneracy causes various kinds of efficiency and convergence problems and special precautions must be taken when performing SA for a degenerate extreme point. Degeneracy influences even POA, especially the determination of

opportunity costs and shadow prices. When performing SA, the main rule – determining the critical interval such that the original optimal basis does not change – is no longer valid because for a degenerate solution many bases are associated with it. A theoretical discussion of this problem is given in Kruse (1986), a bibliography is found in Gal (1994a). Note that standard software analysis for RHS-or OFC-ranging yield false results when degeneracy is involved.

Concluding Remarks

For linear programming and related mathematical areas, SA and PP have become important tools for analyzing variations in initial data, for obtaining better insight into and gaining more information about the related mathematical model, for improving understanding of model building in general, and as aids in solving a wide range of mathematical problems.

See

- ▶ Degeneracy
- ▶ Degeneracy Graphs
- ▶ Linear Programming
- ▶ Multiobjective Programming
- ▶ Perturbation Methods
- ▶ Sensitivity Analysis

References

Ashram, H. (2007). Construction of the largest sensitivity region for general linear programs. *Applied Mathematics and Computation, 189*, 1435–1447.

Bank, B., Guddat, J., Klatte, D., Kummer, B., & Tammer, T. (1982). *Nonlinear parametric optimization*. Berlin: Akademie Verlag.

Bradley, S. P., Hax, A. C., & Magnanti, T. L. (1977). *Applied mathematical programming*. Reading, MA: Addison-Wesley.

Dawande, M. W., & Hooker, J. N. (2000). Inference-based sensitivity analysis for mixed integer/linear programming. *Operations Research, 48*, 623–634.

Dinkelbach, W. (1969). *Sensitivitätsanalysen und parametrische Programmierung*. Berlin: Springer Verlag.

Drud, A. S., & Lasdon, L. (1997). Nonlinear programming. In T. Gal & H. J. Greenberg (Eds.), *Advances in sensitivity analysis and parametric programming*. Norwell, MA: Kluwer.

Faisca, N. P., Kosmidis, V. D., Rustem, B., & Pistikopoulos, E. N. (2009). Global optimization of multi-parametric MILP problems. *Journal Global Optimization, 45*(1), 131–151.

Filippi, C. (2005). A fresh view on the tolerance approach to sensitivity analysis in linear programming. *European Journal of Operational Research, 167*, 1–19.

Gal, T. (1973). *Betriebliche Entscheidungsprobleme, Sensitivitätsanalyse und parametrische Programmierung*. Berlin: W. de Gruyter.

Gal, T. (1979). *Postoptimal analyses, parametric programming and related topics*. New York: McGraw Hill.

Gal, T. (1980). A 'historiogramme' of parametric programming. *Journal of the Operational Research Society, 31*, 449–451.

Gal, T. (1983). A note on the history of parametric programming. *Journal of the Operational Research Society, 34*, 162–163.

Gal, T. (1993). Putting the LP survey into perspective. *OR/MS Today, 19*(6), 93.

Gal, T. (1994a). Selected bibliography on degeneracy. *Annals Operations Research*.

Gal, T. (1994b). *Postoptimal analyses and parametric programming*. Berlin: W. de Gruyter. Revised and updated edition.

Gal, T., & Greenberg, H. J. (Eds.). (1997). *Advances in sensitivity analysis and parametric programming*. Norwell, MA: Kluwer.

Greenberg, H. J. (1993). *A computer-assisted analysis system for mathematical programming models and solutions: A user's guide for ANALYZE*. Norwell, MA: Kluwer.

Gass, S. I. (1985). *Linear programming* (5th ed.). New York: McGraw-Hill.

Gass, S. I., & Saaty, T. L. (1955). The parametric objective function. *Naval Research Logistics Quarterly, 2*, 39–45.

Guddat, J., Guerra Vazquez, F., & Jongen, H. T. (1991). *Parametric optimization: Singularities, path following and jumps*. Stuttgart/New York: B. G. Teubner/Wiley.

Hadigheh, A. G., Mirnia, K., & Terlaky, T. (2007). Active constraint set invariancy sensitivity analysis in linear optimization. *JOTA, 133*, 303–315.

Hladik, M. (2008a). Additive and multiplicative tolerance in multiobjective linear programming. *Operations Research Letters, 36*, 393–396.

Hladik, M. (2008b). Computing the tolerance in multiobjective linear programming. *Optimization Methods and Software, 23*, 731–739.

Hladik, M. (2010). Multiparametric linear programming: Support set and optimal partition invariancy. *European Journal of Operational Research, 202*, 25–31.

Kheirfam, B. (2010). Sensitivity analysis in multi-parametric strictly convex quadratic optimization. *Matem. Vesnik, 62*, 95–107.

Kruse, H.-J. (1986). *Degeneracy graphs and the neighborhood problem* (Lecture Notes in economics and mathematical systems No. 260). Berlin: Springer Verlag.

Manne, A. S. (1953). *Notes on parametric linear programming, RAND Report P-468*. Santa Monica, CA: The Rand Corporation.

Ravi, N., & Wendell, R. E. (1988). Tolerance approach to sensitivity analysis in network linear programming. *Networks, 18*, 159–181.

Saaty, T. L., & Gass, S. I. (1954). The parametric objective function, Part I. *Operations Research, 2*, 316–319.
Steuer, R. E. (1986). *Multiple criteria optimization: Theory, computation, and application*. New York: Wiley.
Wendell, R. E. (1985). The tolerance approach to sensitivity analysis in linear programming. *Management Science, 31*, 564–578.
Wendell, R. E. (2004). Tolerance sensitivity and optimality bounds in linear programming. *Management Science, 50*, 797–803.

Parametric Solution

A solution expressed as a function of problem parameters.

Pareto-Optimal Solution

If a feasible deviation from a solution to a multiobjective problem causes one of the objectives to improve while some other objective degrades, the solution is termed a Pareto-optimal. Such a solution is also called an efficient or nondominated solution.

See

► Efficient Solution

Partial Balance Equations

In Markov chain models of queueing networks, a subset of the global balance equations that may be satisfied at a node (station), i.e., a balance of mean flow rates or probability flux. Also known as local balance equations, falling between global balance equations and detailed balance equations.

See

► Detailed Balance Equations
► Global Balance Equations
► Markov Chains
► Networks of Queues
► Queueing Theory

Partial Pricing

When determining a new variable to enter the basis by the simplex method, it is somewhat computationally inefficient to price out all nonbasic columns, as is the way of the standard simplex algorithm or its multiple pricing refinement. The scheme of partial pricing starts by searching the nonbasic variables in index order until a set of candidate vectors has been found. These vectors are then used as possible vectors to enter the basis, as is done in multiple pricing. After the candidate set is depleted, another set is found by searching the nonbasic vectors from the point where the first set stopped its search. The process continues in this manner by searching and selecting candidate sets until the optimal solution is found. Although the total number of iterations to solve a problem usually increases, computational time is saved by this type of pricing strategy.

See

► Simplex Method (Algorithm)

Partially Observed Markov Decision Processes

A Markov decision process (MDP) in which the state of the system cannot be fully or precisely observed, e.g., only part of the state is known and/or the state observation has some error. In principle, such a model can be converted to a fully observed MDP by introducing an "information" or "belief" state that may be infinite dimensional, corresponding to a probability distribution over the original state.

See

► Dynamic Programming
► Markov Decision Processes

Particle Swarm Optimization

A population-based search approach for global optimization based on ideas from animal flocking.

See

- ▶ Ant Colony Optimization
- ▶ Metaheuristics
- ▶ Swarm Intelligence

References

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*, Vol. IV, pp. 1942–1948.

PASTA

Poisson Arrivals See Time Averages.

For a Poisson arrival process, the (limiting) fraction of arrivals that find (see) a process in some state equals the (limiting) overall fraction of time that the process is in that state (Wolff 1982, 1990).

See

- ▶ Poisson Arrivals

References

Wolff, R. W. (1982). Poisson arrivals see time averages. *Operations Research, 30*, 223–231.
Wolff, R. W. (1990). A note on PASTA and anti-PASTA for continuous-time Markov chains. *Operations Research, 38*, 176–177.

Path

A path in a network is a sequence of nodes and arcs that connect a designated initial node to a designated terminal node.

See

- ▶ Chain
- ▶ Cycle

Payoff Function

In a game, the mapping from the players' strategies (decisions, actions) to the gains and losses they receive. In a two-person finite action game, the payoff function is often depicted in the form of a matrix, with a single number for each matrix element in a zero-sum game.

In financial engineering, the mapping from the underlying asset(s) to the payout of a contingent claim or financial derivative.

See

- ▶ Financial Engineering
- ▶ Game Theory

Payoff Matrix

For a zero-sum, two-person game, the payoff matrix is an $m \times n$ matrix of real numbers with the entry a_{ij} representing the payoff to the maximizing player if the maximizing player plays strategy i and the minimizing player plays strategy j.

See

- ▶ Game Theory

PDA

Parametric decomposition approach.

See

- ▶ Production Management

PDF

Probability density function.

PDSA

Plan, do, study, act.

See

► Total Quality Management

Periodic Review

A type of inventory control policy in which the inventory position is assessed at the end of each of a prescribed number of discrete time periods, in contrast with continuous review, where the inventory position is monitored continuously so that orders can be placed at any time.

See

► Inventory Modeling

PERT

Program evaluation and review technique; an event-oriented, project-network diagramming technique used for planning and scheduling.

See

► Network Planning
► Program Evaluation and Review Technique (PERT)
► Project Management
► Research and Development

Perturbation

A change in a parameter, function or set.

Perturbation Analysis

Michael C. Fu
University of Maryland, College Park, MD, USA

Introduction

Perturbation analysis (PA) is a sample path technique for analyzing changes in performance measures of stochastic systems due to changes in system parameters. In terms of stochastic simulation, which is the main setting for PA, the objective is to estimate sensitivities of the performance measures of interest with respect to system parameters, preferably without the need for additional simulation runs over what is required to estimate the system performance itself. The primary application is gradient estimation during the simulation of discrete-event systems, e.g., queueing and inventory systems. Besides their importance in sensitivity analysis, these gradient estimators are a critical component in gradient-based simulation optimization methods.

Let $l(\theta)$ be a performance measure of interest with parameter (possibly vector) of interest θ, focusing on those systems where $l(\theta)$ cannot be easily obtained through analytical means and therefore must be estimated from sample paths, e.g., via stochastic simulation. Denote by $L(\theta, \omega)$ the sample performance obtained from a sample path realization ω such that $l(\theta) = E[L(\theta, \omega)]$. Although the assumption here is that the performance measure is an expectation, PA has also been applied more recently to quantiles (Hong 2009; Fu et al. 2009). The goal of PA is to efficiently estimate the effects on l of a perturbation $\theta \to \theta + \Delta\theta$, using information from a sample path ω at θ. PA addresses two different types of problems:

- $\Delta\theta \to 0$: estimating the gradient $\nabla l(\theta)$, when l is differentiable in θ.
- $\Delta\theta \neq 0$: estimating changes due to a finite perturbation, i.e., $l(\theta + \Delta\theta)$.

In the former case, no perturbation is ever actually introduced into the system (or simulation), although the idea of a perturbation may be employed as a heuristic tool in preliminary analysis.

Brief Taxonomy

To sort out the abundance of acronyms in the PA field, a brief definition of each corresponding approach is provided here, accompanied with at least one reference. Among gradient estimation techniques, the most well-known is infinitesimal perturbation analysis (IPA), which simply uses the sample derivative $dL/d\theta$ to estimate $dl/d\theta$. It is straightforward to implement and very computationally efficient; however, as shall be discussed shortly in more detail, its applicability is not universal. The books by Ho and Cao (1991), Glasserman (1991), and Cao (1994) cover IPA in detail. A very general and well-developed extension of IPA is smoothed perturbation analysis (SPA), based on the ideas of conditional expectation (Gong and Ho 1987) Although its applicability is quite broad, its implementation is usually very problem dependent. The book by Fu and Hu (1997) covers this method in full generality. Other gradient estimation techniques include rare perturbation analysis (RPA), originally based on the thinning of point processes (Brémaud and Vázquez-Abad 1992); structural IPA (SIPA), dealing specifically with structural parameters (Dai and Ho 1995); discontinuous perturbation analysis (DPA), based on the use of generalized functions (the Dirac-delta function) to model discontinuities in the sample performance function (Shi 1996); and augmented IPA (APA), another extension of IPA different from SPA (Gaivoronski et al. 1992). Techniques to estimate the effect of a finite perturbation in the parameter include finite perturbation analysis (FPA) – Ho et al. (1983); extended perturbation analysis (EPA) – Ho and Li (1988); and the augmented chain method−Cassandras and Strickland (1989). A related technique is the standard clock (SC) method, based on the uniformization of Markov chains (Vakili 1991). The books by Ho and Cao (1991) and Cassandras and Lafortune (2008) provide further references. This entry focuses on the gradient estimation techniques IPA and SPA, the most well-known and developed of the PA techniques.

Infinitesimal Perturbation Analysis

The applicability of IPA is illustrated through the use of some simple examples, at the same time contrasting the approach with the likelihood ratio/score function (LR/SF) and weak derivative (WD) estimators. Consider first the expectation of a single positive random variable X, written in two forms:

$$\begin{aligned} E[X] &= \int_0^\infty x f(x;\theta)dx \\ &= \int_0^1 X(\theta;u)du, \end{aligned}$$

where f is the PDF of X. In the first interpretation, the parameter appears inside the density, whereas in the second interpretation it appears inside the random variable defined on an underlying $U(0,1)$ random number. For example, the latter could be the inverse transform $X = F^{-1}$, where F is the CDF of X.

Differentiating $E[X]$, assuming the interchange of expectation and differentiation is permissible (via the dominated convergence theorem),

$$\frac{dE[X]}{d\theta} = \int_0^\infty x\frac{df(x;\theta)}{d\theta}dx \quad (1)$$

$$= \int_0^1 \frac{dX(\theta;u)}{d\theta}du. \quad (2)$$

Notice, however, that the conditions for the exchange will be quite different for the two interpretations. In the first interpretation, corresponding to the LR/SF and WD estimators, the conditions will be placed on the underlying density; in the case of discrete-event stochastic simulation, this means the input distributions. Since the input distributions must be known in order to perform the simulation, it is relatively easy to check the conditions. In the second interpretation, corresponding to PA estimators, the conditions will be placed on the sample performance function that is usually defined on an output stochastic process of the system.

As an example, consider an exponential random variable X with mean θ. Then $E[X] = \theta$ and $dE[X]/d\theta = 1$. The respective PDF and one random variable representation are given by

$$\begin{aligned} f(x;\theta) &= \frac{1}{\theta}e^{-x/\theta}1\{x > 0\}, \\ X(\theta;u) &= -\theta \ln u, \end{aligned}$$

where $1\{\cdot\}$ denotes the indicator function. Differentiating,

$$\begin{aligned}\frac{df(x;\theta)}{d\theta} &= \left[\frac{x}{\theta^2}\frac{1}{\theta}e^{-x/\theta} - \frac{1}{\theta^2}e^{-x/\theta}\right]1\{x>0\}\\ &= f(x;\theta)\left[\frac{x}{\theta^2}-\frac{1}{\theta}\right]\\ &= \frac{1}{\theta e}\left[\frac{e}{\theta}\left(1-\frac{x}{\theta}\right)e^{-x/\theta}1\{0<x\le\theta\}\right.\\ &\quad \left.-\frac{e}{\theta}\left(\frac{x}{\theta}-1\right)e^{-x/\theta}1\{x>\theta\}\right],\\ \frac{dX(\theta;u)}{d\theta} &= -\ln u = \frac{X(\theta;u)}{\theta}.\end{aligned}$$

The last expression for the derivative of the density (which is itself *not* a density) expresses the quantity as the difference of two densities multiplied by a constant, known as a weak derivative representation; Fu (2006, 2008) for references. Substituting each of the three expressions into the corresponding equations (1) or (2), yields three unbiased derivative estimators:

$$\begin{aligned}\text{LR/SF}: &\frac{X}{\theta}\left(\frac{X}{\theta}-1\right),\\ \text{WD}: &\frac{1}{\theta e}\left[X^{(2)}-X^{(1)}\right],\\ \text{IPA}: &\frac{X}{\theta},\end{aligned}$$

where $X^{(1)}$ and $X^{(2)}$ are random variables with PDFs $\frac{e}{\theta}\left(\frac{x}{\theta}-1\right)e^{-x/\theta}$, $x>\theta$, and $\frac{e}{\theta}\left(1-\frac{x}{\theta}\right)e^{-x/\theta}$, $0<x\le\theta$, respectively.

Extending to a function of the underlying random variable,

$$\begin{aligned}\frac{dE[L(X)]}{d\theta} &= \int_0^\infty L(x)\frac{df(x;\theta)}{d\theta}dx\\ &= \int_0^1 \frac{dL}{dX}\frac{dX(\theta;u)}{d\theta}du.\end{aligned}$$

The conditions for interchanging expectation and differentiation are unaltered when differentiating the underlying density, since that portion remains unchanged, whereas they are more involved for the sample path derivative. Basically, for the chain rule to be applicable requires some sort of continuity to hold for the sample performance function with respect to the underlying random variable. This translates into requirements on the form of the performance measure and on the dynamics of the underlying stochastic system such that the interchange

$$\frac{dE[L]}{d\theta} = E\left[\frac{dL}{d\theta}\right] \tag{3}$$

holds. Roughly speaking, sample pathwise continuity of L with respect to θ will result in the interchange being valid. An important structural condition for determining the applicability of IPA for general discrete-event systems modeled as generalized semi-Markov processes is the commuting condition (Glasserman 1991).

Smoothed Perturbation Analysis

The main idea of smoothed perturbation analysis (SPA) is to use conditional expectation to smooth out discontinuities in L that cause IPA to fail. This is achieved by selecting a set of sample path quantities $\mathcal{Z}$, called the characterization, such that $E[L|\mathcal{Z}]$ – as opposed to L itself – will satisfy the interchange in (3):

$$\frac{dE[E[L|\mathcal{Z}]]}{d\theta} = E\left[\frac{dE[L|\mathcal{Z}]}{d\theta}\right].$$

Applying SPA is analogous to the variance reduction technique of conditional Monte Carlo, consisting of two main steps: choosing an appropriate $\mathcal{Z}$ and calculating $dE[L|\mathcal{Z}]/d\theta$. For generalized semi-Markov processes, as well as for other stochastic systems, this is fully explored in Fu and Hu (1997).

Queueing Example

IPA and SPA estimators are illustrated for a single-server, first come, first-served (FCFS) queue. Let A_n be the interarrival time between the $(n-1)$th and nth customer (i.i.d. with PDF f_1 and CDF F_1), X_n the service time of the nth customer (i.i.d. with PDF f_2 and CDF F_2), and T_n the system time (in queue plus in service) of the nth customer. Consider the case where θ is a parameter in the service time distribution, and the

sample performance of interest is the average system time over the first N customers $\overline{T}_N = \frac{1}{N}\sum_{n=1}^{N} T_n$. The system time of a customer for a FCFS single-server queue satisfies the well-known recursive Lindley equation:

$$T_{n+1} = X_{n+1} + (T_n - A_{n+1})^+. \tag{4}$$

The IPA estimator is obtained by differentiating (4):

$$\frac{dT_{n+1}}{d\theta} = \frac{dX_{n+1}}{d\theta} + \frac{dT_n}{d\theta} 1\{T_n \geq A_{n+1}\}, \tag{5}$$

where

$$\frac{dX}{d\theta} = -\frac{dF_2(X;\theta)/d\theta}{dF_2(X;\theta)/dX}.$$

For example, for scale parameters, such as if θ is the mean of an exponential distribution, $dX/d\theta = X/\theta$. Using the above recursion, the IPA estimator for the derivative of average system time is given by

$$\begin{aligned}\frac{d\overline{T}_N}{d\theta} &= \frac{1}{N}\sum_{n=1}^{N}\frac{dT_n}{d\theta} \\ &= \frac{1}{N}\sum_{m=1}^{M}\sum_{i=n_{m-1}+1}^{n_m}\sum_{j=n_{m-1}+1}^{i}\frac{dX_j}{d\theta},\end{aligned} \tag{6}$$

where M is the number of busy periods observed and n_m is the index of the last customer served in the mth busy period ($n_0 = 0$). Implementation of the estimator involves keeping track of two running quantities, one for (5) and another for the summation in (6); thus, the additional computational overhead is minimal, and *no alteration of the underlying simulation is required*. IPA is also applicable to multi-server queues and Jackson-like queueing networks (Jackson networks without the exponential distribution assumptions).

The implicit assumption used in deriving an IPA estimator is that small changes in the parameter will result in small changes in the sample performance. For example, small changes in the interarrival and service times lead to small changes in system times, as can be seen by the Lindley equation (4), but can lead to large changes in the derivative given by (5), due to the indicator function. In general, the interchange (3) will hold if the sample performance is continuous with respect to the parameter. For the Lindley equation, although T_{n+1} in (4) has a kink at $T_n = A_{n+1}$, it is still continuous at that point, which explains why IPA works. Unfortunately, the kink means that the derivative given by (5) has a discontinuity at $T_n = A_{n+1}$, so that IPA will fail for the second derivative.

For the FCFS single-server queue, SPA can be used to derive the following estimator for the second derivative of mean system time:

$$\begin{aligned}\left(\frac{d^2\overline{T}_N}{d\theta^2}\right)_{SPA} &= \frac{1}{N}\sum_{m=1}^{M}\sum_{i=n_{m-1}+1}^{n_m}\sum_{j=n_{m-1}+1}^{i}\frac{d^2X_j}{d\theta^2} \\ &\quad + \frac{1}{M}\sum_{m=1}^{M}\frac{f_1(T_{n_m})}{1-F_1(T_{n_m})}\left(\sum_{i=n_{m-1}+1}^{n_m}\frac{dX_i}{d\theta}\right)^2,\end{aligned}$$

where $d^2X/d\theta^2$ is well-defined when $F_2(X;\theta)$ is twice differentiable.

Inventory Example

IPA and SPA estimators are illustrated for a single-item periodic review (s, S) inventory system, in which once every period the inventory level is reviewed and, if necessary, orders are placed to replenish depleted inventory. An (s, S) ordering policy specifies that an order be placed when the level of inventory on hand plus that on order (known as inventory position) falls below the level s, and that the amount of the order be the difference between S and the present inventory position, i.e., order amounts are placed "up to S." For average inventory as the performance measure of interest, derivative estimators with respect to the policy parameters s and $q = S - s$ are provided. Note that the parameters in this example are structural, as opposed to distributional in the previous queueing example.

In the model considered, all excess demand is backlogged and eventually filled, and orders are immediately received (zero lead time), so that. inventory level and inventory position coincide. At the end of a period, demand is satisfied *before* the order placement decision is made. Let D_n be the demand in period n (i.i.d. with PDF f and CDF F), and V_n be the inventory level in period n after demand

satisfaction. This quantity satisfies a recursive equation somewhat analogous to the Lindley equation:

$$V_{n+1} = \begin{cases} V_n - D_{n+1} & \text{if } V_n \geq s, \\ S - D_{n+1} & \text{if } V_n < s. \end{cases} \quad (7)$$

The sample performance is the average inventory level over N periods given by $\overline{V}_N = \frac{1}{N}\sum_{n=1}^{N} V_n$.

From a sample path point of view, the key discrete event in the system is the ordering decision each period. A change in s, with q held fixed, has no effect on these decisions, so infinitesimal perturbations in s result in infinitesimal changes in the inventory level, and hence in the sample performance function $\overline{V}_N$. In particular, for a perturbation of size Δs (of any size, not necessarily infinitesimal), $V_n(s + \Delta s) = V_n(s) + \Delta s$, and hence $\partial \overline{V}_N / \partial s = 1$ is an unbiased estimator for $\partial E[\overline{V}_N]/\partial s$. Intuitively, the shape of sample paths are unaltered by changes in s if q is held constant; the entire sample path is merely shifted by the size of the change. The IPA estimator can also be obtained by simply differentiating the recursive relationship (7), noting that D_n does not depend on s or q:

$$\frac{dV_{n+1}}{d\theta} = \begin{cases} \frac{dV_n}{d\theta} & \text{if } V_n \geq s, \\ 1 & \text{if } V_n < s. \end{cases}$$

for either $\theta = s$ or $\theta = q$. Taking $V_0 = S = s + q$, the expression reduces to 1 for all n, which is in accord with the sample path analysis.

On the other hand, a change in q with s held fixed may cause a change in the set of ordering decisions, resulting in radical changes in the sample path and hence in the sample performance function $\overline{V}_N$. Thus, SPA is required to derive an unbiased derivative estimator with respect to $\theta = q$. An SPA estimator for $\partial E[\overline{V}_N]/\partial s$ that can be easily and efficiently estimated from the original sample path is given by

$$1 + \frac{1}{N} \sum_{n \leq N: V_n < s} \frac{f(V_n + D_n - s)}{1 - F(V_n + D_n - s)} \left[s - E[D] - \overline{V}_N \right].$$

Real-World Application Example

In the October 30, 2000 issue of *Fortune* magazine, an article entitled, "New Victories in the Supply-Chain Revolution" (Siekman 2000) describes "a classic distribution challenge: how to avoid lost sales without incurring the cost of carrying extra inventory" when Caterpillar, the "world's largest builder of construction equipment . . . posed daunting supply chain questions" regarding the distribution of a new line of compact construction machines, specifically related to determining appropriate inventory levels for the U.S. market. "Among the techniques . . . used to attack this complex (supply chain inventory control) problem was . . . infinitesimal perturbation analysis, for which no complete explanation is possible for the faint-hearted or mathematically disadvantaged."

Historical Notes

PA was developed by Ho et al. (1979) when the first author was consulting on a real-world buffer design problem for a Fiat Motor Company serial production line. The single-server queue example was first considered in Suri and Zazanis (1988), and the inventory example in Fu (1994). The other area in which PA has been most widely used after queueing and inventory is financial engineering, where IPA is called the pathwise method in Glasserman (2004); see also Fu and Hu (1995). Other applications include PERT networks, dams, insurance, preventive maintenance, statistical process control, and traffic light signal control; see Ho and Cao (1991), Fu and Hu (1997), and Fu (2006) for examples and references.

See

- Inverse Transform Method
- Score Functions
- Sensitivity Analysis
- Simulation of Stochastic Discrete-Event Systems
- Simulation Optimization
- Variance Reduction Techniques in Monte Carlo Methods

References

Brémaud, P., & Vázquez-Abad, F. J. (1992). On the pathwise computation of derivatives with respect to the rate of a point process: The phantom RPA method. *Queueing Systems: Theory and Applications, 10*, 249–270.

Cao, X. R. (1994). *Realization probabilities: The dynamics of queueing systems*. Boston: Springer.

Cassandras, C. G., & Larfortune, S. (2008). *Introduction to discrete event systems*. New York: Springer.
Cassandras, C. G., & Strickland, S. G. (1989). On-line sensitivity analysis of Markov chains. *IEEE Transactions on Automatic Control, 34*, 76–86.
Dai, L. Y., & Ho, Y. C. (1995). Structural infinitesimal perturbation analysis for derivative estimation in discrete event dynamic systems. *IEEE Transaction on Automatic Control, 40*, 1154–1166.
Fu, M. C. (1994). Sample path derivatives for (*s, S*) inventory systems. *Operations Research, 42*(2), 351–364.
Fu, M. C. (2006). Gradient estimation. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science: Simulation, chapter 19* (pp. 575–616). Amsterdam: Elsevier.
Fu, M. C. (2008). What you should know about simulation and derivatives. *Naval Research Logistics, 55*(8), 723–736.
Fu, M. C., Hong, L. J., & Hu, J. Q. (2009). Conditional Monte Carlo estimation of quantile sensitivities. *Management Science, 55*(12), 2019–2027.
Fu, M. C., & Hu, J. Q. (1995). Sensitivity analysis for Monte Carlo simulation of option pricing. *Probability in the Engineering and Informational Sciences, 9*(3), 417–446.
Fu, M. C., & Hu, J. Q. (1997). *Conditional Monte Carlo: Gradient estimation and optimization applications*. Boston: Kluwer Academic.
Gaivoronski, A., Shi, L. Y., & Sreenivas, R. S. (1992). Augmented infinitesimal perturbation analysis: An alternate explanation. *Discrete Event Dynamic Systems: Theory and Applications, 2*, 121–138.
Glasserman, P. (1991). *Gradient estimation via perturbation analysis*. Boston: Kluwer Academic.
Glasserman, P. (2004). *Monte Carlo methods in financial engineering*. New York: Springer.
Gong, W. B., & Ho, Y. C. (1987). Smoothed perturbation analysis of discrete-event dynamic systems. *IEEE Transactions on Automatic Control, AC-32*, 858–867.
Ho, Y. C., & Cao, X. R. (1991). *Perturbation analysis and discrete event dynamic systems*. Boston: Kluwer Academic.
Ho, Y. C., Cao, X. R., & Cassandras, C. G. (1983). Infinitesimal and finite perturbation analysis for queueing networks. *Automatica, 19*, 439–445.
Ho, Y. C., Eyler, M. A., & Chien, T. T. (1979). A gradient technique for general buffer storage design in a serial production line. *International Journal of Production Research, 17*, 557–580.
Ho, Y. C., & Li, S. (1988). Extensions of infinitesimal perturbation analysis. *IEEE Transactions on Automatic Control, AC-33*, 827–838.
Hong, L. J. (2009). Estimating quantile sensitivities. *Operations Research, 57*(1), 118–130.
Shi, L. Y. (1996). Discontinuous perturbation analysis of discrete event dynamic systems. *IEEE Transactions on Automatic Control, 41*, 1676–1681.
Siekman, P. (2000). New victories in the supply-chain revolution. *Fortune*, (October 30).
Suri, R., & Zazanis, M. A. (1988). Perturbation analysis gives strongly consistent sensitivity estimates for the M/G/1 queue. *Management Science, 34*, 39–64.
Vakili, P. (1991). Using a standard clock technique for efficient simulation. *Operations Research Letters, 10*(8), 445–452.

Perturbation Methods

Procedures that modify the constraints of a linear-programming problem so that all basic feasible solutions will be nondegenerate, thus removing the possibility of cycling in the simplex method. The modification can be either explicitly done by adding small quantities to the right-hand sides or implicitly by using lexicographic procedures.

See

- ▶ Cycling
- ▶ Degeneracy
- ▶ Lexicographic Ordering

Petroleum Refining

David S. Hirshfeld
MathPro Inc., Bethesda, MD, USA

Introduction

By many financial and physical measures, the petroleum industry is the world's largest industry. The industry's operations comprise a global supply chain that produces, transports, refines, and distributes more than 85 million barrels of oil per day – nearly 5 billion tons per year.

Because of its scale, global scope, and huge capital requirements, the petroleum industry is populated with many large, vertically-integrated companies (many of them national oil companies) with global operations. The industry is highly competitive because it has many participants and because it produces basic commodities (e.g., gasoline, diesel fuel, petrochemical feedstocks, etc.) that are difficult to differentiate by brand. The industry's huge volume and low margins mean that even small changes in operating costs have important effects on operating results. The petroleum industry is a leader in the development and application of new technology; it develops and applies advanced technologies in every phase of operations. Consequently, the industry

employs large numbers of scientists, engineers, and applied mathematicians, many with advanced degrees.

For these and other reasons, the petroleum industry has been a pioneer in the application of OR/MS across all of its primary operations and has successfully applied virtually every OR/MS tool in these operations. During the 1960s and 1970s, most large integrated oil companies had strong OR/MS groups or departments with concentrations of expertise in linear programming, simulation, and statistical analysis (Baker, 2000). These groups consistently stretched the limits of OR/MS tools and methods, and they provided the impetus and the financial support for many advances in OR/MS software tools and analytical methods. Most of these groups no longer exist. But even so, OR/MS applications in the petroleum industry are ubiquitous and fully embedded in the various business functions that use them. Nowhere is this more evident than in the petroleum refining sector.

OR/MS and Petroleum Refining

Petroleum refining is a unique and critical link in the petroleum supply chain. The other links add value mainly by performing spatial transformations on petroleum (e.g., lifting crude oil to the surface; moving crude oil from oil fields to storage facilities and then to refineries; moving refined products from refinery to terminals and end-use locations, etc.). Refining adds value by performing chemical transformations and blending operations on petroleum – converting crude oil (which in itself has little end-use value) into a broad spectrum of valuable refined products. The primary economic objective in refining is to maximize that added value.

Petroleum refineries are large, continuous-flow process plants with extremely complex processing schemes for processing multiple crude oils and other input streams into a large number of refined (co-) products, most notably LPG, gasoline, jet fuel, diesel fuel, petrochemical feedstocks, home heating oil, fuel oil, and asphalt. Each refinery has a unique configuration and operating characteristics, determined primarily by its location, vintage, preferred crude oil slate, and market requirements for refined products. More than 660 refineries, in 116 countries, are currently in operation; virtually every one has OR/MS tools, including optimization models, embedded in its operations.

Since the earliest days of OR/MS and continuing to the present, refining has been a particularly rewarding domain for applying OR/MS methods in general, and linear programming (LP) and its extensions in particular (mixed integer programming (MIP), special ordered sets (SOS1 and SOS2), and successive linear programming (SLP), etc.).

OR/MS Applications in Petroleum Refining

Baker (2000) reports, "The refining industry began using linear programming (LP) shortly after its invention (Bodington and Baker 1990). In the early 1950s, many major oil companies began using LP-based product blending models (Charnes et al. 1952) which severely tested the available computational capabilities of that time. As computer capabilities expanded, so did the scope of LP models, encompassing whole refineries (Symonds 1955) and the US refining industry (Manne 1958)."

"The nonlinear nature of petroleum and chemical processes was first incorporated by Shell Oil via successive linear programming (SLP), a straightforward technique based on the iterative solution of linearized models (Griffith and Stewart 1961). SLP... was applied by most major companies in the 1960s (Baker and Lasdon 1985). Distributed recursion (DR), a specific form of SLP dealing with the distribution of nonlinear error terms across [multiple] blended pools, is widely used in contemporary models of petroleum refining."

"Literally... every other form of nonlinear optimization has been applied in the [refining] industry. Lasdon and Waren (1980) provided a comprehensive survey of applications. Production planning and scheduling has seen a wide variety of hybrid approaches combining mathematical programming, expert systems, decision support systems, forecasting techniques and simulation. Klingman et al. (1987) describes the integrated logistics system developed at Citgo. A combination of network flow algorithms, mixed-integer programming, and decision support were applied to ship scheduling at Ethyl Corporation (Miller, 1987). Brown et al. (1987) reports on a vehicle loading and

routing system developed for Mobil Oil. The design and development of integrated systems for planning and scheduling is an area of active interest both in academic and industrial settings (Baker 1994)."

Today, mathematical programming and other OR/MS techniques are embedded in numerous refining sector functions, including (in roughly decreasing order of time horizon):

- Capital investment planning
  - Economic evaluation of alternative designs for new refineries
  - Evaluation of alternative configurations for refinery upgrading projects
- Process design
- Tactical planning
  - Evaluation of inter-company product exchanges and processing agreements
  - Optimization of multi-period operations of multi-refinery, multi-terminal logistics systems
  - Evaluation of new processes and technologies
  - Regulatory compliance
- Operations planning
  - Crude oil valuation and supply planning
  - Crude oil cargo selection (Pawde and Singh 2010)
  - Development of quarterly and monthly refinery operating plans
  - Integration of refinery operations and refined product distribution (Guyonnet et al. 2009)
  - Concurrent multi-product blending
- Operations scheduling
  - Process sequencing
  - Inventory (tankage) management
  - Batch blending of refined products
- Process control

The planning and design applications have time horizons measured in years or months, are forecast-driven, and can return solutions in which multiple operations or operating modes employing the same resources or facilities are executed in the given time period. Scheduling applications, on the other hand, have much shorter time horizons (weeks or days), are order- or sequence-driven, and recognize operating policies or physical constraints on the utilization of specific facilities – e.g., only one activity or operation at a time can be performed in a particular facility. Plans returned by planning models may not be physically implementable without being subjected to a detailed scheduling analysis.

Refining organizations use their refining optimization models across many planning horizons:

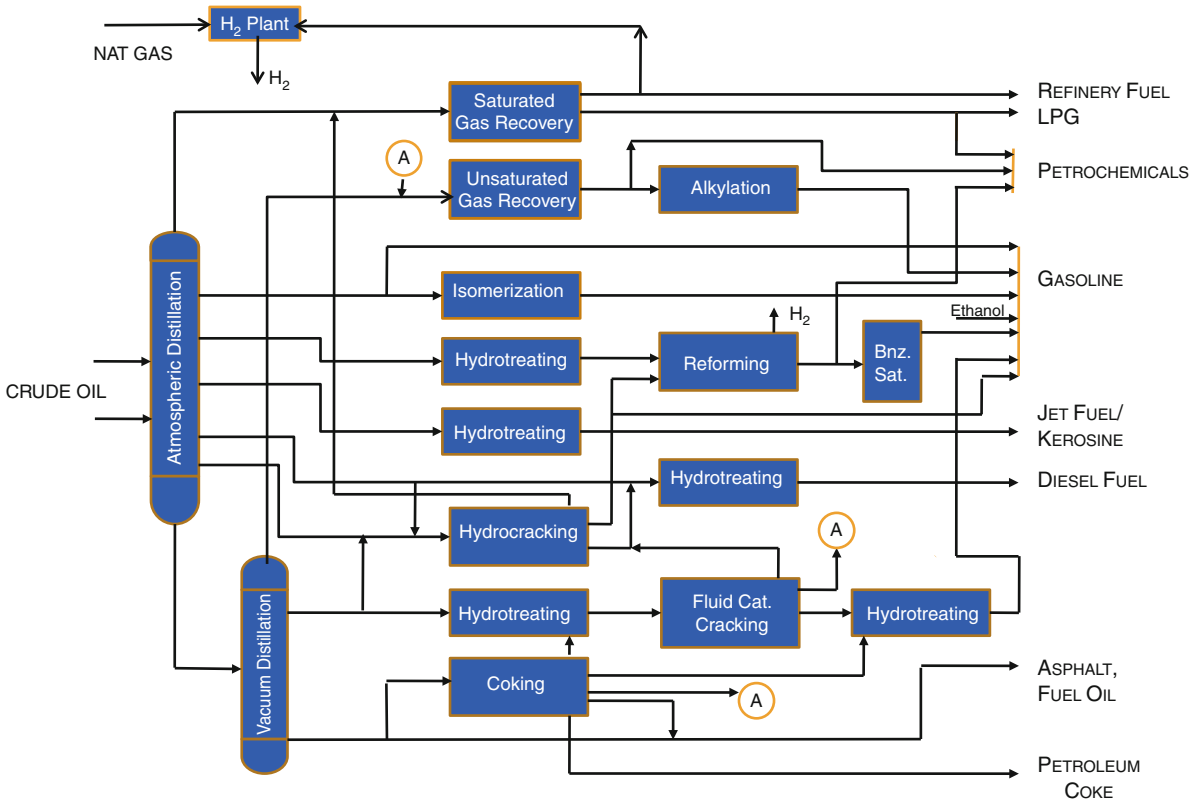
- **Long-term** (3+ years): capital investment planning, regulatory compliance, restructuring
- **Annual**: annual budgeting, evaluation of term contracts for crude supply and product sales, maintenance and turn-around planning
- **Quarterly/monthly**: operations planning to meet product demands and seasonal transitions in product specifications, evaluation of spot transactions for crude purchases and product sales, estimation of dispatches to product pipelines and tankers
- **Weekly**: scheduling operations and batch blending to make optimal use of crudes on hand and available processes

Refinery planning applications are practiced not only by refinery organizations but also by other organizations having interest in the refining sector, such as engineering firms, independent technology providers (e.g., process licensors), catalyst and chemical manufacturers, and consulting firms. Government agencies also apply LP to analyze refining operations, for various purposes – for example, the U.S. Environmental Protection Agency in estimating the costs of new regulatory standards for transportation fuels, and the U.S. Energy Information Administration in producing its annual projections of U.S. energy supply and demand).

Refining Operations and the Driving Forces for Refinery Modeling

Understanding the rationale for and benefits of OR/MS methods in refining industry requires some understanding of refining itself (The National Petroleum Council (2000) Web site includes an excellent tutorial on the fundamentals of refinery operations).

Figure 1 is a highly simplified flow chart of a notional complex refinery, illustrating a typical pattern of oil flow through the refinery – from the crude oil distillation unit that separates crude oil into various boiling range fractions, or cuts, through the various downstream processing units that chemically transform these fractions into blendstocks (the refinery streams that are the constituents of blended products) and ultimately to product blending. For purposes of



Petroleum Refining, Fig. 1 Simplified Flow Chart of a Notional Refinery

this discussion, the importance of Fig. 1 is not in its details, but in the overall picture it conveys of the complexity of refining operations in general.

Several broad aspects of refining operations suggested by Fig. 1 merit comment in the context of refinery modeling applications.

- Refinery operations are extremely complex.

Figure 1 only hints at the actual complexity of refinery operations – with respect to the physical facilities of the refinery, the interaction of these facilities with one another, and the range of operations of which they are capable. The complexity is such that refinery operations can be fully understood only with formal, refinery-wide models and can be optimized, in an economic sense, only through the use of mathematical programming.

Refiners can change the operations of their refineries to respond to the continual changes in crude oil and product markets, but only within physical limits defined by the performance characteristics of their refineries and the properties of the crude oils they process. Mathematical programming models of refinery operations that express these physical constraints are the only reliable means of generating achievable (i.e., feasible) and economic (i.e., optimal) responses to changes in market environment.

- Refineries produce a wide range (or slate) of products – actually co-products.

Refineries produce a range of co-products not only because of market demand for the various products but also because of the constraints imposed by the refining facilities themselves. Refiners need to know the marginal cost of production for each refined product, because these marginal costs are the primary determinants of the products' spot prices – the prices at which products change hands at the refinery gate. Mathematical programming models of refinery operations routinely produce rigorous estimates of marginal production costs that are well grounded in theory, for every co-product produced (The solution values for certain of the dual variables in a refinery

Petroleum Refining, Table 1 Classification of Refining Processes

Primary Classes of Refining Processes in Complex Refineries		
Class	**Function**	**Examples**
Crude distillation	Separate crude oil charge into boiling range fractions for further processing	Atmospheric distillation Vacuum distillation
Conversion	Break down ("crack") heavy crude fractions into lighter, higher-valued streams for further processing	Fluid cat cracking Coking, Hydrocracking
Upgrading	Enhance the blending properties (e.g., octane) and value of gasoline and diesel blendstocks	Reforming Alkylation, Isomerization
Treating	Remove hetero-atom impurities from refinery streams and blendstocks	Hydrotreating Caustic treating
Separation	Separate, by physical or chemical means, constituents of refinery streams for further processing	Fractionation Extraction
Blending	Combine blendstocks to produce finished products that meet product specifications and environmental standards	
Utilities	Supply refinery fuel, power, steam, oil movements, storage, emissions control, etc.	Power generation Sulfur recovery

model are precisely the marginal values in question). Indeed, mathematical programming is essentially the only practical and useful tool for computing the marginal costs of refined products.

All of this was readily apparent to the engineers and applied mathematicians working in the refining sector in the 1950s and provided the impetus for the early adoption of linear and mathematical programming throughout the refining industry.

Refinery Processes and Operations

Complex, world-class refineries (including virtually all U.S. refineries) comprise as many as fifty or more distinct refining processes, which carry out multiple physical and chemical transformations to convert crude oil into a broad slate of refined products. Despite their number and diversity, refining processes can be thought of in terms of a few broad classes based on their functions, as shown in Table 1.

Crude Oil and the Crude Oil Distillation Process

Crude oil distillation, the process at the front end of every refinery, regardless of size or overall configuration, has a unique function that affects all of the processes downstream of it. In a refinery model, the representation of crude oil properties and of the crude distillation process in a refinery model influences all of the other process representations in the model.

Crude oil comprises tens of thousands of chemical compounds (primarily hydrocarbons). These compounds range from the very light – low molecular weight, simple structure, low density, low boiling point ($<60^{o}$ F) – to the very heavy – high molecular weight, complex structure, high density, high boiling point ($>1000^{o}$ F).

Each of the more than 1,500 crude oils in commerce has its own unique signature, with respect to composition, proportions of light and heavy components, and physical properties. The unique composition and properties of a crude oil largely determine its value as a refinery input and the range of refined products that a given refinery can produce from it.

The crude distillation unit in a refinery accepts a combination of different crude oils and separates it into a number of streams (known as crude fractions or cuts). Each fraction leaving the crude distillation unit (1) is defined by a unique boiling point range (e.g., 180^{o}–250^{o} F, 250^{o}–350^{o} F, etc.), (2) contains material from each crude oil fed to the crude distillation unit, and (3) is made up of hundreds of distinct hydrocarbon compounds, all of which have boiling points within the cut range. An essential simplifying assumption in the analysis of refining operations is that the crude distillation unit makes "sharp" cuts – that is, any

given hydrocarbon species in the crude oil mixture is present in one and only one cut (i.e., there is no "overlap" between the crude fractions leaving the crude distillation unit).

Each crude fraction leaving the crude distillation goes to a different refinery process for further processing (Fig. 1). The highest boiling fractions of the crude, collectively known as the heavy ends, have relatively little economic value – indeed lower value than the crude oil from which they come. Refineries must convert, or upgrade, these heavy ends into more valuable light products (gasoline, jet fuel, diesel fuel, etc.).

Stream Properties and Refining Processes

In a refinery model, the specification of the temperature ranges of the cuts and the representation of the various properties of the crude fractions exerts a strong influence on the representations of all of refining processes downstream of the crude distillation and on the results returned by the model.

In general, each refining process handles multiple feed streams and produces multiple outputs (co-products). The yields of the co-products, their physical and chemical properties, and the direct operating costs of each process depend on the properties of the input streams (which in turn depend on the mixture of crude oils processed and the temperature ranges of the crude cuts). Consequently, analyzing refinery operations requires keeping track of not only the various streams flowing through the refinery but also numerous properties associated with each stream.

Tracking stream properties is essential in analyzing the blending operations at the back end of every refinery. Refineries produce a diverse set of co-products (e.g., gasolines, jet fuel, diesel fuels, petrochemical feedstocks, etc.); large, complex refineries may produce as many as forty distinct products. Most of these products are blends of various streams produced in crude distillation or in the downstream processes (usually five to ten refinery streams per product). Each product is blended to meet a vector of specifications on the products' properties (e.g., density, sulfur content) and performance characteristics (e.g., octane, emissions from vehicle tailpipes, etc.). These specifications represent industry standards and government regulations.

The Content of Refinery LP/MP Models

Structure

An LP or MP model of a single refinery in a single time period is essentially an assembly of

- Equations and inequalities representing
  - Volume balances on refinery inputs, refinery-produced streams, and refinery outputs (volume supplied + volume produced = volume consumed + volume blended or sold)
  - Mass balances and energy balances (conservation of mass and energy)
  - Blending property balances linking individual refinery streams and their blending properties to specification-blended product pools
  - Accounting identities to capture refinery-wide operating costs, consumption of energy and utilities, and generation of effluents (including CO_2)
  - Upper limits on the through-put capacity of the various refining processes
  - Special constraints reflecting internal technical restrictions or limitations
  - Special constraints reflecting external requirements
  - Regulatory standards (such as the federal and California standards for reformulated gasoline).
- Variables representing
  - Volumes of refinery inputs, such as crude oil purchases
  - Volumes of refinery streams flowing into or out of each process unit (such as those shown in Fig. 1) at specified operating conditions
  - Volumes of produced refinery streams going to each blended product pools
  - Volumes of finished products leaving the refinery
  - Amounts of new refinery process capacity (if any) added through capital investment

Multi-time-period models contain, in addition to the above elements, equations and variables representing inventory transfers from one time to the next of crude oils, other refinery inputs, certain intermediate refinery streams, and finished products.

Multi-refinery models contain, in addition to the above elements, equations and variables representing the transport of refined products from the refineries to individual destinations (product terminals, end-use sites, etc.) or destination regions, through various capacitated transportation modes.

In all of these variants, the objective function usually represents gross profit or, as it sometimes called, profit contribution:

Refinery netback minus (the sum of direct operating costs + capital recovery charges)

where

- **Refinery netback** is the net revenues (price*quantity) received by the refinery from the sale of all refined products
- **Direct operating costs** include the total purchase costs (price*quantity) of crude oil and other refinery inputs, purchased utilities, and catalyst and chemicals consumption; inventory carrying costs (in multi-period models); transportation costs for product movements to demand sites (in multi-refinery models), and regulatory compliance costs
- **Capital recovery charges** denote return on un-depreciated refinery investment, per unit of throughput.

In multi-period models, the profit contribution terms for future time periods can be discounted by multiplying them by a discount rate factor: $(1+\text{discount rate})^{-t}$, where t is the time-period index.

Models of refinery operations contain distinct representations of each of the refining processes that have a significant effect on the refinery's economics. A complex refinery can comprise forty or more such processes. Each process (or process/refinery combination, in a multi-refinery model) is represented in a discrete sub-matrix of the overall model. Each process sub-matrix consists of one or more operating mode or input/output variables, any number of which can be active in a given solution. Each operating mode variable intersects certain equations representing volume balances on the streams flowing into and out of the process, energy balances, and accounting relationships. The vector of input/output coefficients associated with each operating mode variable denote the quantities of individual inputs (refinery streams, utilities, capacity, costs) and outputs (different refinery streams) per unit of process throughput in a particular operating mode, as well as the relevant properties of the output streams.

Depending on the number of processes and refinery streams represented, a typical single-refinery, single-time-period LP model contains about 1,500–5,000 constraints, and 5,000–15,000 variables. Refinery models have highly structured matrices, composed of the various process and blending sub-matrices, linked by the volume balance and property balance constraints. The matrices are relatively dense, but have low super-sparsity (because the input/output coefficients in the process representations tend to be unique).

Coefficients

The coefficients for the crude oil distillation sub-matrix usually are drawn from *crude oil assays*. A crude oil assay is an assembly of data on the composition and property of a whole crude oil and of 15–20 boiling range fractions of that of that crude, developed through laboratory testing.

Crude assays exist for all crude oils in commerce; many, but not all, of these assays are in the public domain.

Commercial software products called crude oil assay managers with associated assay libraries are widely used to generate the coefficients for representing the crude oil distillation process in a refinery model, with user-specified boiling ranges for the crude fractions.

The coefficients for the sub-matrices representing the refining processes are refinery-specific in most models and are derived, directly or indirectly, from experimental data. Depending on the process, the data may come from laboratory testing, pilot plant operations, refinery-level plant testing, refinery accounting systems, and process simulators (detailed engineering models of individual refining processes). In general, all of these sources of refinery data are proprietary.

Some non-proprietary, generalized correlations and data for characterizing refining processes are available in the open literature, primarily in a few textbooks (e.g., Maples (2000), Gary et al. (2000)) and articles in refining industry trade journals.

Populating a refinery optimization model with realistic input/output coefficients is a highly specialized undertaking, requiring considerable knowledge of refinery operations and refining technology – subjects that are at some remove from operations research.

Nonlinearities in Refinery Models

To this point, this overview of refinery optimization models seems to imply that refining operations are

linear in nature and therefore can be suitably represented as linear programming models. Refining operations are subject to mass balance, energy balance, and volume balance constraints, all of which are linear, as are the constraints that govern multi-ingredient blending to meet product specifications (as long as the blending is simply physical mixing with no chemical interactions between ingredients). Consequently, refinery optimization was a natural pioneering application for linear programming. And even today, LP remains the optimization method of choice for many refinery modeling applications.

However, refinery operations actually embody many nonlinear phenomena, some of which can have a strong influence on refining operations and economics. Almost from the beginning, a steadily increasing number of refining organizations have sought to enhance their capabilities to capture these nonlinearities in their refinery optimization models and thereby more accurately represent the true capabilities and limitations of their refining facilities.

Some of the nonlinearities of interest are economic in nature and bear on the objective function; others involve underlying physical processes and relationships and bear on the constraint set. Many of these nonlinearities, including the five discussed below, are incorporated readily in refinery models, facilitated in many instances by the capabilities of commercial solvers.

Investments in New Refining Capacity

Existing refineries often invest in additional processing capacity – either new process units or expansion of existing ones – in order to increase total production capacity, produce new products, upgrade the value of existing products, or comply with new regulatory standards bearing on product quality or performance characteristics.

Often, the capacity added for a given process is represented by a continuous variable (whose value is expressed in a capacity measure, such as K barrels/day), and the corresponding investment is approximated by multiplying this variable by a constant investment rate coefficient (whose value is in $/(barrel/day)).

$$\mathbf{I} = \mathbf{a}^{*}\mathbf{Q} \tag{1}$$

where **I** is the investment (in K$), **Q** is the capacity added (in $/barrel/day), and **a** is the investment rate factor ($/(barrel/day)). The value of the investment rate factor depends on the refining process and the refinery's location.

However, the capital investment required to add new refining capacity enjoys economies of scale; that is, the investment per unit of added capacity is not a constant, but decreases with increasing total amount of added capacity. The standard relationship between the amount of new capacity added and the required capital investment is

$$\mathbf{I} = \mathbf{b}^{*}\mathbf{Q}^{\beta} \tag{2}$$

where **I** is the investment (in K $), **Q** is the capacity added (in K barrels/day), **b** is a constant whose value depends on the refinery's location, and β is an exponent whose value depends on the refining process in question. Most refining processes have a β value in the range of 0.6–0.7.

Equation (2) is a non-convex function. It can be represented in a refinery MP model in one of several ways.

One approach is to (1) assign a set of binary (0–1) variables to each of three or four standard levels of new capacity addition (e.g., 10 K barrels/day, 20 K barrels/day, etc.) for each refining process that is a candidate for investment and (2) for each such set, add a constraint specifying that at most one of the variables in the set can take on the value 1 in an optimal solution (or, equivalently, define the set of binary variables for each refining process as a Special Ordered Set Type 1 (SOS1)). Each of the binary variables carries a coefficient denoting the capital investment for the capacity addition it represents, obtained from the (2) for each process.

Another approach is to represent (2) for each process that is a candidate for investment as a piecewise linear function by means of a Special Ordered Set Type 2 (SOS2) for each such process.

Semi-Continuous Quantities

In many situations, restrictions exist on the minimum and maximum volume of a particular flow or the minimum and maximum extents to which a particular operation can be performed. For example, pipeline off-takes from a refinery are subject to the pipeline's regulations on the minimum and maximum size shipments that it will accept. Similarly, purchases of tanker-borne crude oil are

subject to volume to volume limits determined by the size of the tanker and its cargo compartments.

These and similar constraints can be represented in refinery models by means of semi-continuous variables: variables that can be either zero or continuous within a range defined by a strictly non-zero lower bound and (optionally) an upper bound. Semi-continuous variable capability is available in most commercial having mixed-integer-programming (MIP) capability.

Quality Blending

In the canonical product blending problem, the ingredients blend linearly with respect to the blend properties that are subject to limits (specifications). That is, the properties of the blended product requirements are the weighted averages of the corresponding properties of the various ingredients. This is linear blending.

Many refinery models represent the blending of refined products to specifications just that way. However, there is more to the specification blending of refined products than simple linear blending. Some of the specifications to which refined products are blended pertain to purely physical properties (e.g., sulfur content, density); other to chemical properties (e.g., octane, volatility, etc.). Blending to specifications on physical properties is indeed linear, as defined above. However, blending of chemical properties often is not linear, because of the interactions among different chemical interactions that occur when individual ingredients (*blendstocks* in refining parlance) are blended together. For example, consider two gasoline blendstocks, one having 90 octane, the other 70 octane. A 50/50 blend of the two might yield a blend octane of, say, 82 or 77 (not 80), depending on the chemical interactions involved. Moreover, the blend octane may vary with the relative amounts of the two blendstocks. This is nonlinear blending.

Several techniques are available for representing nonlinear blending. The most widely used one involves the use of blending indices in place of blendstock properties. A blending index for a given nonlinear property is an empirically determined function of that property such that the function blends linearly, even though the property itself does not. For example, consider the property Reid Vapor Pressure (RVP), a standard measure of gasoline volatility. RVP blends nonlinearly, but the RVP Index, defined here, blends linearly.

$$\text{RVP Index} = \text{RVP}^{\rho} \tag{3}$$

where the value of the exponent ρ is about 1.17 (Different refiners may use slightly different values for ρ).

Some blending indices involve more complicated functions of the underlying property. For example, Pour Point (PP), a measure of diesel fuel's ability to flow at low temperature, has a Pour Point Index given by:

$$\text{PP Index} = \text{EXP}[1.85 + 0.042^{*}(\text{PP})] \tag{4}$$

Many gasoline and diesel fuel blending properties are represented by such blending indices in refinery models.

Gasoline octane blending is a special instance of nonlinear blending for two reasons. First, octane has a relatively high marginal refining cost; refiners do not wish to "give away" octane in the course of meeting the octane standards. Second, the blending octane of a gasoline blendstock (i.e., the apparent octane contribution of the blendstock to the finished blend) is a function not only of the blendstock's native octane but also the composition of the finished blend. The refining industry has developed special methods, based on laboratory data, to estimate blend octanes over a range of compositions. These methods, outlined by Maples (2000), are beyond the scope of this article.

Pooling

Pooling is the mixing or commingling of multiple streams (crude fractions or refinery streams) into a new stream (the pool), whose properties (e.g., density, sulfur content, etc.) are the volume-weighted averages of the properties of the individual streams entering the pool:

$$\mathbf{Q}_j V = \Sigma_i q_{ij} V_i \quad \Rightarrow \quad \mathbf{Q}_j = \Sigma_i q_{ij} V_i \; / \; \Sigma_i V_i \tag{5}$$

where $\mathbf{V}$ is the volume of the pool stream, $\mathbf{Q}_j$ is the j^{th} property (e.g., density) of the pooled stream, V_i is the volume of the i^{th} stream making up the pool, and q_{ij} is the j^{th} property of that stream.

The q_{ij} are constant coefficients, but **V** and the V_i are variables, whose values are known only when the model returns a solution. Thus, the properties ($\mathbf{Q_j}$) of the pooled stream are nonlinear function of model variables and can be determined only after a solution is in hand.

Consequently, it is not possible to define exact representations of those effects on downstream refining operations that depend on the properties of the pooled stream. These effects reside in the refining process and specification blending sub-matrices. Thus, not only do the optimal volumes of the pooled stream, **V**, and the streams making up the pool, V_i, depend on the properties of the pool stream, but also the economic value of the pool stream **V**.

The original, or traditional, approach to formulating refinery models does not address pooling at all – not because the problem was not recognized but because the analytical tools needed to address it were not then at hand. In the traditional approach (still widely used), crude distillation and each of the downstream refining processes are represented in discrete sub-matrices. In the crude distillation sub-matrix, each crude oil is represented by its own input/output vector, in which the output coefficients are the volumetric yields of the various cuts. This representation implies that (1) the various crude oils, each with their own properties and yield patterns, are segregated from one another as they go through the crude distillation unit and (2) the boiling range cuts from the various crude oils are likewise segregated from one another as they move to the downstream processes. In the downstream process sub-matrices, each feed is attributable to a particular crude oil and each is represented by its own input/out vector. This scheme represents each process operating as if it were processing a group of segregated feed streams, each with its own operating mode, rather than one pool stream.

Refinery models formulated in this way tend to contain many more stream flow variables, and many more blendstock variables and blending options, than there are in the "real" refinery. This can lead, in certain situations, to over-optimization – the model's returning solutions indicating better refining economics than the real refinery can achieve.

Explicit representation of the stream pooling that occurs in real refineries calls for special model formulation and solution techniques. The most widely used modeling technique is called Distributive Recursion (DR), a variant of SLP developed expressly to deal with the pooling problem in models of refining and other process flow industries. First developed in the late 1970s, DR has come into increasingly wide use as the required software tools have become more widely available.

In DR, the model user provides initial estimates of the Q_i for all of the pool streams. The procedure uses these estimates to conduct an initial solution pass, which returns (1) the downstream dispositions and marginal value of each pool and (2) the volumes, V_i, of each stream entering each pool. Using the new set of V_i values, the DR procedure re-estimates the various pool qualities. The difference between the n^{th} and $n + 1^{st}$ estimates for a given pool is called its quality error. DR distributes each quality error across the various downstream dispositions of each pool and initiates a new solution pass incorporating the new estimates of pool qualities and quality errors. DR conducts a series of such solution passes that seek to converge to an optimal solution in which the quality errors are driven to zero (to within a user-specified tolerance).

Performance of Refining Processes

In the original, or traditional, approach to formulating refinery models, each downstream process is represented in a discrete sub-matrix. Each process sub-matrix comprises a set of variables (vectors), each denoting a unique combination of (1) a segregated (not pooled) feed stream to the process and (2) a particular operating mode for the process (defined by physical operating conditions, such as temperature). Each such variable has a unique set of input/out coefficients, defining the operation of the process. This representation implies that (1) processes behave linearly, independent of the composition and properties of their feeds and (2) each (notionally) segregated stream can be processed at its own set of operating conditions as it flows through the process. In reality, process performance depends on the properties of the pooled feed to the process.

With the advent of DR, some refining companies sought a more rigorous representation of refining processes that used pooled input streams and captured the effects of input stream properties on the

yields and properties of the output streams. This effort led to the *base-delta* (B-D) approach to representing refining processes in optimization models (Bodington 1995).

In the B-D approach, each downstream process is represented in a discrete sub-matrix, comprising:

- One or more base vectors, each denoting operation of the process with a typical, or base, feed and a standard, or base, operating mode.
  The input coefficients on the base vector(s) denote those properties of the pooled feed that affect the yields and properties of the process outputs. The output coefficients denote the yields and properties of the various outputs, when the process is operating at base conditions.
- A set of delta vectors for each base vector. The solution values taken on by the various delta vectors are determined as part of the overall model solution obtained via DR.
  The coefficients on each delta vector denote the effects of a small change in one pooled feed property (relative to the base property) on the yields and properties of the various outputs. Each delta vector coefficient is, in effect, the partial first derivative of a particular process output property with respect to an input property. The set of all delta vector coefficients for a process is equivalent to a Jacobian matrix for the process.

Both the base yield coefficients and the delta vector coefficients usually are generated by means of detailed engineering models (called process simulators) of the various processes, or (less likely) by generalized correlations or plant testing. Modern refinery modeling systems that offer DR now provide interfaces to process simulators. These interfaces link the process simulators directly to a refinery optimization model and allow them to be invoked at each DR solution pass to dynamically update the some or all of the delta coefficients in response to the current DR solution. Use of this facility increases the likelihood of reaching a local optimum.

Finally, the traditional representation of crude distillation, the refinery's front-end process, treats the cut point temperatures of the various crude fractions (e.g., 160^{o}–250^{o} F for a light naphtha stream) as constants. The advent of DR allows the cut points themselves to be recursed variables, an option that is now widely used.

Comments on Distributed Recursion (DR)

As with any non-linear technique, DR can – and often does – return solutions that are only locally optimal. In particular, the DR procedure requires initial estimates of the properties of each pool and the fractional distributions of each pool to its various downstream dispositions. The specific values of these estimates determine whether the DR procedure converges to a global optimum or to a local optimum. The more pooled streams and the greater the number of pool dispositions in the model, the more likely that the model will return a local optimum.

Capturing the analytical benefits of DR requires considerable software and intellectual resources, including:

- Some means – whether process simulators or sets of correlations – of dynamically representing the effects of process input properties on process output yields and properties;
- An array of special software, including a crude oil assay manager, process simulators (or their functional equivalent), and facilities to execute and control the recursive solution process; and
- Sound model formulation practices, careful estimation of the initial values of stream properties and distributions, and proper settings for the DR procedure's control parameters and tolerances.

Only analysts with access to the necessary system resources and with extensive experience in refinery modeling in general and DR in particular are likely to obtain useful and timely results with DR.

However, many refinery modeling applications do not require the degree of precision that DR is intended to provide in representing the capabilities and limitations of refining facilities. In particular, high accuracy in representing refining facilities may not be warranted in applications, such as tactical and strategic planning, that have planning horizons measured in years, rather than months or weeks. Long planning horizons involve substantial uncertainty regarding crude oil prices, product demands, and other economic factors. These applications place a premium on the ability to rapidly analyze and compare many different model instances, each representing a future economic scenario – as opposed to analyzing a few model instances with greater precision in the representation of refining facilities.

In these situations, the conventional refinery modeling approach, not the DR approach, is usually the method of choice.

Model Management for Refinery Models

Model management is "the care and feeding" of large scale modeling applications. It is a complex of information processing functions that includes model formulation (in an electronic format), data up-dating, case management, matrix generation, optimizer control, solution reporting, model and solution analysis, and model maintenance. All operational use of large scale optimization models involves the performance of these and other functions – whether manually, with an ad hoc collection of software tools, or with a purpose-built software system.

As the size and scope of refinery optimization models increased, the burdens of model management became apparent. The first software tools designed specifically to address elements of model management (as opposed to model solution) were the matrix generation languages fielded in the late 1950s and early 1960s (e.g., Haverly System's MaGen™ and (later) OMNI™; Bonner & Moore's MARVEL™ and (later) GAMMA™). These were procedural programming languages with special functionality and features for generating refinery LP models in optimizer input format and generating output reports on model solutions. The matrix generation languages were a large step forward, but they did not provide a full range of model management functionality.

Somewhat later, a new set of software tools for matrix generation and reporting entered commercial use: the algebraic modeling languages: (e.g., GAMS™, AMPL™, MPL™, MODELER™, and AIMMS™). These are symbolic modeling languages, in which the model formulator expresses the model's constraints and variables symbolically, in an algebra-like syntax. They also provide facilities for model up-dating and report generation.

Starting in the 1950s, many of the major refining companies undertook development of their own comprehensive refinery modeling systems, some using commercial matrix generation languages, others using standard programming languages of the times. Beale (1978) describes British Petroleum's approach to model management. Palmer et al. (1984) describes the conceptual and design foundations for Exxon's PLATOFORM™ model management system. At one time, PLATOFORM routinely handled more than one hundred mathematical programming applications in Exxon. Bodington and Baker (1990) reference other companies' efforts in model management system development.

As a consequence of the waves of consolidation and down-sizing that swept through the petroleum industry starting in the 1980s, most refining companies curtailed or abandoned their efforts to develop and maintain their own model management systems. A few companies still maintain their in-house model management systems. But most refining companies have now supplanted their in-house systems with one of the generalized refinery modeling systems brought into commerce by independent developers (e.g., PIMS™ (AspenTech), GRTMPS™ (Haverly Systems), and RPMS™ (Honeywell Hi-Spec Solutions)).

Commercially available modeling systems must be instantiated with data specific to the refinery of interest: crude oil assays, process capacities and performance characteristics, stream properties, and product specifications. Once instantiated, the generalized refinery modeling systems offer extensive functionality for refinery modeling, including DR (as an option), comprehensive model management functionality, and compatibility with crude oil assay managers, process simulators, spreadsheets, relational databases, and a number of standard commercial solvers.

Concluding Remarks

The petroleum industry pioneered the application of OR/MS across all of its primary operations, and has provided the impetus and the financial support for many advances in OR/MS software tools and analytical methods. This symbiotic relationship is particularly strong in the petroleum refining sector. Since the earliest days of OR/MS, refining has been a particularly rewarding domain for applying OR/MS methods in general, and especially linear programming (LP) and its extensions (in particular, mixed integer programming (MIP), special ordered sets

(SOS1 and SOS2), and successive linear programming (SLP)). As a result, OR/MS applications – especially linear and mathematical programming applications – are ubiquitous and fully embedded in refining operations.

Although petroleum refining is a mature area of application for OR/MS, the tools and methods available to refining industry practitioners continue to improve in terms of speed and functionality. Further advances are likely to come in the realm of model management.

Development and application of optimization models in the refining sector requires deep knowledge of refining technology and economics. Knowledge of optimization algorithms and software tools is necessary but not sufficient for successful application of OR/MS in the refining sector.

See

- ▶ Linear Programming
- ▶ Mathematical Programming
- ▶ Model Management
- ▶ Nonlinear Programming
- ▶ Special-Ordered Sets (SOS)

References

Baker, T. E. (2000). Petrochemical industry. *Encyclopedia of operations research and management science* (2nd ed). Kluwer Academic Publishers.

Baker, T. E. (1994). An integrated approach to planning and scheduling. In D. W. T. Rippin (Ed.), *Foundations of computer-aided process operations* (pp. 237–251). Texas: Austin. CACHE.

Baker, T. E., & Lasdon, L. S. (1985). Successive linear programming at Exxon. *Management Science, 31*, 264–274.

Bammi, D. (1990). Northern border pipeline logistics simulation. *Interfaces, 20*(3), 1–13.

Beale, E. M. L. (1978). Nonlinear programming using a general mathematical programming system. In H. J. Greenberg (Ed.), *Design and implementation of optimization software* (pp. 259–279). The Netherlands: Sijthoff and Noordhoff.

Bodington, C. E. (1995). *Planning, scheduling and control integration in the process industries*. New York: McGraw-Hill.

Bodington, C. E., & Baker, T. E. (1990). A history of mathematical programming in the petroleum industry. *Interfaces, 20*(3), 117–127.

Brown, G. G., et al. (1987). Real-time, wide area dispatch of Mobil tank trucks. *Interfaces, 17*(1), 107–120.

Charnes, A., Cooper, W. W., & Mellon, B. (1952). Blending aviation gasoline–a study in programming interdependent activities in an integrated oil company. *Econometrica, 20*(2), 135–139.

Council, N. P. (2000). *U.S. Petroleum refining: Assuring the adequacy and affordability of cleaner fuels*. Washington, DC: National Petroleum Council.

Edgar, T. F., & Himmelblau, D. M. (1988). *Optimization of chemical processes*. New York: McGraw-Hill.

Findlay, P. L., et al. (1989). Optimization of the daily production rates for an offshore oilfield. *Journal of Operational Research Society, 40*, 1079–1088.

Gary, J. H., Handwerk, G. E., & Kaiser, M. J. (2007). *Petroleum refining technology and economics*. Boca Raton, FL: CRC Press.

Griffith, R. E., & Stewart, R. A. (1961). A nonlinear programming technique for the optimization of continuous processing systems. *Management Science, 7*, 379–392.

Guyonnet, P., Grant, F. H., & Bagajewicz, M. J. (2009). Integrated model for refinery planning, oil procuring, and product distribution. *Industrial and Engineering Chemistry Research, 48*(463–482), 2009.

Hansen, P., et al. (1992). Location and sizing of off-shore platforms for oil exploration. *European Journal of Operational Research, 58*(2), 202–214.

Higgins, J. G. (1993). Planning for risk and uncertainty in oil exploration. *Long Range Planning, 26*(1), 111–122.

Klingman, D., et al. (1987). The successful deployment of management science throughout citgo petroleum corporation. *Interfaces, 17*(1), 4–25.

Lasdon, L. S., & Waren, A. D. (1980). A survey of nonlinear programming applications. *Operations Research, 28*, 102–1073.

Main, R. A. (1993). Large recursion models: Practical aspects of recursion techniques. In T. A. Ciriani & R. C. Leachman (Eds.), *Optimization in industry*. New York: Wiley.

Manne, A. (1958). A linear programming model of the US petroleum refining industry. *Econometrica, 26*(1), 67–106.

Maples, R. E. (2000). *Petroleum refinery process economics* (2nd ed.). Tulsa, Oklahoma: PennWell Corporation.

Miller, D., et al. (1994). A modular system for scheduling chemical plant production. In D. W. T. Rippin (Ed.), *Foundations of computer-aided process operations* (pp. 355–372). Texas: Austin. CACHE.

Miller, D. (1987). An interactive, computer-aided ship scheduling system. *European Journal Operational Research*, *32*(3), 363–379.

Palacios-Gomez, F., Lasdon, L., & Enquist, M. (1982). Nonlinear optimization by successive linear programming. *Management Science, 28*(10), 1106–1120.

Palmer, K. H., et al. (1984). *A model-management framework for mathematical programming*. New York: Wiley.

Pawde, M. D., & Singh, S. (2010). "Crude oil cargo selection and time frame of LP optimization," Petroleum Technology Quarterly, Third Quarter, 2010.

Symonds, G. H. (1955). *Linear programming–the solution of refinery problems*. New York: Esso Standard Oil Company.

Tucker, M. A. (2001). "LP modeling – past, present, and future," National Petrochemical and Refiners Association (NPRA) 2001 Computer conference, Paper CC-01-153.

PFI

▶ Product Form of the Inverse (PFI)

Phase I Procedure

That part of the simplex method directed towards finding a first basic feasible solution.

See

▶ Artificial Variables
▶ Linear Programming
▶ Phase II Procedure
▶ Simplex Method (Algorithm)

Phase II Procedure

The part of the simplex algorithm that finds an optimal basic feasible solution, starting with Phase I basic feasible solution or an initial basic feasible solution.

See

▶ Linear Programming
▶ Phase I Procedure
▶ Simplex Method (Algorithm)

Phase-type Distribution

▶ Phase-type Probability Distributions

Phase-type Probability Distributions

Marcel F. Neuts
The University of Arizona, Tucson, AZ, USA

The probability distributions of phase-type, or *PH*-distributions, form a useful general class for the representation of nonnegative random variables. A comprehensive discussion of their basic properties is given in Neuts (1981). There are parallel definitions and properties of discrete and continuous *PH*-distributions, but the discussion here emphasizes the continuous case.

The simplest example is the Erlang random variable, which can be expressed as the sum of independent exponentially distributed random variables. As a result, one can construct a realization of an Erlang random variable by going through a series of phases, one for each exponential random variable; hence, the Erlang distribution is a phase-type distribution. Generalizing this phase-type idea governs the movement through the phases by a Markov chain that permits movement back and forth between the interior phases, with the final stage being an absorbing barrier.

More specifically, a probability distribution $F(\cdot)$ on $[0, \infty)$ is of phase type if it can arise as the absorption time distribution of an $(m+1)$-state Markov chain with m transient states $1,\ldots, m$ and an absorbing state 0. The generator $\boldsymbol{Q}$ of such a Markov chain is written as

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{T} & \boldsymbol{T}^0 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix},$$

where $\boldsymbol{T}$ is a nonsingular $m \times m$ matrix with negative diagonal elements and nonnegative off-diagonal elements. If $\boldsymbol{e}$ denotes a column vector with all components equal to one, then the vector $\boldsymbol{T}^0$ satisfies $\boldsymbol{T}^0 = -\boldsymbol{T}\boldsymbol{e}$. The initial probability vector of the Markov chain is specified as $(\alpha,\ \alpha_0)$. Without loss of generality, it may be assumed that the generator, $\boldsymbol{Q}^* = \boldsymbol{T} + (1-\alpha_0)^{-1}\boldsymbol{T}^0\alpha$, is irreducible.

The general formula for the *PH*-distribution $F(\cdot)$ is then

$$F(x) = 1 - \alpha\ \exp(\boldsymbol{T}x)\boldsymbol{e}, \qquad \text{for } \boldsymbol{x} \geq \boldsymbol{0}.$$

The pair $(\alpha,\ \boldsymbol{T})$ is called a representation of $F(\cdot)$. The *PH*-distribution $F(\cdot)$ has a point mass α_0 at 0 and a density $F'(x) = -\exp(\boldsymbol{T}x)\boldsymbol{T}\boldsymbol{e} = \alpha\exp(\boldsymbol{T}\boldsymbol{x})\boldsymbol{T}^0$, on $(0, \infty)$. The Laplace-Stieltjes transform $f(s)$ of $F(\cdot)$ is

$$f(s) = \alpha_{m+1} + \boldsymbol{\alpha}(s\boldsymbol{I} - \boldsymbol{T})^{-1}\boldsymbol{T}^0, \quad \text{for Re } s \geq 0.$$

Its moments λ_v, $v \geq 1$, are all finite and given by $\lambda_v = (-1)^v\, v!\, \alpha T^{-v}\ \boldsymbol{e}$. Some special classes of

PH-distributions are the hyperexponential distributions

$$F(x) = \sum_{v=1}^{m} \alpha_v \left(1 - e^{-\lambda_v x}\right),$$

which may be represented by $\alpha = (\alpha_1, \ldots, \alpha_m)$, $\alpha_{m+1} = 0$, and $\boldsymbol{T} = -\text{diag}(\lambda_1, \ldots, \lambda_m)$, and the (mixed) Erlang distributions

$$F(x) = \sum_{v=1}^{m} p_v E_v(\lambda; x),$$

which are represented by $\alpha = (p_m, p_{m-1}, \ldots, p_1), \alpha_{m+1} = 0$, and

$$\boldsymbol{T} = \begin{bmatrix} -\lambda & \lambda & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\lambda & \lambda & \cdots & 0 & 0 & 0 \\ & \cdots & & & & \cdots & \\ 0 & 0 & 0 & \cdots & 0 & -\lambda & \lambda \\ 0 & 0 & 0 & \cdots & 0 & 0 & -\lambda \end{bmatrix}$$

Uses of Phase-type Distributions

The utility of *PH*-distributions is due to their closure properties, which allow standard operations such as convolution and mixing to be represented by matrix operations. Many classical simplifying properties of the exponential distribution have analogs in the matrix formalism for *PH*-distributions. In the analysis of probability models, *PH*-distributions often lead to tractable results without the severe restriction of exponential assumptions. Integrals involving *PH*-distributions also can usually be evaluated by stable recurrence relations or differential equations. Moreover, the phase-type distributions form a dense subset of the probability distributions on $[0, \infty)$, in that any such distribution can in principle be uniformly approximated by a sequence of *PH*-distributions.

Examples of closure properties are:

(a) If $F(\cdot)$ is a *PH*-distribution with representation $(\alpha, \boldsymbol{T})$ and mean λ_1', the corresponding delay distribution $F^*(\cdot)$ with density $(\lambda_1')^{-1}[1 - F(x)]$ is *PH* with representation $(\boldsymbol{\pi}, \boldsymbol{T})$ where $\boldsymbol{\pi} = (\lambda_1')^{-1}\alpha(-\boldsymbol{T})^{-1}$.

(b) If $F(\cdot)$ (with $\alpha_0 = 0$) is the service time distribution of a stable M/G/1 queue with arrival rate θ and service time distribution $H(\cdot)$ of mean μ_1', such that $\rho = \theta\mu_1' < 1$, the (steady-state) distribution $W(\cdot)$ of the waiting time is *PH*. Its representation is given by $(\boldsymbol{\gamma}, \boldsymbol{L})$, where $\boldsymbol{\gamma} = \rho\boldsymbol{\pi}$, $\boldsymbol{L} = \boldsymbol{T} + \rho\boldsymbol{T}^0\boldsymbol{\pi}$. For the *M/PH/1* queue, the distribution $W(\cdot)$ may therefore be computed by integrating a system of linear differential equations, rather than by solving the Pollaczek-Khinchin integral equation.

The fact that any probability distribution on $[0, \infty)$ can be approximated by *PH*-distributions is of somewhat limited practical application, although very good *PH*-approximations to classes such as the Weibull distributions have been obtained. Because of the following general result, that denseness property is, however, of considerable theoretical utility.

Suppose that a stochastic model involves one or more general probability distributions $F_j(\cdot)$, $1 \le j \le N$, on $[0, \infty)$, requiring evaluation of a continuous functional $\Phi[F_1(\cdot), \ldots, F_N(\cdot)]$. If an expression for $\Phi(\cdot)$ can be found for the case where $F_1(\cdot), \ldots, F_N(\cdot)$ are *PH*-distributions and if that expression does not explicitly depend on the formalism of *PH*-distributions, then it is also valid for arbitrary distributions $F_1(\cdot), \ldots, F_N(\cdot)$. This result has been used to establish various moment and other formulas in the theory of queues.

There is an extensive literature on phase-type distributions and their applications, including topics such as the structural geometric properties of families of *PH*-distributions, the approximation of other families of distributions by those of phase-type, and the fitting of *PH*-distributions to data. An important characterization of *PH*-distributions was proved in O'Cinneide (1990). Procedures for the approximation by *PH*-distributions are discussed in Asmussen et al. (1992), Johnson (1993) and Schmickler (1992). The appearance of phase-type distributions in some unexpected places in queueing theory was noted in Asmussen (1992).

See

▶ Erlang Distribution
▶ Hyperexponential Distribution

- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Queueing Theory

References

Asmussen, S. (1992). Phase-type representations in random walk and queueing problems. *Annals of Probability, 20*, 772–789.

Asmussen, S., Haggström, O., & Nerman, O. (1992). *EMPHT — A program for fitting phase-type distributions* (Studies in Statistical Quality Control and Reliability, Mathematical Statistics). Sweden: Chalmers University and University of Göteborg.

Johnson, M. A. (1993a). Selecting parameters of phase distributions: Combining nonlinear programming, heuristics, and Erlang distributions. *ORSA Journal on Computing, 5*, 69–83.

Johnson, M. A. (1993b). An empirical study of queueing approximations based on phase-type distributions. *Stochastic Models, 9*, 531–561.

Neuts, M. F. (1981). *Matrix-geometric solutions in stochastic models: An algorithmic approach*. Baltimore: The Johns Hopkins University Press (Reprinted by Dover Publications, 1994).

O'Cinneide, C. A. (1990). Characterization of phase-type distributions. *Stochastic Models, 6*, 1–57.

Pagano, M. E., & Neuts, M. F. (1981). Generating random variates from a distribution of phase type. In T. I. Oren, C. M. Delfosse, & C. M. Shub (Eds.), *1981 Winter simulation conference proceedings* (pp. 381–387). New Jersey: Institute of Electrical and Electronics Engineers.

Schmickler, L. (1992). MEDA: Mixed Erlang distributions as phase-type representations of empirical distribution functions. *Stochastic Models, 8*, 131–156.

Piecewise Linear Function

A function that is formed by linear segments or one that approximates a nonlinear function by linear segments.

Pivot Column

The column vector of coefficients associated with the entering basis variable in a simplex method iteration. Also, more generally, the column that contains the pivot element of a Gaussian elimination step or similar process.

See

- ▶ Eta Vector
- ▶ Gaussian Elimination
- ▶ Matrices and Matrix Algebra
- ▶ Pivot Element
- ▶ Pivot Row
- ▶ Simplex Method (Algorithm)

Pivot Element

In the simplex method, the coefficient of the pivot column whose row index corresponds to the basic variable that is to be dropped from the basis. Also, the element of the pivot column in a Gaussian elimination step that is selected to be on the diagonal of the associated upper triangular matrix.

See

- ▶ Eta Vector
- ▶ Gaussian Elimination
- ▶ Matrices and Matrix Algebra
- ▶ Pivot Column
- ▶ Pivot Row
- ▶ Simplex Method (Algorithm)

Pivot Row

The row corresponding to the position of the basic variable that is to be dropped from the basis in a simplex method iteration. In general, the row correspoding to the row position of a pivot element in a Gaussian elimination step.

See

- ▶ Eta Vector
- ▶ Gaussian Elimination
- ▶ Matrices and Matrix Algebra
- ▶ Pivot Column
- ▶ Pivot Element
- ▶ Simplex Method (Algorithm)

Pivot-Selection Rules

In the simplex method, the pivot selection rules determine which variable is to enter the basic solution and which variable is to be dropped. Depending on the solution at hand, the rules are designed to preserve feasibility (nonnegativity) of the solution (primal-simplex method), or to preserve the optimality conditions (dual-simplex method). In either case, the rules attempt to select an entering variable that would cause and improvement in the objective function. These rules are often augmented with anti-degeneracy or anticycling rules, and procedures for maintaining sparsity and numerical accuracy.

See

- ▶ Bland's Anticycling Rules
- ▶ Density
- ▶ Devex Pricing
- ▶ Linear Programming
- ▶ Matrices and Matrix Algebra
- ▶ Perturbation Methods
- ▶ Simplex Method (Algorithm)

PMF

Probability mass function.

PO

- ▶ Postoptimal Analysis

Point Stochastic Processes

Igor Ushakov
Qualcomm Inc., San Diego, CA, USA

Introduction

A point process is a stochastic process $\{N(t), t \geq 0\}$, where $N(t)$ = number of occurrences by time t, which describes the appearance of a sequence of instant random events in time. Usually (though not always) intervals between two neighboring events are considered to be independently distributed. A process of this type is called a point process with restricted memory. If times between occurrences are a sequence of independent and identically distributed (i.i.d.) random variables, the point process is called a renewal or recurrent point process. The Poisson process represents a particular case of a renewal process in which the intervals between occurrences are exponentially distributed (Cox and Isham, 1980; Daley and Vere-Jones, 2002, 2007; Franken et al. 1981).

A special type of point process can be formed by two independent subsequences of random variables that alternate, as in the sequence X_1, Y_1, X_2, Y_2,.... Such a process is called an alternating point process, and more specifically, an alternating renewal process if the X and Y subsequences are themselves ordinary renewal processes.

Thinning of a Point Process

In some cases, events are excluded from the point process with a specified probability. For instance, a unit failure leads to a system failure only if several additional random circumstances happen. This exclusion of events is called a thinning procedure. If the thinning procedure results in the (normalized) probability of the event exclusion going to 1, the resulting point process converges to a Poisson process. This statement is reflected in strong terms in Renyi's Limit Theorem and in its generalization made by Yu. K. Belyaev (see Gnedenko et al. 1969). For practical purposes, the result means that if the mean time between neighboring events in the initial recurrent process equals T, and each event is excluded from this process with the probability p close to 1, the resulting process will be a Poisson process with parameter

$$\lambda = \frac{1-p}{T}.$$

The Superposition of Point Processes

The next important statement concerns the superposition of point processes, which is formulated

in the Khinchine-Osokov Limit Theorem (Khinchine 1960; Osokov 1956) and later generalized in the Grigelionis-Pogozhev Limit Theorem (Grigelionis 1964; Pogozhev 1964). On a qualitative level, the theorem states that a limiting point process, which is formed by the superposition of independent "infinitesimally rare" point processes, converges to a Poisson process. For instance, if a piece of equipment consists of a large number of blocks and modules, the flow of its failures may well be considered to form a Poisson process. The parameter of this resulting process is expressed as a sum of the parameters of the initial processes, that is, if there are n recurrent processes ($n >> 1$), each of them with mean T_i, then the resulting process will be close to a Poisson process with parameter

$$\lambda = \sum_{1 \le i \le n} \frac{1}{T_i}.$$

As a consequence of these results, the Poisson process plays a role in the theory of stochastic processes that is analogous to that of the normal distribution in general probability and statistical theory.

See

- ▶ Poisson Process
- ▶ Queueing Theory
- ▶ Renewal Process
- ▶ Stochastic Model

References

Cox, D. R., & Isham, V. (1980). *Point processes*. New York: Chapman and Hall.

Daley, D. J., & Vere-Jones, D. (2002). *An introduction to the theory of point processes, volume 1: Elementary theory and methods* (2nd ed.). New York: Springer.

Daley, D. J., & Vere-Jones, D. (2007). *An introduction to the theory of point processes, volume 2: General theory and structure* (2nd ed.). New York: Springer.

Franken, P., König, D., Arndt, U., & Schmidt, V. (1981). *Queues and point processes*. Berlin, Germany: Akademie-Verlag.

Gnedenko, B. V., Belyaev, Y. K., & Solovyev, A. D. (1969). *Mathematical methods of reliability theory*. New York: Academic Press.

Grigelionis, B. I. (1964). Limit theorems for sums of renewal processes. In A. I. Berg, N. G. Bruevich, & B. V. Gnedenko (Eds.), *Cybernetics in the service of communism, vol. 2: reliability theory and queueing theory* (pp. 246–266). Moscow: Energiya.

Khintchine, A. Y. (1960). *Mathematical methods in the theory of queueing*. London: Charles Griffin.

Osokov, G. A. (1956). A limit theorem for flows of similar events. *Theory Probability and Its Applications, 1*, 246–255.

Pogozhev, I. B. (1964). Estimation of deviation of failure flow in multi-use equipment from Poisson Process (Russian). *Cybernetics in Service for Communism* (vol. 2). Moscow: Energiya.

Point-to-Set Map

A function that maps a point of one space into a subset of another.

Poisson Arrivals

Term used when customers coming to a queueing system follow a Poisson process; this also implies that the time between customer arrivals are independent and identical distributed random variables following an exponential distribution with mean equal to the inverse of the Poisson arrival rate.

See

- ▶ Exponential Arrivals
- ▶ Poisson Process
- ▶ Queueing Theory

Poisson Process

A stochastic, renewal-counting point process beginning from time $t = 0$ with $N(0) = 0$ that satisfies the following assumptions is called a Poisson process with rate λ: (1) the probability of one event happening in the interval $(t, t + h]$ is $\lambda h + o(h)$, where $o(h)$ is a function which goes to zero faster than h; (2) the probability of more than one event

happening in $(t, t + h]$ is $o(h)$; and (3) events happening in non-overlapping intervals are statistically independent. (Either (1) or (2) can be replaced by: the probability of no event happening in the interval $(t, t + h]$ is $1 - \lambda h + o(h)$). For such a Poisson process, the times between events (renewals) are independent and identically exponentially distributed with mean $1/\lambda$. In Kendall's queueing notation, arrivals following a Poisson process would be represented by "*M*" as in an *M*/*G*/*1* queue. An important property of Poisson arrival processes in queueing theory is PASTA (Poisson arrivals see time averages).

See

- ▶ Kendall's Notation
- ▶ Markov Chains
- ▶ Markov Processes
- ▶ PASTA
- ▶ Queueing Theory

Politics

Frederic H. Murphy[1], Sidney W. Hess[2] and Carlos G. Wong-Martinez[3]
[1]Temple University, Philadelphia, PA, USA
[2]Chadds Ford, Philadelphia, PA, USA
[3]Woosong University, Daejeon, Korea

Introduction

Applications of OR/MS to the representation and electoral processes are considered here. The narrower definition of politics is followed, denoting the theory and practice of managing political affairs in a party sense (Webster's New Collegiate Dictionary 1951). In particular, applications to the following are considered:

- Apportionment
- Districting
- Voting methods and logistics, and
- Election analysis

Apportionment

This is the process of equitably assigning a fixed number of legislators to a lesser number of political subdivisions. In the United States, 435 congressional districts must be apportioned to 50 states with each state receiving at least one district. The method of rounding to an integer solution influences the political result.

Balinski and Young (1982) have provided an exceptional mathematical analysis of the issue along with an historical, nontechnical exposition. In 1791, following the first U.S. census, Jefferson and Hamilton proposed alternative methods for apportionment, the method of greatest divisors (take the ratio of every state's population and the largest divisor such that the integer portions of the ratios add up to the number of representatives to allocate,) and the method of greatest remainders (take the population in a political unit, divide by the total population and multiply by the number of seats, allocate the integer portion, allocate the remaining seats in order of the size of the remainders until there are none left). Washington exercised the first presidential veto when he disagreed with Congress' support of Hamilton's method.

Most methods are biased; for example Jefferson's favors the more populated states while the method used in the United States since 1941, the "method of equal proportions" (also known as the Hill or Huntington method) discriminates against them. In this method a multiplier for adding the nth congressperson to a state is constructed by taking the square root of $1/[n(n-1)]$, $n > 1$. The product of the multipliers and the states' populations are sorted from highest to lowest for all states together. After each state is given one seat, the remaining seats are given to the 385 highest products of the populations and the multipliers. Other methods exhibit the paradox of a state's apportioned number of seats declining as the total number of representatives increases even when all states' populations are unchanged!

Balinski and Young (1982) conclude that there can be no perfect method. However, Senator Daniel Webster promoted a method called "major fractions" (frequently used between 1842 and 1932), which has been felt by many to be preferable. It is simple, and exhibits neither bias nor the population paradox.

Furthermore, Webster's method (find a divisor for the populations of each political unit such that the rounded quotients sum to the total number of legislators to be allocated), is more likely than the other methods to give each state its proportional number of seats, either rounded up or rounded down (Ernst 1994). See Apportionment Politics for detailed descriptions of apportionment methods and examples of the paradoxes that result from the different apportionment methods.

In most countries once the districts are established the candidate with the most votes wins. In Switzerland an alternative approach is taken to ensure that smaller parties are represented, Beroggi (2010). First, seats are allocated to states using major fractions. Second, seats are allocated to parties at the national level using the same method. With these allocations as constraints, the seats in every district are allocated to parties, minimizing the deviations between the real-valued allocation and integer number of seats ultimately given to each party in each district.

Redistricting

This is the process of defining geographic boundaries for the representatives in a political unit such as a city, state, province, or country. Historically, the party controlling the legislature draws districting maps to protect incumbents and increase their party's chances of maintaining control.

In 1962, the Supreme Court required population equality among districts, demanding more careful mapping than the usual prior political process (Baker v. Carr 1962). A variety of techniques to computerize the mapping process appeared. Most approaches incorporated population equality with the additional criteria that each district be:

- Contiguous, a single land parcel,
- Compact, consolidated rather than spread out, and
- Designed without political consideration.

Hess et al. (1965) solved a sequence of transportation linear programs. In each LP, equal population was allocated to trial district centers to minimize total cost. The measure of cost was compactness defined as the second moment of population about its district center. Centroids of the resultant districts became new centers for repeating the linear program. Successive solution of the transportation problems trended to more compactness while maintaining near population equality. Their heuristic handled problems as large as 350 population units by 19 districts. Larger problems were apportioned into smaller ones. This Ford Foundation-supported program was used for districting in at least seven states.

Hojati (1996) used Lagrangian relaxation to determine the center of districts and then the transportation model to assign population units to districts, followed by a capacitated transportation model to rejoin split population units. George et al. (1997) have generalized the transportation LP into a minimum-cost network-flow formulation that permits more flexible objective functions. They demonstrate objective (cost) functions that include penalties for:

- District populations deviating from the average or exceeding some maximum deviation,
- Districts crossing geographic barriers, and
- Changes from prior district boundaries.

The procedure has been applied in preparing New Zealand legislative-district boundaries involving assignment of 35,000 geographic units to 95 Parliamentary districts.

Garfinkel and Nemhauser (1969) developed a tree search algorithm that minimizes compactness while constraining maximum allowable population deviation. Their measure of district compactness is the diameter squared divided by area. Computation speed and capacity limited the problem size to about 50 population units by seven districts.

Nygreen (1988) redistricted Wales by three different solution methods: solving the integer programming formulation directly, using set partitioning (a variant of Garfinkel and Nemhauser's technique), and using implicit enumeration to structure the search of the tree of solutions. Although his example was small, he concluded that the integer programming technique was inferior. He felt problems to about 500 population units by 60 districts could be solved efficiently by set partitioning. Twenty years of computer improvement permit a tenfold larger problem!

All these redistricting techniques require apportioning a problem too large for solution into many smaller and solvable ones. Apportioning first has added benefits: small political subdivisions are more likely to remain intact and district boundaries will more often coincide with political boundaries.

Hess (1971) showed how first apportioning New York legislative seats to groups of counties minimizes the number of counties that must be in more than one district.

Mehrotra et al. (1998) model the problem as a constrained graph-partitioning problem as in Garfinkel and Nemhauser (1969) and develop a specialized branch-and-price based solution methodology rather than use implicit enumeration. Their reason for generating districts and solving the partitioning problem is to guarantee contiguous districts.

They did not work directly with the facility location/p median problem because ensuring contiguity would require an exponential number of constraints as with sub-tour elimination in the traveling salesman problem. Bozkaya et al. (2003) developed a tabu search approach to solving the problem while restricting the search to contiguous districts, again, not representing contiguity directly because of the perceived difficulty of capturing contiguity.

For such a heavily researched problem with so many successful researchers working on it over decades one would not expect an important breakthrough on a problem as difficult as the contiguity problem. However, two approaches represent contiguity directly in a model without any combinatorial explosion. Williams (2002) shows how to enforce contiguity using constraints on trees defined over the primal and dual planar graphs of the districts. Shirabe (2009), building on work by Zoltners and Sinha (1983), imposes contiguity by modeling trees with constraints that require the adjacent nodes connected by a positive flow be in the same district and the root node have an inflow that matches the number of geographic units assigned to the district. Thus, there has been substantial analytic progress in developing usable models for doing districting using integer programming formulations.

Meanwhile, the courts and legislatures have been slow to articulate permissible or required criteria for districting. A multitude of definitions or measures of compactness are available for Court selection, but all suffer from one flaw or another (Young 1988). In the United States "one man, one vote" is still the law of the land. The 1982 Voting Rights Act requires states with histories of racial discrimination to provide a reasonable chance of minority elections (Van Biema 1993). However, the Supreme Court (Shaw v. Hunt 1996) ruled that racial considerations cannot alone justify bizarre shaped districts. While the courts scrutinize the results of districting, they have not yet challenged the process (Browdy 1990), let alone find political gerrymandering to be unconstitutional. Associate Supreme Court Justice Breyer has regretted that the Court failed to take a stand (King 2010).

Political parties have been free to use proprietary software to generate districting plans that would make Governor Gerry blush. Computer services generated over 1,000 plans for Florida alone, making it difficult for the press and public to criticize gerrymandering (Miniter 1992). It is possible to predict when gerrymandering will happen: if only one political party controls the legislature and the politicians control the process without an independent oversight board, the districts will be drawn to the advantage of that party. That is, the process is important in determining the outcome.

The problem with gerrymandered districts after they are drawn is, like pornography, we know it when we see it. However, it is very difficult to define what gerrymandering is in advance. Consequently, any effort to reduce the degree of gerrymandering has to include not only good analytical models but also good governance processes.

Should the courts order an open districting process or bipartisanship necessitate, optimization models and algorithms could provide a viable approach to aid in redrawing representative boundaries (Browdy 1990). Given the unwillingness of politicians to give up the advantages that come from manipulating district boundaries, the likely eventual outcome will be a mix with optimization modeling establishing baselines and politicians making limited adjustments. Designing such a process will be an interesting challenge.

Voting Methods and Logistics

The application of approval voting was pioneered in the election processes of The Institute of Management Sciences (Fishburn and Little 1988). Here, a voter checks off (approves) any number of the candidates on a ballot, from a single one to potentially every one, with the person having the most checks being declared the winner. Regenwetter, and Grofman (1998) confirm the value of approval voting by examining the outcomes of seven elections, one of them being an INFORMS election.

Savas et al. (1972) reduced the number of New York City election districts by locating multiple voting machines at polling places. The City achieved significant cost savings and increased the probability voters would find functioning machines, without a significant increase in voter distance to the polls.

Election Analysis

The literature on OR in elections is sparse. The main roles seem to be in forecasting and game-theoretic analyses of policies. Barkan and Bruno (1972) used allocation techniques and statistical analysis to aid the 1970 California election campaign of Senator Tunney. Their analyses targeted precincts for voter registration and get-out-the-vote efforts. The key to their success was the ability to identify swing precincts by estimating party loyalty. Soberman and Sadoulet (2007) provide a game-theoretic analysis of rules to limit campaign spending.

A great deal of effort has been put into forecasting the outcome of elections. Campbell and Lewis-Beck (2008) survey past work in forecasting U.S. presidential elections and Lewis-Beck (2010) covers European election forecasting. Both of these articles are introductions to special issues on election forecasting, covering the broadly defined approaches of surveys, econometric analyses, and crowd sourcing such as the Iowa Electronic Market where people bet on the outcome and the prices and odds are set as in pari-mutuel betting. See also Kaplan and Barnett (2003).

See

- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Location Analysis
- ▶ Transportation Problem

References

Baker v. Carr. (1962). 369 U.S. 186.

Balinski, M. L., & Young, H. P. (1982). *Fair representation. Meeting the ideal of one man, one vote*. New Haven, CT: Yale University Press.

Barkan, J. D., & Bruno, J. E. (1972). Operations research in planning political campaign strategies. *Operations Research, 20*, 925–941.

Beroggi, G. E. G. (2010). When O.R. becomes the law. *OR/MS today, 37*(3), 44–47.

Bozkaya, B., Erkut, E., & Laporte, G. (2003). A tabu search heuristic and adaptive memory procedure for political districting. *European Journal of Operational Research, 144*, 12–26.

Browdy, M. H. (1990). Computer models and post-Bandemer redistricting. *Yale Law Journal, 99*, 1379–1398.

Campbell, J. E., & Lewis-Beck, M. S. (2008). US presidential election forecasting: An introduction. *International Journal of Forecasting, 24*, 189–192.

Ernst, L. R. (1994). Apportionment methods for the house of representatives and the court challenges. *Management Science, 40*, 1207–1227.

Fishburn, P. C., & Little, J. D. C. (1988). An experiment in approval voting. *Management Science, 34*, 555–568.

Garfinkel, R. S., & Nemhauser, G. L. (1969). Optimal political districting by implicit enumeration techniques. *Management Science, 16*, B495–B508.

George, J. A., Lamar, B. W., & Wallace, C. A. (1997). Political district determination using large-scale network optimization. *Socio-Economic Planning Sciences, 31*, 11–28.

Hess, S. W. (1971). One-man one-vote and county political integrity: Apportion to satisfy both. *Jurimetrics Journal, 11*, 123–141.

Hess, S. W., Weaver, J. B., Seigfeldt, H. J., Whelan, J. N., & Zitlau, P. A. (1965). Nonpartisan political redistricting by computer. *Operations Research, 13*, 998–1006.

Hojati, M. (1996). Optimal political districting. *Computers and Operations Research, 23*(12), 1147–1161.

Kaplan, E., & Barnett, A. (2003). A new approach to estimating the probability of winning the presidency. *Operations Research, 51*(1), 32–40.

King, L. (2010). CNN television interview of associate Supreme Court Justice Stephen Breyer, September 15.

Lewis-Beck, M. S. (2010). European election forecasting: An introduction. *International Journal of Forecasting, 26*, 9–10. Intro to special issue on election forecasting in Europe.

Mehrotra, A., Johnson, E. L., & Nemhauser, G. L. (1998). An optimization based heuristic for political districting. *Operations Research, 44*(8), 1100–1114.

Miniter, R. (1992). Running against the computer; Stephen Solarz and the technician-designed congressional district. *The Washington Post*, September 20, C5.

Nygreen, B. (1988). European assembly constituencies for Wales – Comparing of methods for solving a political districting problem. *Mathematical Programming, 42*, 159–169.

Regenwetter, M., & Grofman, B. (1998). Approval voting, Borda winners, and Condorcet winners: Evidence from seven elections. *Management Science, 44*(4), 520–533.

Savas, E. S., Lipton, H., & Burkholz, L. (1972). Implementation of an OR approach for forming efficient districts. *Operations Research, 20*, 46–48.

Shaw v. Hunt. (1996). 116 S. Ct. 1894, 135 L. Ed. 2d 207.

Shirabe, T. (2009). Districting modeling with exact contiguity constraints. *Environment and Planning B: Planning and Design, 36*, 1053–1066.

Soberman, D., & Sadoulet, L. (2007). Campaign spending limits and political advertising. *Management Science, 53*(10), 1521–1532.
Van Biema, D. (1993). Snakes or ladders. *Time, 12*, 30–33.
Webster's New Collegiate Dictionary. (1951). *Politics* (p. 654). New York: Mirriam.
Wikipedia, Apportionment. Retrieved from http://en.wikipedia.org/wiki/Apportionment_%28politics%29
Williams, J. C. (2002). A zero-one programming model for contiguous land acquisition. *Geographical Analysis, 34*(4), 330–349.
Young, H. P. (1988). Measuring the compactness of legislative districts. *Legislative Studies Quarterly, 13*(1), 105–115.
Zoltners, A. A., & Sinha, P. (1983). Sales territory alignment: A review and model. *Management Science, 29*, 1237–1256.

Pollaczek-Khintchine Formula

For the M/G/1 queueing system, with L defined as the steady-state expected number of customers in the system, λ the customer arrival rate, $1/\mu$ the mean service time and σ^2 the variance of the service distribution, the Pollaczek-Khintchine (P-K) (mean-value) formula gives

$$L = \rho + \left(\rho^2 + \lambda^2\sigma^2\right)/[2(1-\rho)]$$

where $\rho = \lambda/\mu$. Sometimes, the formulas for mean queue size, L_q, mean line delay, W_q, and mean system waiting time, W, which can be easily derived from L using Little's formula, are also called the P-K formulas. More generally, there are associated transform relationships giving the generating function of the steady-state number in system (or queue length) and the Laplace transform of the steady-state delay/waiting times in terms of the Laplace transform of the service time distribution, which are referred to as Pollaczek-Khintchine (P-K) transform formulas.

See

▶ Queueing Theory

Polling System

Where a single server visits each group of customers (queue) in cyclic order and then polls to see if there is anyone present. If yes, the service facility serves those customers under such rules as gated (serve only those present when polled) or exhaustive (serve until no customers are left at the location).

See

▶ Networks of Queues
▶ Queueing Theory

Polyhedron

The solution space defined by the intersection of a finite number of linear constraints, an example of which is the solution space of a linear-programming problem. Such a space is convex.

See

▶ Convex Set
▶ Linear Programming

Polynomial Hierarchy

A general term used to refer to all of the various computational complexity classes.

See

▶ Computational Complexity

Polynomially Bounded (−Time) Algorithm (Polynomial Algorithm)

An algorithm for which it can be shown that the number of steps required to find a solution to a problem is bounded by a polynomial function of the problem's data.

See

- ▶ Computational Complexity
- ▶ Exponential-Bounded (–Time) Algorithm

Polynomial-Time

- ▶ Computational Complexity

Polynomial-Time Reductions and Transformations

- ▶ Computational Complexity

POMDP

- ▶ Partially Observed Markov Decision Processes

Population-based Search Methods

Optimization search methods that propagate a population of solutions from iteration to iteration of the algorithm, generally using evolutionary operators. Examples include genetic algorithms, ant colony optimization, and particle swarm optimization.

See

- ▶ Evolutionary Algorithms
- ▶ Genetic Algorithms
- ▶ Particle Swarm Optimization
- ▶ Swarm Intelligence

Portfolio Analysis

- ▶ Financial Engineering
- ▶ Portfolio Theory: Mean-Variance Model

Portfolio Theory: Mean-Variance Model

John L. G. Board[1], Charles M. S. Sutcliffe[2] and William T. Ziemba[3,4]
[1]Henley Business School, University of Reading, Reading, UK
[2]University of Reading, Reading, UK
[3]University of British Columbia, Vancouver, British Columbia, Canada
[4]Oxford University, Oxford, UK

Introduction

The heart of the portfolio problem is the selection of an optimal set of investment assets by rational economic agents. Although elements of portfolio problems were discussed in the 1930s and 1950s by Allais, De Finetti, Hicks, Marschak and others, the first formal specification of such a selection model was by Markowitz (1952, 1959), who defined a mean-variance model for calculating optimal portfolios. Following Tobin (1958, 1965), Sharpe (1970) and Roll (1972), this portfolio selection model may be stated as

$$\begin{aligned} &\text{Minimize} && \boldsymbol{x}'\boldsymbol{V}\boldsymbol{x} \\ &\text{subject to} && \boldsymbol{x}'\boldsymbol{r} = r_p \\ & && \boldsymbol{x}'\boldsymbol{e} = 1 \end{aligned} \tag{1}$$

where $\boldsymbol{x}$ is a column vector of investment proportions in each of the risky assets, $\boldsymbol{V}$ is a positive semi-definite variance-covariance matrix of asset returns, $\boldsymbol{r}$ is a column vector of expected asset returns, r_p is the investor's target rate of return and $\boldsymbol{e}$ is a column unit vector. An explicit solution for the problem can be found using the procedures described in Merton (1972), Ziemba and Vickson (1975), or Roll (1972).

Restrictions on short selling can be modeled by augmenting (1) by the constraints

$$\boldsymbol{x} \geq \boldsymbol{0} \tag{2}$$

where $\boldsymbol{0}$ is a column vector of zeros. The problem now becomes a classic example of quadratic mathematical programming; indeed, the development of the portfolio problem coincided with early developments in nonlinear programming. Formal investigations of the properties of both formulations,

and variants, appear in Szegö (1980), Huang and Litzenberger (1988), and the references above.

The Use of Mean and Variance

The economic justification for this model is based on the von Neumann-Morgenstern expected utility results, discussed in this context by Markowitz (1959). The model can also be viewed in terms of consumer choice theory together with the characteristics model developed by Lancaster (1971). His argument is that goods purchased by consumers seldom yield a single, well-defined service; instead, each good may be viewed as a collection of attributes, each of which gives the consumer some benefit (or disbenefit). Thus, preference is defined over those characteristics embodied in a good rather than over the good itself. The analysis focuses attention on the attributes of assets rather than on the assets per se. This requires the assumption that utility depends only on the characteristics. With k characteristics, C_k,

$$U = f(W) = g(C_l, \ldots, C_k)$$

where U and W represent utility and wealth. Modeling too few characteristics will yield apparently false empirical results. Clearly, the benefits of this approach increase as the number of assets rises relative to the number of characteristics. The objects of choice are the characteristics $C_1, \ldots, C_k$. In portfolio theory, these are taken to be payoff (return) and risk.

At Markowitz's suggestion, when dealing with choice among risky assets, payoff is measured as the expected return of the distribution of returns, and risk by the standard deviation of returns. Apart from minor exceptions (Ziemba and Vickson 1975), this pair of characteristics form a complete description of assets which is consistent with expected utility theory in only two cases: assets have normal distributions, or investors have quadratic utility of wealth functions. The adequacy of these assumptions has been investigated by a number of authors (e.g., Borch 1969; Feldstein 1969; Tsiang 1972). Although returns have been found to be non-normal and the quadratic utility has a number of objectionable features (not least diminishing marginal utility of wealth for high wealth), several authors demonstrate approximation results that are sufficient for mean-variance analysis (Samuelson 1970; Ohlson 1975; Levy and Markowitz 1979).

A number of authors, including Markowitz (1959), consider alternatives to the variance and suggest the use of the semi-variance. This suggestion has been extended into workable portfolio selection rules. Fama (1971) and Tsiang (1973) have argued the usefulness of the semi-interquartile range as a measure of risk. Kraus and Litzenberger (1976) and others have examined the effect of preferences defined in terms of the third moment, which allows investor choice in terms of skewness. Kallberg and Ziemba (1979, 1983) show that risk aversion preferences are sufficient to determine optimal portfolio choice if assets have normally distributed returns whatever the form of the assumed, concave, utility function.

Solution of Portfolio Selection Model

In the absence of short sales restrictions, (1) can be rewritten as

$$\begin{aligned}\text{Minimize } \boldsymbol{L} = \tfrac{1}{2}\boldsymbol{x}'\boldsymbol{V}\boldsymbol{x} - \lambda_1(\boldsymbol{x}'\boldsymbol{r} - r_p) \\ - \lambda_2(\boldsymbol{x}'\boldsymbol{e} - 1)\end{aligned} \tag{3}$$

The first-order conditions are

$$\boldsymbol{V}\boldsymbol{x} = \lambda_1\boldsymbol{r} + \lambda_2\boldsymbol{e}$$

which shows that, for any efficient $\boldsymbol{x}$, there is a linear relation between expected returns $\boldsymbol{r}$ and their covariances, $\boldsymbol{V}\boldsymbol{x}$.

Solving for $\boldsymbol{x}$:

$$\boldsymbol{x} = \lambda_1\boldsymbol{V}^{-1}\boldsymbol{r} + \lambda_2\boldsymbol{V}^{-1}\boldsymbol{e} = \boldsymbol{V}^{-1}[\boldsymbol{r}\ \boldsymbol{e}]\boldsymbol{A}^{-1}[r_p 1]' \tag{4}$$

where

$$\boldsymbol{A} = \begin{bmatrix} a & b \\ b & c \end{bmatrix} = \begin{bmatrix} \boldsymbol{r}'\boldsymbol{V}^{-1}\boldsymbol{r} & \boldsymbol{r}'\boldsymbol{V}^{-1}\boldsymbol{e} \\ \boldsymbol{r}'\boldsymbol{V}^{-1}\boldsymbol{e} & \boldsymbol{e}'\boldsymbol{V}^{-1}\boldsymbol{e} \end{bmatrix}$$

Substituting (4) into the definition of portfolio variance, $\boldsymbol{x}'\boldsymbol{V}\boldsymbol{x}$, yields

$$\begin{aligned}\boldsymbol{V}_p &= [r_p 1]\boldsymbol{A}^{-1}[r_p 1]', \text{ and} \\ \boldsymbol{S}_p &= \left[\frac{cr_p^2 - 2br_p + a}{ac - b^2}\right]^{1/2}\end{aligned} \tag{5}$$

where V_p and S_p represent portfolio variance and standard deviation, respectively. This defines the efficient set, which is a hyperbola in mean/standard-deviation space (or a parabola in mean/variance space). The minimum risk is at $S_{\min} = c^{1/2}$ and $r_{\min} = b/c$ (both strictly positive). Rational risk averse investors will hold portfolios lying on this boundary with $r \geq r_{\min}$.

Each efficient portfolio, p, has an orthogonal portfolio z (i.e., such that $\mathrm{Cov}(r_p, r_z) = 0$) with return

$$r_z = (a - br_p)/(b - cr_p)$$

Using this, the efficient set degenerates into the straight line tangent to the hyperbola at p which has intercept $\boldsymbol{r}_z$,

$$\boldsymbol{r} = \boldsymbol{r}_z + \lambda \boldsymbol{s} \tag{6}$$

where $\boldsymbol{r}$ and $\boldsymbol{s}$ represent vectors of the expected return and risks of efficient portfolios, and $\lambda = (r_p - r_z)/S_p$ can be interpreted as the additional expected return per unit of risk. This is known as the Sharpe ratio (Sharpe 1966, 1994). Equation (6) shows a two-fund separation theorem, such that linear combinations of only two portfolios are sufficient to describe the entire efficient set.

Under the additional assumptions of homogeneous beliefs (so that all investors perceive the same parameters) and equilibrium, (6) becomes the Capital Market Line. The Security Market Line (i.e., the relationship between expected returns and systematic risk or $\boldsymbol{\beta}$), which is the outcome of the Capital Asset Pricing Model (CAPM), can be derived by pre-multiplying (4) by $\boldsymbol{V}$ and simplifying using the definitions of V_p and r_z:

$$\boldsymbol{r} = r_z \boldsymbol{e} + (r_p - r_z)\boldsymbol{\beta} \tag{7}$$

where $\boldsymbol{\beta} = \boldsymbol{V}\boldsymbol{x}/V_p$. If it exists, the risk-free rate of interest may be substituted for r_z (definitionally, the risk-free return will be uncorrelated with the return on all risky assets). Equation (7) then becomes the original CAPM in which expected return is calculated as the risk-free rate plus a risk premium (measured in terms of an asset's covariance with the market portfolio). The CAPM forms one of the cornerstones of modern finance theory and is not appropriately addressed here. Discussion of the CAPM can be found in Huang and Litzenberger (1988) and Ferson (1995), while systematic fundamental and seasonal violations of the theory are presented in Ziemba (1994) and Keim and Ziemba (1999).

Short Selling

The assumption that assets may be sold short (i.e., $x_i < 0$) is justified when the model is used to derive analytical results for the portfolio problem. Also, when considering equilibrium (e.g., the CAPM), none of the short selling constraints should be binding (because in aggregate, short selling must net out to zero). However, significant short selling restrictions do face investors in most real markets. These restrictions may be in the form of absolute prohibition, the extra cost of deposits to back short selling or self imposed controls designed to limit potential losses.

The set of quadratic programming problems to find the efficient frontier when short sales are ruled out can be formulated as either minimizing the portfolio risk for a specified sequence of portfolio returns (r_p) by repeatedly solving (1) and (2), or maximizing the weighted sum of portfolio risk and return for a chosen range of risk-return tradeoff parameters (μ) by repeatedly solving (8) as below. This latter approach has the advantages of locating only points on the efficient frontier and, for evenly spaced increments in μ, locating more points on the efficient frontier where its curvature is greatest:

$$\begin{aligned} \text{Maximize} \quad & \alpha = \boldsymbol{x}'\boldsymbol{V}\boldsymbol{x} - \mu(\boldsymbol{x}'\boldsymbol{r} - r_p) \\ \text{Subject to} \quad & \boldsymbol{x} \geq \boldsymbol{0} \\ & \boldsymbol{x}'\boldsymbol{e} = 1 \end{aligned} \tag{8}$$

When short sales are permitted, a position (long or short) is taken in every asset, while when short selling is ruled out, the solution involves long positions in only about 10% of the available assets. When short selling is permitted, about half the assets are required to be sold short, often in large amounts, and sometimes in amounts exceeding the initial value of the investment portfolio. Indeed, this is the main activity of 'short seller' funds.

In contrast, most models based on portfolio theory, in particular the CAPM, ignore short selling

constraints (Markowitz 1983, 1987). This change is consistent with the development of equilibrium models for which institutional restrictions are inappropriate (and if imposed would not be binding). However, when short selling is permitted, the number of asset return observations is required to exceed the number of assets, while complementary slackness means that this condition need not be met when short selling is ruled out. Computational procedures to solve mean-variance models with various types of constraints, and the optimal combination of safe and risky assets for various utility functions are discussed by Ziemba et al. (1974).

Estimation Problems

The model (1) requires estimates of $\boldsymbol{r}$ and $\boldsymbol{V}$ for the period during which the portfolio is to be held. This estimation problem has been given relatively little attention, and many authors, both practitioners and academics, have used historical values as if they were precise estimates of future values. However, Hodges and Brealey (1973), among others, demonstrate the benefits obtained from even slight improvements on historical data.

Estimation risk can be allowed for either by using different methods to forecast asset returns, variances and covariances, which are then used in place of the historical values in the portfolio model, or by using the historical values in a modified portfolio selection technique (Bawa et al. 1979). Since the portfolio selection model of Markowitz takes these estimates as parametric, there is no theoretical guidance on the estimation method and a variety of methods have been proposed to provide the estimates. The single index market model of Sharpe (1963) has been widely applied in the literature to forecast the covariance matrix. Originally proposed to reduce the computation required by the full model, it assumes a linear relation between stock returns and some measure of the market, $\boldsymbol{r} = \boldsymbol{\alpha} + \boldsymbol{\beta}'\boldsymbol{m} = \boldsymbol{\varepsilon}$ (for market index $\boldsymbol{m}$ and residuals $\boldsymbol{\varepsilon}$). This uses historical estimates of the means and variances. However, the implied covariance matrix is $\boldsymbol{V}_1 = \upsilon_m \boldsymbol{\beta}\boldsymbol{\beta}' + \boldsymbol{V}$, where υ_m is the variance of the index, $\boldsymbol{\beta}$ is a column vector of slope coefficients from regressing each asset on the market index and $\boldsymbol{V}$ is a diagonal matrix of the variances of the residuals from each of these regressions. A number of studies have found that models based on the single index model outperform those based on the full historical method (e.g., Board and Sutcliffe 1994).

The overall mean method, first proposed by Elton and Gruber (1973), is based on the finding that, although historical estimates of means are satisfactory, data are typically not stable enough to allow accurate estimation of the $N(N-1)/2$ covariance terms. The crudest solution is to assume that the correlations between all pairs of assets expected in the next period are equal to the mean of all the historic correlations. An estimate of $\boldsymbol{V}$ can then be derived from this. Elton et al. (1978) compared the overall mean method of forecasting the covariance matrix with forecasts made using historical values, and four alternative versions of the single index model. They concluded that the overall mean model was clearly superior. A simplified procedure for estimating the overall mean correlation appears in Aneja et al. (1989).

Statisticians have shown increasing interest in Bayesian methods (Hodges 1976) and particularly James-Stein estimators (Efron and Morris 1975, 1977; Judge and Bock 1978; Morris 1983). The intuition behind this approach is that returns that are far from the norm have a higher chance of containing measurement error than those close to it. Thus, estimates of returns, based on individual share data, are cross-sectionally 'shrunk' towards a global estimate of expected returns which is based on all the data. Although these estimators have unusual properties, they are generally expected to perform well in large samples.

Jorion (1985, 1986) examined the performance of Bayes-Stein estimation using both simulated and small real data sets and concluded that the Bayes-Stein approach outperformed the use of historical estimates of returns and the covariance matrix. However, Jorion (1991) found that the index model outperformed Stein and historical models. Board and Sutcliffe (1994) applied these and other methods to large data sets. They found that, in contrast to earlier studies, the relative performance of Bayes-Stein was mixed. While it produced reasonable estimates of the mean returns vector, there were superior methods (e.g., use of the overall mean) for estimating the covariance matrix when short sales

were permitted. They also found that, when short sales were prohibited, actual portfolio performance was clearly improved, although there was little to choose between the various estimation methods.

An alternative approach is to try to control for errors in the parameter estimates by imposing additional constraints on (1). Clearly, ex-ante the solution to such a model cannot dominate (1), however, ex-post, dominance might emerge (i.e., what seems, in advance, to be an inferior portfolio might actually perform better than others). The argument is that adding constraints to (1) to impose lower bounds (i.e., prohibiting short sales) and/or upper bounds (forcing diversification) can be used as an ad hoc method of avoiding the worst effects of estimation risk. Of course, extreme, but possibly desirable, corner solutions will also be excluded by this technique. Cohen and Pogue (1967) imposed upper bounds of 2.5% on any asset. Board and Sutcliffe (1988) studied the effects of placing upper bounds on the investment proportions, which may be interpreted as a response to estimation risk. Using historical forecasts of returns and the covariance matrix, and with short sales excluded, they found that forcing diversification leads to improved actual performance over the unconstrained model. Hensel and Turner (1998) have also studied adjusting the inputs and outputs to improve portfolio performance.

Chopra and Ziemba (1993), following the work of Kallberg and Ziemba (1984), showed that errors in the mean values have a much greater effect than errors in the variances, which are in turn more important than errors in the covariances. Their simulations show errors of the order of 20 to 2 to 1. This quantifies the earlier findings and stresses the importance of having good estimates of the asset means.

Another approach is to use fundamental analysis to provide external information to modify the estimates (Hodges and Brealey 1973). Clearly, among the simplest external data to add are the seasonal (e.g., turn of the year, and month and weekend) effects that have been found in most stock markets around the world. Incorporation of these into the parameter estimates can substantially improve the performance of the model. Ziemba (1994) demonstrated the benefits of factor models to estimate the mean returns.

Concluding Remarks

Only the single period mean-variance portfolio theory model has been considered here. Most of the extensions to multi-period models assume frictionless capital markets, which require the solution of a sequence of instantaneous mean-variance models in which the existence of transactions costs adds enormously to the complexity of the problem. Surveys covering dynamic portfolio theory appear in Constantinides and Malliaris (1995), Ziemba and Vickson (1975), Huang and Litzenberger (1988), and Ingersoll (1987); see also Ziemba and Mulvey (1998).

See

- ▶ Banking
- ▶ Financial Engineering
- ▶ Financial Markets
- ▶ Linear Programming
- ▶ Nonlinear Programming
- ▶ Quadratic Programming

References

Aneja, Y. P., Chandra, R., & Gunay, E. (1989). A portfolio approach to estimating the average correlation coefficient for the constant correlation model. *Journal of Finance, 44*, 1435–1438.

Bawa, V. S., Brown, S. J., & Klein, R. W. (1979). *Estimation risk and optimal portfolio choice*. Amsterdam: North Holland.

Board, J. L. G., & Sutcliffe, C. M. S. (1988). Forced diversification. *Quarterly Review Economics and Business, 28*(3), 43–52.

Board, J. L. G., & Sutcliffe, C. M. S. (1994). Estimation methods in portfolio selection and the effectiveness of short sales restrictions: UK evidence. *Management Science, 40*, 516–534.

Borch, K. (1969). A note on uncertainty and indifference curves. *Review Economic Studies, 36*, 1–4.

Chopra, V. R. (1993). Improving optimization. *Journal of Investing, 2*, 51–59.

Chopra, V. R., & Ziemba, W. T. (1993). The effect of errors in means, variances and covariances on optimal portfolio choice. *Journal of Portfolio Management, 19*(2), 6–13.

Cohen, K. J., & Pogue, J. A. (1967). An empirical evaluation of alternative portfolio selection models. *Journal of Business, 40*, 166–193.

Constantinides, G., & Malliaris, G. (1995). Portfolio theory. In R. A. Jarrow, V. Maksimovic, & W. T. Ziemba (Eds.), *Handbook of finance*. Amsterdam: North-Holland.

Efron, B., & Morris, C. (1975). Data analysis using stein's estimator and its generalizations. *Journal of American Statistical Association, 70*, 311–319.

Efron, B., & Morris, C. (1977). Stein's paradox in statistics. *Scientific American, 236*(5), 119–127.

Elton, E. J., & Gruber, M. J. (1973). Estimating the dependence structure of share prices: Implications for portfolio selection. *Journal of Finance, 28*, 1203–1232.

Elton, E. J., Gruber, M. J., & Urich, T. J. (1978). Are betas best? *Journal of Finance, 33*, 1375–1384.

Fama, E. (1971). Risk, return and equilibrium. *Journal of Political Economy, 79*, 30–55.

Fama, E. F. (1976). *Foundations of finance*. Oxford: Basil Blackwell.

Feldstein, M. (1969). Mean variance analysis in the theory of liquidity preference and portfolio selection. *Review Economic Studies, 36*, 5–12.

Ferson, W. (1995). Theory and testing of asset pricing models. In Jarrow, Maksimovic, & Zeimba (Eds.), *Handbook of finance*. Amsterdam: North-Holland.

Hensel, C. R., & Turner, A. L. (1998). Making superior asset allocation decisions: A practitioner's guide. In Ziemba & Mulvey (Eds.), *Worldwide asset and liability modelling* (pp. 62–83). Cambridge: Cambridge University Press.

Hodges, S. D. (1976). Problems in the application of portfolio selection. *Omega, 4*, 699–709.

Hodges, S. D., & Brealey, R. A. (1973). Portfolio selection in a dynamic and uncertain world. *Journal of Financial Analysts, 29*, 50–65.

Huang, C. F., & Litzenberger, R. H. (1988). *Foundations for financial economics*. Amsterdam: North-Holland.

Ingersoll, J. (1987). *Theory of financial decision making*. Lanham: Rowman & Littlefield.

Jarrow, R., Maksimovic, V., & Ziemba, W. T. (Eds.). (1995). *Finance*. Amsterdam: North-Holland.

Jobson, J. D., & Korkie, B. (1981). Putting markowitz theory to work. *Journal of Portfolio Management, 7*, 70–74.

Jobson, J. D., Korkie, B., & Ratti, V. (1979). Improved estimation for Markowitz portfolios using James-Stein type estimators. *Proceedings Business Economics and Statistics Section, American Statistical Association, 41*, 279–284.

Jorion, P. (1985). International portfolio diversification with estimation error. *Journal of Business, 58*, 259–278.

Jorion, P. (1986). Bayes-Stein estimation for portfolio analysis. *Journal of Financial and Quantitative Analysis, 21*, 279–292.

Jorion, P. (1991). Bayesian and CAPM estimators of the means: Implications for portfolio selection. *Journal of Banking and Finance, 15*, 717–727.

Judge, G. G., & Bock, M. E. (1978). *The statistical implications of pre-test and stein-rule estimators in econometrics*. Amsterdam: North-Holland.

Kallberg, J. G., & Ziemba, W. T. (1979). On the robustness of the Arrow-Pratt risk aversion measure. *Economics Letters, 2*, 21–26.

Kallberg, J. G., & Ziemba, W. T. (1983). Comparison of alternative utility functions in portfolio selection. *Management Science, 29*, 1257–1276.

Kallberg, J. G. & Ziemba, W. T. (1984). Mis-specification in Portfolio Selection Problems. In G. Bamberg & K. Spremann, (Eds.)., Risk and Capital : Lecture Notes In Economic and Mathematical Systems, Springer-Verlag, New York.

Keim, D. B., & Ziemba, W. T. (Eds.). (1999). *Security market imperfections in worldwide equity markets*. Cambridge: Cambridge University Press.

Kraus, A., & Litzenberger, R. F. (1976). Skewness preference and the valuation of risk assets. *Journal of Finance, 31*, 1085–1100.

Lancaster, K. (1971). *Consumer demand: a new approach*. New York: Columbia University Press.

Levy, H. (1969). A utility function depending on the first three moments. *Journal of Finance, 24*, 715–719.

Levy, H., & Markowitz, H. (1979). Approximating executed utility by a function of mean and variance. *American Economic Review, 69*, 308–317.

Markowitz, H. M. (1952). Portfolio selection. *Journal of Finance, 7*, 77–91.

Markowitz, H. M. (1959). *Portfolio selection: Efficient diversification of investments*. New Haven: Yale University Press.

Markowitz, H. M. (1983). Nonnegative or not non-negative: A question about CAPMs. *Journal of Finance, 38*, 283–295.

Markowitz, H. M. (1987). *Mean-variance in portfolio choice and capital Markets*. Oxford: Blackwell.

Merton, R. C. (1972). An analytic derivation of the efficient portfolio frontier. *Journal of Financial and Quantitative Analysis, 7*, 1851–1872.

Morris, C. (1983). Parametric empirical Bayes inference: Theory and applications. *Journal of American Statistical Association, 78*, 47–55.

Ohlson, J. (1975). Asymptotic validity of quadratic utility as the trading interval approaches zero. In W. T. Ziemba & R. G. Vickson (Eds.), *Stochastic optimization models in finance*. New York: Academic.

Roll, R. (1972). A critique of the asset pricing theory's tests. *Journal of Financial Economics, 4*, 129–176.

Samuelson, P. (1970). The fundamental approximation theorem of portfolio analysis in terms of means variances and higher moments. *Review of Economic Studies, 37*, 537–542.

Sharpe, W. F. (1963). A simplified model for portfolio analysis. *Management science; Operations research (OR), Practice of Management Science, 9*, 277–293.

Sharpe, W. F. (1966). Mutual fund performance. *Journal of Business, 39*, 119–138.

Sharpe, W. F. (1970). *Portfolio theory and capital markets*. New York: McGraw-Hill.

Sharpe, W. F. (1994). The sharpe ratio. *Journal of Portfolio Management, Fall*, 59–68.

Szegö, G. P. (1980). *Portfolio theory, with application to bank assess management*. New York: Academic.

Tobin, J. (1958). Liquidity preference as behaviour towards risk. *Review of Economic Studies, 26*, 65–86.

Tobin, J. (1965). The theory of portfolio selection. In F. H. Hahn & F. P. R. Brechling (Eds.), *The theory of interest rates* (International Economic Association, pp. 3–51). London: Macmillan.

Tsiang, S. (1972). The rationale of the mean standard deviation analysis, skewness preference and the demand for money. *American Economic Review, 62*, 354–371.

Tsiang, S. (1973). Risk, return and portfolio analysis: Comment. *Journal of Political Economy, 81*, 748–751.

Ziemba, W. T. (1994). World wide security market regularities. *European Journal of Operational Research, 74*, 198–229.
Ziemba, W. T., & Mulvey, J. M. (Eds.). (1998). *World-wide asset and liability modelling*. Cambridge: Cambridge University Press.
Ziemba, W. T., Parkan, C., & Brooks-Hill, F. J. (1974). Calculation of investment portfolios with risk free borrowing and lending. *Management Science, 21*, 209–222.
Ziemba, W. T., & Vickson, R. G. (Eds.). (1975). *Stochastic optimization models in finance*. New York: Academic.

POS

Point of sale.

See

► Retailing

Postoptimal Analysis

The study of how a solution changes with respect to (usually) small changes in the problem's data. In particular, this term is applied to the sensitivity analysis and parametric analysis of a solution to a linear-programming problem.

See

► Linear Programming
► Parametric Programming
► Sensitivity Analysis

Posynomial Programming

► Geometric Programming

Power Model

► Learning Curves

PP

► Parametric Programming

PPB(S)

Planning-programming-budgeting (system).

See

► Cost Analysis
► Military Operations Research

Practice of Operations Research and Management Science

Hugh J. Miser
Farmington, CT, USA

Introduction

The practice of OR/MS here will mean using the appropriate models, tools, techniques, and craft skills of these sciences to understand the problems of people/machine/nature systems with a view toward ameliorating these problems, possibly by new understandings, new decisions, new procedures, new structures, or new policies. Such practice calls for a suitable form of professionalism in dealing not only with the phenomena of the problem situation but also with the persons with relevant responsibilities, as well as other parties at interest.

OR/MS as a Science

Following Ravetz (1971), science in general may be described as "craft work operating on intellectually constructed objects," each object defining a class. Scientific work is thus aimed at establishing new properties of these objects and verifying that they reflect the reality of the classes of phenomena that

they represent (Miser 1993). This description has four implications:

1. The intellectual objects – that OR/MS workers usually call models – are created by the imagination, informed by earlier knowledge of the phenomena and objects that have described them successfully, as well as innovative ideas or new evidence from reality.
2. There is a continuing reference to the phenomena of reality.
3. Scientific inquiry then becomes the search for new properties of the classes both by manipulating the objects and seeking new evidence from reality as a basis for revising them.
4. The new properties deduced from the objects – or models – must then be compared with the appropriate aspects of the phenomena of reality.

It is essential to observe that the different sciences – such as physics, biology, or OR/MS – are distinguished, not by their methods, techniques, or models (many of which are widely shared among the sciences), but by the portions of reality in which they are undertaking to understand, explain, and solve problems (Kemeny 1959).

Within the framework established by this conception, it is convenient to distinguish three classes of problems, depending on their goals: to paraphrase Ravetz (1971), scientific problems (where the goal of the work is to establish new properties of the objects of inquiry, and the ultimate function is to achieve knowledge in its field); technical problems (those where the function to be performed specifies the problem); and practical problems (where the goal of the task is to serve or achieve some human purpose and the problem is brought into being by recognizing a problem situation in which some aspect of human welfare should be improved).

Against this background, practice can be recognized as the activity centered on practical problems, even while noting that to solve a practical problem often involves solving technical problems, and, when the basic phenomena underlying a problem situation are not understood, solving scientific problems in order to have the models needed for understanding the practical problem. It is also important to note that this view of science includes work on all three classes of problems within the conception of science as a whole. (For a more extended summary of Ravetz's view of science, see Miser and Quade 1988).

The Context of OR/MS

Since sciences are distinguished by their fields of inquiry, it is important to describe this context for OR/MS if it is to be differentiated from other sciences. In this endeavor the OR/MS community has not reached any sort of brief consensus, so what is said here must be regarded as a personal view, based in part on the literature and in part on personal experience.

While OR/MS deals with systems involving people, elements of nature, and machines (where this last term is intended to include not only artifacts but also laws, standard procedures, common behaviors, and social structures and customs), attempts to take the concept of system beyond this primitive statement as the basis for describing the context of OR/MS have, however, not proved fruitful.

The concept of an action program (Boothroyd 1978) is more useful: a function, operation, or response that is related to and given coherence by a human objective, need, or problem, together with the system of people, equipment, portion of nature, organizational elements, and management or social structure involved.

It is easy to see that an element in an action program may also have membership in other action programs; for example, an executive in one may also play a role in many others, as may also be the case for a major facility or organization, such as a large corporation or a government. Too, an action program may produce effects on other action programs, both through the cross memberships of elements and by the direct impacts of what it does. (For a more extended summary of Boothroyd's concept, see Miser and Quade 1988).

The practice of OR/MS can then be described as the activity that brings the knowledge and skills of the science of OR/MS to bear on the problems of action programs (Miser 1997). While this brief description will suffice as a basis for the argument here, the reader should be aware of the facts that, while it is quite general and covers most of what OR/MS does in practice now, it not only may not cover all of today's activities of practice but also may become even more incomplete with the passage of time.

The Situations of Practice

While each situation in practice may properly be seen as unique, it is nevertheless possible to describe one

that contains elements central to most – if not all – of practice, as follows.

An OR/MS analyst is often consulted when someone with a suitable responsibility in an action program discerns a problem situation that needs improvement. While this responsible person may have diagnosed the problem and even may have a notion about a possible solution, it is commonly the case that the forces actually yielding the source of dissatisfaction lie buried deeply enough to make such a diagnosis questionable, and the preconceived fix inappropriate. Thus, typically it is best for the analyst – or the team of analysts if the problem situation is complex – to approach it with an open mind, and aim to explore it thoroughly before deducing its properties and using them to devise a scheme for ameliorating its undesirable properties.

The analysts may be drawn from two sources:

1. There may be an analysis group inside the organization or action program with which the responsible problem-situation identifier – or client – is associated.
2. Analysts may have to be drawn from outside this organization or action program. In either case, there is abundant experience to support the conclusion that a successful outcome of the practice engagement calls for creating a constructive partnership between the analysis team and the parties at interest in the problem situation, as will be discussed in more detail later.

The Processes of Practice

Figure 1 offers a synoptic view of the elements that may be included in a practice engagement that proceeds from the general unease of a problem situation to the implementation of some policy or course of action and evaluates its effects. Since each situation has its own unique properties, few OR/MS practice engagements follow such a procedure exactly, but it is a common experience for many – if not most – of these elements to occur at some stage of the work.

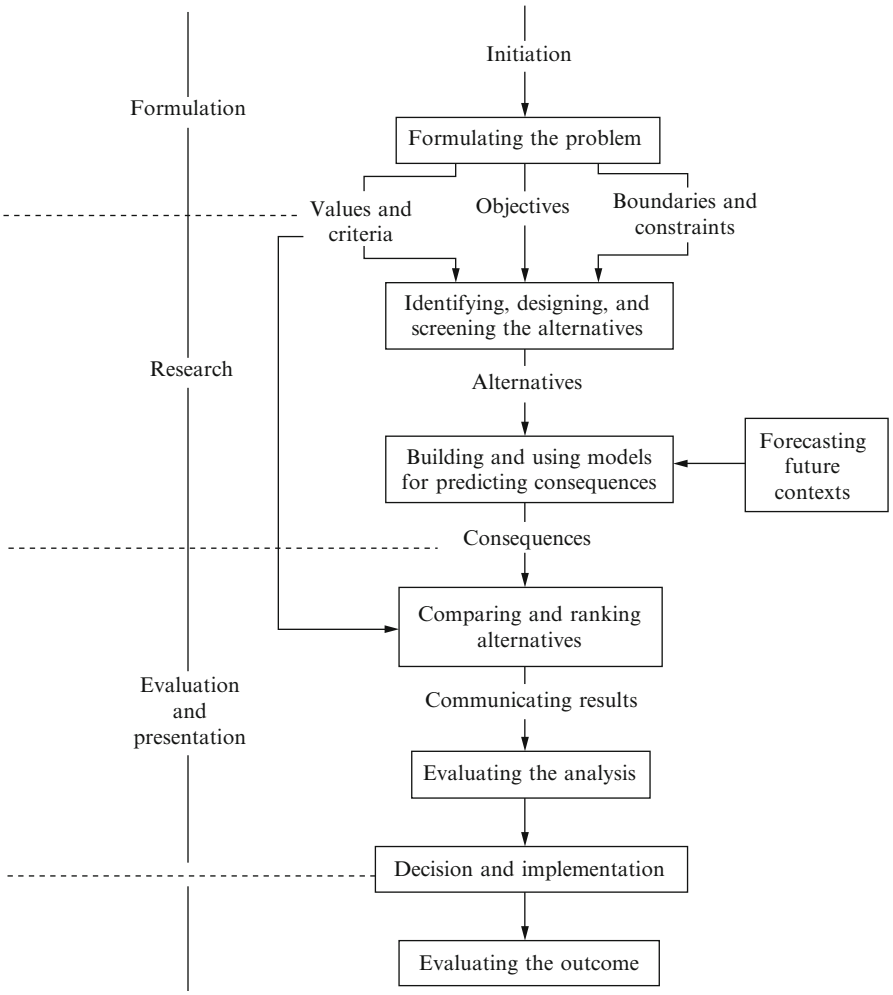
Formulation – The work begins with a thorough exploration of the problem situation in which the client and his/her action program cooperate. The purpose is to formulate the problem to be addressed, which commonly is quite different from the one originally conceived by the client. Once this is done, and the client has agreed with the analysis team on the problem, it is possible to plan the work to be done. This early work also identifies the values and criteria that should inform the choice of what eventually will be done to ameliorate the client's concerns, sets up the objectives to be sought by the solution, and agrees with the client on the boundaries and constraints that must be observed in devising it.

Usually this problem formulation step is one in which the analysts take the lead and work through it in informal cooperation with the client's staff. On occasion, however, it is best for a group consisting of both analysts and members of the client's staff to work together somewhat more formally toward a problem structure. To this end, there are various types of methods (Rosenhead 1996) that can be adapted to these situations. While the results of such a problem-structuring activity are usually a prelude to a more detailed analysis to follow, it sometimes happens that the insights from the group activity shared between the analysts and the client's staff are adequate to show what should be done to ameliorate the problem situation.

Research – This stage extends the information-and data-gathering that began in the formulation stage. The findings that emerge from processing these results allow the analysis team to identify, design, and screen possible alternatives that may help with the problem. Against this background, the analysis team can build models capable of deducing the consequences of adopting each of the alternatives chosen for further investigation within the contexts of possible future conditions.

Evaluation and Presentation – With estimates of the consequences in hand, the analysts may compare – and possibly rank – the alternatives against the criteria chosen earlier in the analysis, plus any new ones that may have emerged during the work. These findings must then be presented to the client and other parties at interest in a way that enables them not only to appreciate the results but also have at least a broad overview of the logic that produced them. These understandings may then enable the client to adopt a suitable policy or course of action.

Although the client, and not the analysts, must decide on what to do and how to carry it out effectively, experience shows that it is very important for the analysis team, or at least analysts who



Practice of Operations Research and Management Science, Fig. 1 Important elements in an OR/MS practice engagement that runs from problem formulation through research and implementation to evaluating the outcome (Source: Miser and Quade (1988), p. 23; reproduced by permission.)

understand and appreciate what was done, to work cooperatively throughout the implementation stage, as discussed later.

Variations

While it is possible to specify a core diagram of the principal elements of OR/MS practice, it must be admitted immediately that few, if any, such engagements follow this outline exactly. Rather, since each problem situation is different, the analysis activity must be adapted to it. Thus, in studying a series of cases, one sees variations like these:

- Instead of proceeding linearly from the top to the bottom of Fig. 1, the work cycles from intermediate stages back to earlier ones as the progress brings new insights and fresh intermediate results that may prompt reconsideration of the beginning foundations of the work.
- Some work may be aimed more at fleshing out the client's understanding of his situation than prompting him/her to change it significantly, so it may stop at one of the intermediate stages.
- The relative effort expended in the various stages may vary tremendously from case to case: one case may have to expend its major effort in just the information-and data-gathering stage, after which what needs to be done may be fairly apparent without much further analysis. Another case may proceed fairly expeditiously through the outline of Fig. 1 and then have a very long and complicated period of work to achieve what may appear to the outsider to be the implementation of a relatively simple set of proposals.

– In some cases an intermediate stage may dominate the work, owing to such factors as technical difficulty in devising proper models, major uncertainties in forecasting future contexts, complexities of the underlying situation, and so on.

In any case, the procedure specified here as the basis for discussion must be regarded as one that has stitched together the key elements that may enter OR/MS practice to varying extents depending on the peculiarities of the situation being studied.

The Importance of Following Through

The interest of the OR/MS professional, particularly if academically oriented, may flag after the research stage is completed and its results obtained. However, experience shows strongly that to stop there is almost always to waste the earlier effort. Two essential steps must follow: effective communication of the results, and cooperative aid in the implementation process.

Communication – This process, which may not be as appealing to the analyst as the research that preceded it, is nevertheless equally important and deserves great care, since communicating the findings inadequately can vitiate their potential effect, and thus waste the earlier effort. In view of the importance of this step in the OR/MS process, it is surprising that there is no systematic literature describing the skills needed and setting forth how they are best used (for a brief exception see Miser 1985). The discussion will be restricted to these points:

– Few clients will devote a large block of time to such communications, so it is very important to work very hard to condense the principal ideas and findings into as economical a space as possible, whether the form used is oral or written. For example, a top executive may want the key findings presented to him or her in a two-page memorandum or a 20-min briefing. It is perhaps surprising to the uninitiated to see how much important information can be condensed into so small a space, but only if great care is taken to make the best use of it. Graphs and charts accompanying the words can do much to aid this condensation.
– To communicate effectively, the client's vocabulary must be used, with as few technical terms introduced as possible.
– The whole must be focused on the interests of the client or the audience; after a major study many different groups may have to be addressed, and when this is the case the communication instruments must in each case be tailored to the group in view.
– The analysts must be prepared to stand behind their work and to discuss its implications, even those that may go beyond what was done as part of the analysis.

Implementation – It is clear that, if the findings of an OR/MS practice engagement do not find their way into some sort of changed reality, the work is ineffective. Therefore, it is obviously important for the analysis to consider the issue of eventual implementation throughout the work, keeping these points in mind:

1. Since the setting in which the work is being done has properties that will affect how change can be achieved, it is important for the peculiarities of this setting to be kept in mind from the beginning of the analysis. For example, can possible prospective changes be accommodated easily within the existing structure, or will it need to be changed significantly?
2. Since the settings in which OR/MS work is done are so various, it is impossible to stipulate a standard pattern for implementation work. This implies that the findings of the analysis may have to include a prospective implementation structure and program for the decision makers to consider as part of their judgment about the worth of the findings.
3. If the analysis considers different programs of action, the comparisons leading to a preferred choice should consider the relative difficulties of implementation as part of the analysis.
4. The history of analysis records that many well developed and clearly desirable program proposals failed to be implemented because the needed resources either did not exist or could not be made available. Therefore, in conceiving an implementation program as part of the findings of an OR/MS study, it is important to consider its resource requirements, as they will almost surely be an important issue to consider in whether or not to adopt the findings and translate them into action.

No matter how thoroughly the client – or members of his or her staff who participated in the analysis – understand what was found and its

prospective implementation, it is a common experience that the implementation process demands the continuing interest and cooperation of the analysis team, or at least some member of it who is able to follow through. The process of change invariably brings up new problems and issues that, wrongly handled, can vitiate the effects of what the original implementation set out to do. Too, these new problems may call for additional complementary analysis that must take account of what was done earlier.

This continuing involvement by analysts in the implementation process may take a variety of forms, ranging from occasional consultation to a continuing direct involvement of a substantial effort over such a long period of time as to make the implementation involvement a more ambitious enterprise than the original analysis (for an example illustrating this last point, see, Mechling 1995).

The roles of the analysts during implementation may include such activities as these:

1. Conducting supplementary analyses when situations arise calling for such work.
2. Helping all concerned keep the goals of the implementation program in sight. (It is all too easy for staff members involved, all of whom have personal in institutional goals in mind, to corrupt what is being done sufficiently that the original goals emerging from the analysis are vitiated.)
3. Proposing changes in the implementation strategy when they are called for by changing circumstance or the appearance of difficulties not foreseen in the beginning.
4. Acting as an on-site agent of persuasion when those directly involved in the implementation program need to have its goals clarified.

In sum, since an effective implementation phase is essential to the success of an OR/MS engagement, analysts should give it as much analytic and administrative importance and support as the analysis phase itself. For further elaboration of these points about implementation, see Tomlinson et al. (1985).

Outcome evaluation – It not infrequently happens that the outcomes of implementations are sufficiently clear to satisfy all concerned. Sometimes, however, in situations complex enough to make the outcomes unclear, it is necessary to conduct additional analysis to estimate the effectiveness of the implemented program or policy. The familiarity of the analysis team with the situation gives it an advantage in conducting such an analysis. However, to eliminate what may appear to be the original analysis team's bias in favor of a good outcome, clients may prefer to call in a new group to conduct such an outcome evaluation.

The Relation Between Analyst and Client

Emerging from a close scrutiny of the relations that should exist between analyst and client for effective cooperation, Schön (1983) advocates a "reflective contract" that works in this way: "... in a reflective contract between practitioner and client, the client does not agree to accept the practitioner's authority but to suspend disbelief in it. He agrees to join the practitioner in inquiring into the situation for which the client seeks help; to try to understand what he is experiencing and to make that understanding accessible to the practitioner; to confront the practitioner when he does not understand or agree; to test the practitioner's competence by observing his effectiveness and to make public his questions over what should be counted as effectiveness; to pay for services rendered and to appreciate competence demonstrated. The practitioner agrees to deliver competent performance to the limits of his capacity; to help the client understand the meaning of the professional's advice and the rationale for his actions, while at the same time he tries to learn the meanings his actions have for the client; and to reflect on his own tacit understanding when he needs to do so in order to play his part in fulfilling the contract."

Under this concept for OR/MS work, the client's obligation to share his experience and understanding of the problem situation is often discharged by assigning a member of his staff to work with the analysis team, an arrangement that has many benefits, among which these may be listed: it helps the analysis team identify and gather the information that it needs as a background and basis for its work; it helps the analysts avoid foolish mistakes related to the client's operations; and it acts to keep the client informed of what is emerging from the analysis, which often helps to pre-sell the findings that eventually merge.

Since OR/MS practice may be viewed as a dialogue between analyst and client related to the problem situation and the problem from it that is eventually

chosen for analysis, this arrangement serves as a useful continuing conduit for this dialogue, beyond what can be achieved with periodic progress meetings with the client (Miser 1994).

Other practical arrangements between the analysis team and the client to implement Schön's concept of a reflective contract must, of necessity, be evolved in the light of the circumstances peculiar to each engagement. An inhouse analysis group that has been able to achieve a reflective contract with the organization of which it is a part has a special opportunity: it can often identify problem situations that may not yet have been observed by executives in the organization, and thus set to work on them before they grow in size and importance.

How to Learn the Skills of Practice

The OR/MS community has, unfortunately, not evolved a comprehensive epistemology of practice and set it down in easily accessible literature that can be used widely in training courses. Some first steps in this direction for systems analysis, the large-scale efforts that can be thought of as part of OR/MS practice, are taken in Miser and Quade (1985,1988) and Miser (1995); much of what they say can apply equally to OR/MS as a whole. Thus, to learn the needed scientific and craft skills, someone aiming for an OR/MS career must pursue a tripartite program assembled from a variety of sources.

The intellectual basis – The foundation of effective OR/MS practice must be a thorough education in mathematics, with special attention to probability and statistics. Since by now certain models have become associated with OR/MS (as any introductory college textbook makes clear), these should be mastered as well. And a broad view of science with knowledge of other branches is also sure to be helpful.

Beyond a good mathematical and scientific education, however, the potential practitioner must not only be willing but also eager to learn from the problem situation, from the people in it, and from the representatives of other specialties, both practical and intellectual, that may have to be called on to help. As Schön's concept makes clear, to undertake an engagement in practice is to enter a multipartite partnership, and the flow of information must reflect this if the work is to be effective.

Since the action programs that OR/MS practice deals with contain people as essential elements, the analysts must know how to deal effectively and sympathetically with them, since they will enter the problem situation at many levels. In sum, interpersonal skills are an important requisite of good practice.

Familiarity with successful cases – There are by now a great many published accounts of successful cases of OR/MS practice. The journal *Interfaces* specializes in presenting them, and since 1975 has been a treasure-house of such accounts, as well as proven advice about the arts of practice. Assad et al. (1992) accompany a selection of these cases with valuable commentary. For a much wider view, one can consult the "Applications Oriented" section of the *International Abstracts in Operations Research*, the comprehensive abstract journal that has been published since 1961; it will not only exhibit the wide variety of practice being undertaken throughout the world but also identify the many journals and books in which cases appear. Rivett (1994) offers a broad introduction to successful practice based on a lifetime of varied experience.

Apprenticeship – Since the OR/MS community has yet to achieve a widely agreed and centrally documented view of its epistemology of practice, the best way for a person to observe and learn the myriad craft skills of practice is to work with an accomplished and skillful analysis team – in sum, to serve an apprenticeship (Miser and Quade 1985, 1988, offer a substantial body of additional information relating to the craft skills needed for effective OR/MS).

Examples of Good Practice

Since 1975, *Interfaces* has published the finalist papers in the Franz Edelman competition for the best papers on practice each year; there are five or more finalists in each competition. These accounts are an excellent central source of examples of good practice; in recent years tapes of the finalist presentations have also been made available.

There are many other sources of such work – too many to list here; however, both *Operations Research* and the *Journal of the Operational Research Society* contain one or more examples of good practice in each issue, as do the sources mentioned earlier.

See

- ▶ Decision Making and Decision Analysis
- ▶ Ethics in the Practice of Operations Research
- ▶ Field Analysis
- ▶ Implementation of OR/MS in the Public Sector
- ▶ Problem Structuring Methods
- ▶ Systems Analysis

References

Assad, A. A., Wasil, E. A., & Lilien, G. L. (1992). *Excellence in management science practice: A readings book*. New Jersey: Prentice Hall.

Boothroyd, H. (1978). *Articulate intervention*. London: Taylor and Francis.

Kemeny, J. G. (1959). *A philosopher looks at science*. New York: Van Nostrand Reinhold.

Mechling, J. E. (1995). Implementing innovative work schedules in the New York City sanitation department. In H. J. Miser (Ed.), *Handbook of systems analysis: Cases* (pp. 153–196). Chichester, UK: Wiley.

Miser, H. J. (1985). The practice of systems analysis. In H. J. Miser & E. S. Quade (Eds.), *Handbook of systems analysis: Overview of uses, procedures, applications, and practice* (pp. 287–326). Chichester, UK: Wiley.

Miser, H. J. (1993). A foundational concept of science appropriate for validation in operational research. *European Journal of Operational Research, 66*, 204–215.

Miser, H. J. (1994). Systems analysis as dialogue: An overview. *Technological Forecasting and Social Change, 45*, 299–306.

Miser, H. J. (Ed.). (1995). *Handbook of systems analysis: Cases*. Chichester, UK: Wiley.

Miser, H. J. (1997). The easy chair: Is it possible to have a good definitional description of operations research and management science? *Interfaces, 27*(6), 16–21.

Miser, H. J., & Quade, E. S. (Eds.). (1985). *Handbook of systems analysis: Overview of uses, procedures, applications, and practice*. Chichester, UK: Wiley.

Miser, H. J., & Quade, E. S. (Eds.). (1988). *Handbook of systems analysis: Craft issues and procedural choices*. Chichester, UK: Wiley.

Ravetz, J. R. (1971). *Scientific knowledge and its social problems*. Oxford: Oxford University Press (Reprinted 1996, New Brunswick, New Jersey: Transaction Publishers.).

Rivett, P. (1994). *The craft of decision modelling*. Chichester, UK: Wiley.

Rosenhead, J. (1996). What's the problem: An introduction to problem structuring methods. *Interfaces, 26*(6), 117–131.

Schön, D. H. (1983). *The reflective practitioner: How professionals think in action*. New York: Basic Books.

Tomlinson, R., Quade, E. S., & Miser, H. J. (1985). Implementation. In H. J. Miser & E. S. Quade (Eds.), *Handbook of systems analysis: Overview of uses, procedures, applications, and practice* (pp. 249–280). Chichester, UK: Wiley.

Precedence Diagramming

A graphic analysis of a project plan in which the nodes are the work activities (or tasks) and are connected by arrows. Relationships among tasks are designated as start-to-start, start-to-finish, and finish-to-finish, which eliminates the use of dummy arrows.

See

- ▶ Network Planning

Predictive Model

A model used to predict the future course of events and as an aid to decision making.

See

- ▶ Decision Problem
- ▶ Descriptive Model
- ▶ Mathematical Model
- ▶ Model
- ▶ Normative Model
- ▶ Prescriptive Model

Preemption

Concept having to do with how priorities are treated. In queueing theory, this means that an arriving higher priority customer pushes a lower one out of service because the newcomer has higher priority; service of the preempted customer later can either continue from the point of its interruption (preemptive resume queue discipline) or start totally anew. In goal programming problem, it is a statement that stipulates the ordering of the goals, so that a solution that satisfies the priority k goal is always to be preferred to solutions that satisfy the lower priority goals $k + 1$,....

See

- ▶ Goal Programming
- ▶ Queueing Theory

Preemptive Priorities

- ▶ Goal Programming
- ▶ Preemption
- ▶ Queueing Theory

Preference Theory

James S. Dyer[1] and Jianmin Jia[2]
[1]The University of Texas at Austin, Austin, TX, USA
[2]The Chinese University of Hong Kong, Shatin, NT, Hong Kong, China

Introduction

Preference theory studies the fundamental aspects of individual choice behavior, such as how to identify and quantify an individual's preferences over a set of alternatives and how to construct appropriate preference representation functions for decision making. An important feature of preference theory is that it is based on rigorous axioms which characterize individual's choice behavior. These preference axioms are essential for establishing preference representation functions, and provide the rationale for the quantitative analysis of preference. Preference theory provides the foundation for economics and the decision sciences. A basic topic of microeconomics is the study of consumer preferences and choices (Kreps 1990). In decision analysis and operations research, knowledge about the decision maker's preference is necessary to establish objective (or preference) functions that are used for evaluating alternatives. Different decision makers usually have different preference structures, which may imply different objective functions for them. Preference studies can also provide insights into complex decision situations and guidance for simplifying decision problems. The basic categories of preference studies can be divided into characterizations of preferences under conditions of certainty or risk and over alternatives described by a single attribute or by multiple attributes. This article begins with the introduction of basic preference relations and then discusses preference representation under certainty and under risk. A preference representation function under certainty will be referred to as a value function, where as a preference representation function under risk will be referred to as a utility function.

Basic Preference Relations

Preference theory is primarily concerned with properties of a binary preference relation $>_p$ on a choice set $\boldsymbol{X}$, where $\boldsymbol{X}$ could be a set of commodity bundles, decision alternatives, or monetary gambles. For example, an individual might be presented with a pair of alternatives, say x and y (e.g., two cars), and asked how they compare (e.g., do you prefer x or y?). If the individual says that x is preferred to y, then write $x >_p y$, where $>_p$ means strict preference. If the individual states that he or she is indifferent between x and y, then this preference is represented as $x \sim_p y$. Alternatively, define $\sim_p$ as the absence of strict preference, i.e., not $x >_p y$ and not $y >_p x$. If it is not the case that $y >_p x$, then write $x \geq_p y$, where $\geq_p$ represents a weak preference (or preference-indifference) relation. Also define $\geq_p$ as the union of strict preference $>_p$ and indifference $\sim_p$ i.e., both $x >_p y$ and $x \sim_p y$.

Preference studies begin with some basic assumptions (or axioms) of individual choice behavior. First, it seems reasonable to assume that an individual can state preference over a pair of alternatives without contradiction, i.e., the individual cannot strictly prefer x to y and y to x simultaneously. This leads to the following definition for preference asymmetry: preference is asymmetric if there is no pair x and y in $\boldsymbol{X}$ such that $x >_p y$ and $y >_p x$.

Asymmetry can be viewed as a criterion of preference consistency. Furthermore, if' an individual makes the judgment that x is preferred to y, then he or she should be able to place any other alternative z somewhere on the ordinal scale determined by the following: either better than y, or worse than x, or both. Formally, define negative transitivity by saying that preferences are negatively transitive if

given $x >_p y$ in $\boldsymbol{X}$ and any third element z in $\boldsymbol{X}$, it follows that either $x >_p z$ or $z >_p y$, or both.

If the preference relation $>_p$ is asymmetric and negatively transitive, then it is called a weak order. The weak order assumption implies some desirable properties of a preference ordering, and is a basic assumption in many preference studies. If the preference relation $>_p$ is a weak order, then the associated indifference and weak preference relationships are well behaved. The following results summarize some of these.

If strict preference $>_p$ is a weak order, then

1. strict preference $>_p$ is transitive (if $x >_p y$ and $y >_p z$, then $x >_p$ z);
2. indifference $\sim_p$ is transitive, reflexive ($x \sim_p x$ for all x), and symmetric ($x \sim_p y$ implies $y \sim_p x$);
3. exactly one of $x >_p y$, $y >_p x$, $x \sim_p y$ holds for each pair x and y; and
4. weak preference $\geq_p$ is transitive and complete (for a pair x and y, either $x \geq_p y$ or $y \geq_p x$).

Thus, an individual whose preferences can be represented by a weak order can rank all alternatives considered in a unique order. Further discussions of the properties of binary preference relations are presented in Fishburn (1970, Chapter 2) and Kreps (1990, Chapter 2).

Preference Representation Under Certainty

If strict preference $>_p$ on $\boldsymbol{X}$ is a weak order, then there exists a numeric representation of preference, a real-valued function v on $\boldsymbol{X}$ such that

$$x >_p y \text{ if and only if } v(x) > v(y),$$

for all x and y in $\boldsymbol{X}$ (Fishburn 1970). A preference representation function v under certainty is often called a value function (Keeney and Raiffa 1976). A value function is said to be order-preserving since the values $v(x)$, $v(y)$, ... ordered by $>$ are consistent with the preference order of x, y, ..., under $>_p$. Thus, any monotonic transformations of v will be order-preserving. As a result, the units of v have no particular meaning.

It may be desirable to consider a "strength of preference" notion that involves comparisons of preference differences between pairs of alternatives. To do so requires more restrictive preference assumptions, including that of a weak order over preferences between exchanges of pairs of alternatives (Krantz et al. 1971, Chapter 4). These axioms imply the existence of a real-valued function v on x such that, for all w, x, y, and z in $\boldsymbol{X}$, the difference in the strength of preference between w and x exceeds the difference between y and z if and only if

$$v(w) - v(x) > v(y) - v(z).$$

Furthermore, v is unique up to a positive linear transformation, i.e., if v' also satisfies the above difference inequality, then it must follow that $v'(x) = a\ v(x) + b$, where a (>0) and b are constants. This means that v provides an interval scale of measurement, such that v is often called a measurable value function to distinguish it from an order-preserving value function.

For multi-attribute decision problems, $\boldsymbol{X} = X_1, X_2 \ldots, X_n$, where n is the number of attributes and an element $x = (x_1, x_2, \ldots, x_n)$ in $\boldsymbol{X}$ represents an alternative. A multi-attribute value function can be written as $v(x_1, x_2, \ldots, x_n)$. Using some preference independence conditions, the multi-attribute value model can be simplified.

The subset $\boldsymbol{Y}$ of attributes in $\boldsymbol{X}$ is said to be preferentially independent of its complementary set $\bar{Y}$ if preferences for levels of these attributes $\boldsymbol{Y}$ do not depend on the fixed levels of the complementary attributes $\bar{Y}$. Attributes $X_1, X_2, \ldots, X_n$, are mutually preferentially independent if every subset of these attributes is preferentially independent of its complementary set.

A multi-attribute value function $v(x_1, x_2, \ldots, x_n)$ $n \geq 3$, has the following additive form

$$v(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} v_i(x_i), \tag{1}$$

where v_i is a value function over X_i if and only if the attributes are mutually preferentially independent (Keeney and Raiffa 1976; Krantz et al. 1971). When v is bounded, it may be more convenient to scale V such that each of the single-attribute value functions ranges from zero to one, leading to the following form of the additive value function:

$$v(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} w_i v_i(x_i), \tag{2}$$

where v and v_i are scaled from zero to one, and the w_i are positive scaling constants (usually called weights) summing to one. The assessment of models (1) and (2) are discussed in Keeney and Raiffa (1976, Chapter 3).

Dyer and Sarin (1979) proposed multi-attribute measurable value functions based on the concept of preference differences between alternatives that are much easier to assess than the additive form based on preferential independence. In addition to preferential independence, they considered some additional conditions that, loosely speaking, require that the decision maker's comparisons of preference differences between pairs of alternatives that differ in the levels of only a subset of the attributes do not depend on the fixed levels of the other attributes. These conditions allow the decomposition of a multi-attribute value model into additive and multiplicative forms. This development also provides a link between the additive value function and the multi-attribute utility model.

Preference Representation Under Risk

Perhaps the most significant contribution to the area of preference representation for risky options (i.e., lotteries or gambles) was the formalization of expected utility theory by von Neumann and Morgenstern (1947). This development has been refined by a number of researchers and is most commonly presented in terms of three basic axioms (Fishburn 1970).

Let P be a convex set of simple probability distributions or lotteries $\{X, Y, Z, \ldots\}$ on a nonempty set X of outcomes. (X, Y and Z will be used to refer to probability distributions and random variables interchangeably.) For lotteries X, Y, Z in P and all λ, $0 < \lambda < 1$, the expected utility axioms are:

A1. (*Ordering*) $>_p$ is a weak order;

A2. (*Independence*) If $X >_p Y$, then $\lambda X + (1-\lambda) Z >_p \lambda Y + (1-\lambda)Z$ for all Z in P;

A3. (*Continuity*) If $X >_p Y >_p Z$, then there exist some $0 < \alpha < 1$ and $0 < \beta < 1$ such that $\alpha X + (1-\alpha)Z >_p Y >_p \beta X + (1-\beta)Z$.

The von Neumann-Morgenstern expected utility theory asserts that the above axioms hold if and only if there exists a real-valued function u such that for all X, Y in P,

$$X >_p Y, \text{ if and only if } \mathrm{E[u(X)]} > \mathrm{E[u(Y)]},$$

where the expectation is taken over the probability distribution of a lottery. Moreover, such a u is unique up to a positive linear transformation.

The expected utility model can also be used to characterize an individual's risk attitude (Keeney and Raiffa 1976, Chapter 4). If an individual's utility function is concave, linear, or convex, then the individual is risk averse, risk neutral, or risk seeking, respectively. The von Neumann-Morgenstern theory of risky choice presumes that the probabilities of the outcomes of lotteries are provided to the decision maker. Savage (1954) extended the theory of risk choice to allow for the simultaneous development of subjective probabilities for outcomes and for a utility function u defined over those outcomes.

As a normative theory, the expected utility model has played a major role in the prescriptive analysis of decision problems. However, for descriptive purposes, the assumptions of this theory have been challenged by empirical studies (Kahneman and Tversky 1979). Some of these empirical studies demonstrate that subjects may choose alternatives that imply a violation of the independence axiom (A2). Prospect theory (Kahneman and Tversky 1979; Wakker 2010) attempts to explain these discrepancies. One implication of A2 is that the expected utility model is linear in probabilities. A number of contributions have been made by relaxing the independence axiom and developing some nonlinear utility models to accommodate actual decision behavior (Fishburn 1988).

For the case of multi-attribute decisions under risk, when $X = X_1 \times X_2 \times \ldots \times X_n$ in a von Neumann-Morgenstern utility model and the decision maker's preferences are consistent with some additional independence conditions, then $u(x_1, x_2, \ldots, x_n)$, can be decomposed into additive, multiplicative, and other well-structured forms that simplify assessment.

The attributes $X_1, X_2, \ldots, X_n$ are said to be additive independent if preferences over lotteries on $X_1, X_2, \ldots, X_n$ depend only on the marginal probabilities assigned to individual attribute levels, but not on the joint probabilities assigned to two or more attribute levels.

A multi-attribute utility function $u(x_1, x_2, \ldots, x_n)$, can be decomposed as

$$u(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} w_i u_i(x_i), \tag{3}$$

if and only if the additive independence condition holds, where u_i is a single-attribute function over X_i scaled from 0 to 1, and the w_i are positive scaling constants (or weights) summing to one. The additive model (3) has been widely used in practice.

If the decision maker's preferences are not consistent with the additive independence condition, a weaker independence condition that leads to a multiplicative preference representation may be satisfied.

An attribute X_i is said to be utility independent of its complementary attributes if preferences over lotteries with different levels of X_i do not depend on the fixed levels of the remaining attributes. Attributes $X_1, X_2, \ldots, X_n$ are mutually utility independent if all proper subsets of these attributes are utility independent of their complementary subsets.

A multi-attribute utility function $u(x_1, x_2, \ldots, x_n)$ can have the multiplicative form

$$1 + ku(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} [1 + kk_i u_i(x_i)], \qquad (4)$$

if and only if the attributes $X_1, X_2, \ldots, X_n$ are mutually utility independent, where u_i is a single-attribute function over X_i scaled from 0 to 1, the k_i are positive scaling constants, and k is an additional scaling constant. For approaches to the assessment of model (4) and other extensions of multi-attribute utility theory, see Keeney and Raiffa (1976).

The research of multi-attribute utility theory has been advanced from both theoretical and behavioral considerations. In particular, the effort of behavioral research tries to improve the descriptive power of multi-attribute utility models by incorporating psychological factors, such as aspiration level, goal and reference effect, and loss aversion (Tversky and Kahneman 1991). Various decision support systems have also been developed for multi-attribute decision making in the past decades, and applications of the theory and models have been expended to many new areas, including e-commence, public policy and environmental decisions, geographic information systems, and engineering (Dyer et al. 1992; Wallenius et al. 2008).

See

- ▶ Choice Theory
- ▶ Decision Analysis
- ▶ Multi-attribute Utility Theory
- ▶ Prospect Theory
- ▶ Utility Theory

References

Dyer, J. S., Fishburn, P. C., Steuer, R. E., Wallenius, J., & Zionts, S. (1992). Multiple criteria decision making, multiattribute utility theory: The next ten years. *Management Science, 38*, 645–654.

Dyer, J. S., & Sarin, R. K. (1979). Measurable multi-attribute value functions. *Operations Research, 27*, 810–822.

Fishburn, P. C. (1970). *Utility theory for decision making*. New York: Wiley.

Fishburn, P. C. (1988). *Nonlinear preference and utility theory*. Baltimore, MD: The Johns Hopkins University Press.

Kahneman, D. H., & Tversky, A. (1979). Prospect theory: An analysis of decision under risk. *Econometrica, 47*, 263–290.

Keeney, R. L., & Raiffa, H. (1976). *Decisions with multiple objectives: Preferences and value tradeoffs*. New York: Wiley.

Krantz, D. H., Luce, R. D., Suppes, P., & Tversky, A. (1971). *Foundation of measurement*. San Diego, CA: Academic.

Kreps, D. M. (1990). *A course in microeconomics theory*. Princeton, NJ: Princeton University Press.

Savage, L. J. (1954). *The foundations of statistics*. New York: Wiley.

Tversky, A., & Kahneman, D. H. (1991). Loss aversion in riskless choice: A reference-dependent model. *Quarterly Journal of Economics, 106*, 1039–1061.

von Neumann, J., & Morgenstern, O. (1947). *Theory of games and economic behavior*. Princeton, NJ: Princeton University Press.

Wakker, P. P. (2010). *Prospect theory: For risk and ambiguity*. New York: Cambridge University Press.

Wallenius, J., Fishburn, P. C., Zionts, S., Dyer, J. S., Steuer, R. E., & Deb, K. (2008). Multiple criteria decision making, multiattribute utility theory: Recent accomplishments and what lies ahead. *Management Science, 54*, 1336–1349.

Prescriptive Model

A model that attempts to describe the best or optimal solution of a man/machine system. For a decision problem, such a model is used as an aid in selecting the best alternative solution.

See

- ▶ Decision Problem
- ▶ Descriptive Model
- ▶ Mathematical Model
- ▶ Normative Model

Prices

In the simplex method, for a nonbasic variable x_j, the price is defined as $d_j = c_j - z_j$ or $d_j = z_j - c_j$, where c_j is the variable's original cost coefficient and $z_j = \boldsymbol{\pi}\mathbf{A}_j$, with $\mathbf{A}_j$ the variable's original column of coefficients and $\boldsymbol{\pi}$ the multiplier (pricing) vector of the current basis. The d_j is termed the reduced or relative cost. It is the difference between the direct cost c_j and indirect cost z_j. The d_j indicates how much the objective function would change per unit change in the value of x_j. The d_j for the variables in the basic feasible solution are equal to zero.

See

- ▶ Devex Pricing
- ▶ Opportunity Cost
- ▶ Simplex Method (Algorithm)

Pricing Multipliers

- ▶ Multiplier Vector

Pricing Out

In the simplex method, the calculation of the prices associated with the current basic solution.

See

- ▶ Prices
- ▶ Simplex Method (Algorithm)

Pricing Vector

- ▶ Multiplier Vector
- ▶ Prices
- ▶ Simplex Method (Algorithm)

Prim's Algorithm

A procedure for finding a minimum spanning tree in a network. The method starts from any node and connects it to the node nearest to it. Then, for those nodes that are now connected, the unconnected node that is closest to one of the nodes in the connected set is found and connected to these closest nodes. The process continues until all nodes are connected. Ties are broken arbitrarily.

See

- ▶ Greedy Algorithm
- ▶ Kruskal's Algorithm
- ▶ Minimum Spanning Tree Problem

Primal Problem

The primal problem is usually taken to be the original linear-programming problem under investigation.

See

- ▶ Dual Linear-Programming Problem

Primal-Dual Algorithm

An adaptation of the simplex method that starts with a solution to the dual problem and systematically solves a restricted portion of the primal problem while improving the solution to the dual. At each step, a new restricted primal is defined and the process continues until solutions to the original primal and dual problems are obtained.

See

- ▶ Simplex Method (Algorithm)

Primal-Dual Linear-Programming Problems

- ▶ Dual Linear-Programming Problem
- ▶ Linear Programming

Principle of Optimality

Condition that Richard Bellman derived for dynamic programming: "An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision." (Bellman 1957, Chap. III.3)

See

- ▶ Bellman Optimality Equation
- ▶ Dynamic Programming

References

Bellman, R. E. (1957). *Dynamic programming*. Princeton, NJ: Princeton University Press.

Prisoner's Dilemma

A two-person game where neither player knows the other's play (action or decision) a priori. Imagine a situation where two criminals are isolated from each other and the police interrogator offers each the following deal: if the prisoner confesses and the confession leads to the conviction of the other prisoner, he goes free and the other prisoner gets 10 years in prison. However, if both confess, they each get 5 years. If neither confesses, there is enough evidence to convict both on a lesser offense and they both get one year. If there is no trust, then both will confess, whereas if there is complete trust, neither will. Since complete trust is rare, when the game is played one time, players almost always defect. When the game is played repeatedly and there is a chance for a long-term reward, wary cooperation with a willingness to punish defection is the best strategy. This game illustrates many social and business contracts and is important for understanding group behavior, both cheating and cooperation. It has also been used in studying political and military strategies.

See

- ▶ Game Theory

References

Poundstone, W. (1992). *Prisoner's dilemma*. New York: Doubleday.

Probabilistic Algorithm

An algorithm that employs probabilistic elements (as opposed to a deterministic algorithm).

See

- ▶ Genetic Algorithms
- ▶ Randomized Algorithm

Probabilistic Programming

A mathematical programming problem in which some or all of the data are random variables.

See

- ▶ Chance-Constrained Programming
- ▶ Stochastic Programming

Probability Density Function (PDF)

When the derivative $f(x)$ of a cumulative probability distribution function $F(x)$ exists, it is called the density or probability density function.

Probability Distribution

Term used (loosely) to refer to a function describing the probabilistic behavior of a random variable; could refer to the probability measure, the cumulative distribution function (CDF), the probability mass function (PMF) for discrete random variables, or the probability density function (PDF) for continuous-valued random variables.

Probability Generating Function

For a non-negative integer-valued random variable X with probability mass function $p_j = \Pr\{X = j\}$, the probability generating function (often just called the generating function, and known in other fields as the z-transform) is given by $P(z) = E[z^X] = \sum_{j=0}^{\infty} z^j p_j$. The definition can be extended to the setting where X can take all integer values (i.e., including all negative values).

Probability Integral Transformation Method

One of the primary methods for generating random variates for Monte Carlo or discrete-event simulation, using the cumulative distribution function (CDF) commonly known as the inverse transform method.

See

- ▶ Inverse Transform Method
- ▶ Random Number Generators
- ▶ Random Variates
- ▶ Simulation of Stochastic Discrete-Event Systems

Probability Mass Function (PMF)

Function giving the probability of taking on each of the possible discrete values.

Problem Solving

The process of deciding on actions aimed at achieving a goal. Initially, the goal is defined to represent a solution to a problem. During the reasoning process, subgoals are formed, and problem solving becomes recursive.

See

- ▶ Artificial Intelligence
- ▶ Decision Analysis
- ▶ Decision Making and Decision Analysis
- ▶ Decision Support Systems (DSS)
- ▶ Expert Systems

Problem Structuring Methods

Jonathan Rosenhead
The London School of Economics and Political Science, London, UK

Introduction

Problem structuring methods (PSMs) are a broad group of model-based problem handling approaches whose purpose is to assist in the structuring of problems rather than directly to derive a solution. They are participative and interactive in character, and normally operate with groups rather than individual clients. In principle they offer OR/MS access to a range of problem situations for which more classical OR techniques have limited applicability. The most widely adopted of these methods are Soft Systems Methodology, the Strategic Choice Approach, and Strategic Options Development and Analysis (SODA).

PSMs developed out of, or at least intertwined with, a critique of the restricted scope of traditional OR techniques. From the 1970s there developed an active debate over claims for the objectivity of OR/MS models, and about the limitations imposed on OR/MS practice by its concentration on well-defined problems. Significant critical contributions were made by

Rittel and Webber (1973), Ackoff (1979), Checkland (1981), Rosenhead and Thunhurst (1982), Eden (1982), Rosenhead (1986), Jackson (1987), Flood and Jackson (1991), Mingers (1992). The general thrust was that standard OR techniques assume that relevant factors, constraints, and objective function are both established in advance and consensual; commonly the function of the technique is to determine an optimal setting of the controllable variables. Consistently with this, standard formulations of OR methodology were seen to assume a single uncontested representation of the problematic situation under consideration.

Critics have recognized that OR's practice has been considerably more diverse than this, and in particular is far from dominated by considerations of optimality; however, the available tools were held to offer little appropriate assistance outside this area. The methodological framework on offer was equally seen as giving scant guidance to analysts confronting less well-behaved circumstances. There are situations in which intangibles, uncertainty, and value diversity as well as complexity are crucial presences. Skilled operational researchers have been able to make progress in such situations, but only by using tacit skills which are not part of the OR/MS canon. Yet the more socially important the decision situation, the more likely it is that such features will come to dominate.

Out of this critique of the shortcomings of traditional OR/MS a family of alternative methods was developed, with both common features and also differences of focus. When their similarities were recognized the label used to describe them as a group was Problem Structuring Methods (though other names such as Soft OR are also in currency). The over-arching emphasis which the methods share is on helping groups of decision-makers to identify what problem they could usefully work on together, and to assist them in making progress with that task. There is no assumption that the decision-makers share a common perspective, so that they are perhaps more accurately described as stakeholders. Nor are these methods to any significant degree quantitative. This is because the approaches are all based on the participation of those who have the problem. If mathematics were to be the language of the discourse, some (perhaps many) of the participants would be disempowered, or at least prevented from enunciating perceptions important to them which could not be expressed in that format, or only by a distortion which changed their content.

Each of the methods within the PSM family consists of a number of technical procedures linked together through social processes. i.e., unlike the algorithmic approaches that have tended to dominate OR/MS, the consultant does not identify and then input some starting conditions from which the 'answer' will be produced without further human intervention. What happens is that at various points the groups discuss the implications of the analysis to date, and on that basis (and aided by a facilitator) decide how to proceed further, or maybe whether enough progress has been made that the stakeholders can proceed without further analytic assistance. For clarity one should perhaps describe PSMs as 'methodologies' rather than 'methods', taking a methodology as an assembly of technical and process elements.

In short these methods bear very little resemblance to those developed within traditional OR/MS. The one key unifying element is the central use of cause-effect models. Each of them uses formal models to represent the problematic situation perceived by the decision-making group, in order to summarise, coordinate and advance their understanding of the situation they confront. The types of model used are specific to each method, but none of them are 'computable'. Indeed quantification has little if any role in any of them. The concepts that are in play are more usually verbal, and the operations on them are mostly performed by the group, who through discussion transform the models based on their changing understanding. The outcomes of a successful application of a PSM will be a group of decision-makers confident enough to take action; a group of decision-makers who have gained a deeper insight into their problem area; and a group of decision-makers whose shared experience has led to improved relations with each other.

Types of Problem

Before going on to outline the PSM field, it should be helpful to address the apparent paradox of two very different types of methodology, sometimes called 'hard' (i.e., traditional) and 'soft' (PSMs), each addressing problems of complexity in an analytic

manner. One simple explanation lies in the quite wide recognition of two substantially different types of problem situation. Rittel and Webber's (1973) characterization of them as tame vs. wicked has achieved wide currency, as has Schon's (1987) extended metaphor of problems of the swamp contrasted with those of the high ground. Tame problems (on the high ground) have precise, unproblematic formulations permitting powerful analyses of great technical sophistication. Wicked problems (in the swamp) have multiple stakeholders, intangible objectives, key uncertainties, contested or doubtful formulations, etc. In the latter there is no unified representation of the issue or issues that can be established *ahead* of analysis. Rather, a representation or representations of the problematic situation which participants find helpful may be a major product of the analysis.

It follows from this diagnosis that methods that are designed to be effective in handling tame problems are likely to be largely irrelevant for wicked ones. (And vice versa of course.) For the latter type of problem situation, methods that assist argumentation, promote negotiation or generate mutual understanding are needed, rather than those that reliably and efficiently identify an optimum. Methods that can only start once there is an agreed problem (but have no methods for reaching that agreement) are liable to ignore or dismiss alternative perspectives and their contrary formulations.

The much remarked difficulty which OR/MS encountered from the late 1960s in securing access to more strategic levels of decision-making may be attributed at least in part to this factor. As Schon observed, problems of major social importance are commonly located in "swamp" conditions. Attempts to address these "messes" using techniques and methodology developed for handling well-structured problems constituted inappropriate technology transfer. Where solutions based on these methods were adopted they were vulnerable to being savaged in practice by the 'wicked' parts of the problem situation that had been excluded. More commonly however such representations were recognised as an overly thin representation of the rich and complex world that managers and decision-makers inhabit – with the result that OR/MS was confined to the tame (less strategic, more repetitive, operational) aspects of organisational life.

Characteristics of Alternative Methods

Problem structuring methods constitute a family of approaches offering appropriate support to decision-making under these less pacified circumstances. They were developed separately by individual innovators or teams of innovators, and each emphasizes or is organized around particular aspects of the wicked problem environment. Indeed each had been independently developed before a recognition arose of their family resemblance. (Subsequent to that recognition, however, many of the principal originators entered into a constructive dialogue with each other, in which a certain amount of mutual borrowing of particular elements took place.: For example, distinctive post-it 'Ovals', originated for use within SODA, became widely used by other methods.) In other words the new methods grew out of practice. However their similarities are by no means coincidental. Many leading developers of PSMs had been active participants in the critique of traditional methods, and their innovations were designed to remedy particular inadequacies of the conventional repertoire in handling wicked problems. So at that fundamental level there was a common theoretical base.

PSM methods have differing rationales, purposes, technical apparatus, etc. Some of these distinctive attributes will be indicated below. However it will be useful, first, to identify the features which they hold in common.

Rosenhead (1989) has provided one formulation, based on inverting the characteristics of the conventional OR/MS paradigm.

PSMs

- Seek solutions which satisfice on separate dimensions (rather than trade-off onto a single dimension to facilitate optimization);
- Integrate hard and soft data with social judgments (reducing data greed with its problems of quality and distortion);
- Produce transparent models which clarify any conflicts (rather than basing a scientific depoliticization on an assumed consensus);
- Treat people as subjects actively engaged in the decision-making process (rather than as passive objects to be modelled or disregarded);
- Facilitate planning from the bottom-up (and not as a process driven by the abstract objectives of a hierarchically located decision-maker); and

- Accept that some uncertainty is irreducible and aim to preserve options (rather than base current and future decisions on a notionally certain future).

The methods clearly assume a decision-making quite different from that of conventional OR/MS applications, and this environment places particular requirements on the interface with the client group. Where consensual values cannot be assumed, there will be a need to achieve agreement among a range of stakeholders representing different interests and/or holding different perspectives. It follows from this that a PSM should be able to accommodate multiple alternative perspectives, often in a group situation in which holders of those viewpoints are present and participating. From this it follows that for a method to be helpful it must operate iteratively and interactively; as participants internalize and adjust to each others' contributions, new formulations of the problematic situation will emerge which in turn feed new modelling and structuring activity. And since participants have different though overlapping organizational agendas, and also because of the prevalence of uncertainty, any resulting consensus on action is likely to constitute a partial rather than a comprehensive solution to the problems present within the situation under discussion.

These social requirements on a PSM have implications for the technical repertoire that it can deploy. Its handling of complexity must not obstruct lay participation — which points to graphical (rather than, for example, algebraic) representations. The existence of multiple perspectives invalidates the search for an optimum; the need is rather for systematic exploration of the solution space. To elicit meaningful judgments from lay participants, abstract continuous variables need to be eschewed in favour of discrete concrete alternatives that can be compared. And, given the need to avoid illusions of precision when confronting uncertainties, possibilities will be more helpful than probabilities, and alternative scenarios will enrich discussion that forecasts might close down.

These outline specifications for a more appropriate decision-aiding technology eliminate much of the scope for advanced mathematics, probability theory, complex algorithms. They identify, rather, an alternative approach employing representation of relationships, symbolic manipulation, and limited quantification within a systematic framework. These are decidedly low-tech methods: some of them have no software support, and even those that do can be operated in manual mode. The lack of mathematics should not however be taken for lack of rigour. These are methods with their own rigour, which is qualitative in nature.

The Methods

There is no definitive list of problem structuring methods. However to give identity to the field it is appropriate to provide some demarcation criteria.

PSMs

- Can be distinguished from traditional OR methods by the six criteria listed in the previous section.
- Can be distinguished from non-OR modes of working with groups, such as Organizational Development, by the core element of an explicit modelling of cause-effect relationships.
- Can be demarcated from other OR approaches which purport to tackle messy, ambitious problems (e.g., the Analytic Hierarchy Process) by PSMs' transparency of method, restricted mathematization, and focus on supporting judgment rather than representing it.

These limits are imprecise and arguable; and there is scope for approaches developed for other or broader purposes (e.g., spreadsheet models) to be used in a similar spirit. Ackoff's Interactive Planning is close in both spirit and intent (see Ackoff 1999) but nevertheless has never been regarded as falling within PSMs. (Rather than changing this de facto if not de jure circumstance, it will not be discussed further.) Methods that have some degree of similarity to PSMs but also significant differences are (for coherence) best regarded as falling outside the category. These include multi-criteria decision methods, outranking methods such as PROMETHEE and ELECTRE, decision conferencing, scenario planning, system dynamics (in some of its versions) and Viable System Diagnosis. Other parts of the PSM perimeter are bordered by the focus group approach, and by Rapid Rural Appraisal and other participative third world development approaches (for which see Rosenhead and Mingers 2001, pp. 345-7).

A brief introduction to the better established PSMs follows (Rosenhead and Mingers 2001):

Strategic Options Development and Analysis (SODA)

This method is described fully in Eden and Ackermann (1998). It is a general purpose problem identification method that uses cognitive mapping as a modelling device. The concepts that individuals use to make sense of their problematic situation, and the causal links thought to exist between those concepts, are elicited in individual interviews and recorded in map form. The maps drawn from separate interviews with stakeholders are subsequently merged into a single 'strategic map' through pinning together concepts common to more than one of them. The strategic map, commonly structured into clusters, provides the framework for discussion in a workshop of the group of map 'owners', at which a facilitator uses the map to guide participants towards commitment to a portfolio of actions. An alternative and more rapid version known as the Oval Mapping Technique operates in workshop mode throughout, and can in principle achieve results in a 1 day session. The participants commit their concepts to 'Ovals' (specially designed PostIt notes), which the facilitator with the participation of workshop members organises into an agreed structure. This then serves as the strategic map for the discussion that follows.

Soft Systems Methodology (SSM)

Soft Systems Methodology is a general method for system design or redesign, which aims to generate debate about alternative system modifications. It adopts a systems theoretic framework for exploring the nature of problem situations, and how purposeful action to change them might be agreed when there are different perceptions of the situation based on contrasting world views. A systematic exploration of the world views of stakeholders leads to the generation of definitions of alternative systems, the investigation of which is expected to be of interest from at least one of those world views. Each of these abstract 'root definitions' is expanded into the component activities which would be necessary for it to operate successfully. This generates a range of contrasting alternatives for the modification of the system, which are used to generate debate about which changes are both culturally feasible and systemically desirable. Full descriptions of the method are available in Checkland (1981, 2006, 1990).

Strategic Choice Approach (SCA)

Strategic Choice is a planning approach centred on the management of uncertainty and commitment in strategic situations. Typically a Strategic Choice engagement takes place entirely in workshop format, with no backroom work by the consultants. There are four modes of analysis:

- Shaping – in which different areas for choice are elicited from workshop members. A subset of these is selected as a problem focus by reference to their urgency, importance and inter-connectedness
- Designing – here the options for action for each of the decision areas within the problem focus are identified, as well as any incompatibilities between option selections in different decision areas. The feasible decision schemes (consisting of one option choice within each decision area) are derived
- Comparing – criteria for choice, often non-quantitative, are agreed by the group. These are used first in satisficing mode to establish a working shortlist of schemes; pairwise comparisons of shortlisted schemes are made, establishing on each criterion a range of relative advantage between the two schemes. This may be repeated for different pairs. Commonly significant uncertainties are revealed by this process. Other uncertainties will usually have been identified in previous modes
- Choosing – bearing in mind the surfaced uncertainties, a 'progress package' is agreed consisting of partial commitments to be made at this stage, explorations to be launched to reduce key uncertainties, contingency plans, and a timetable for later choices.

Facilitators assist with the deployment of the transparent tools available within the method, and in guiding the, possibly recursive, switching between modes. A detailed account of the method is available in Friend and Hickling (2004).

Robustness Analysis

Robustness Analysis is another approach for use where uncertainty is an important issue. It focuses on one specific strategy for managing that uncertainty - that of maintaining useful flexibility. The focus of the approach is on initial commitments rather than on future plans for the system. The flexibility of an initial commitment relates to its compatibility with a range of

acceptable or desirable future states of the system. It is this flexibility left by an initial commitment that is operationalised as a decision-making criterion by the concept of the robustness. This is defined as a ratio where the denominator is the number of states whose performance at the planning horizon is 'good enough'; the numerator is the number of those states which would remain accessible if the commitment under consideration were to be made. Robustness analysis can be conducted with either a single or multiple futures employed to estimate system performance; and it can be used in conventional or interactive mode. In the latter, participants and analysts assess both the compatibility of initial commitments with possible future configurations of the system, and the performance of each configuration in feasible future environments. This enables them to compare the flexibility maintained by alternative initial commitments. It is in this latter mode that Robustness Analysis qualifies as a PSM, though even when used in non-participatory mode it maintains an accessible transparency. For more detail, see Rosenhead and Mingers (2001).

Drama theory

Drama Theory draws on two earlier approaches, metagames and hypergames. It is an interactive method of analysing co-operation and conflict among multiple actors. A model is built from perceptions of the options available to the various actors, and how they are rated. Drama theory looks for the 'dilemmas' presented to the actors within this model of the situation. Each dilemma is a change point, tending to cause an actor to feel specific emotions and to produce rational arguments by which the model itself is redefined. When and only when such successive redefinitions have eliminated all dilemmas is the actors' joint problem fully resolved. Analysts commonly work with one of the parties, helping it to be more effective in the rational-emotional process of dramatic resolution. For more detail, see Howard (1999).

Applications of PSMs

As can be inferred from their remit to structure wicked problems, the problem situations to which PSMs have been applied have a wide variety. A good source for practical applications of the SCA is Chapter 13 of Friend and Hickling 2004, pp. 298-360. An overview of applications across the range of PSMs is provided by Mingers and Rosenhead (2004), which is the review article for a special issue of the *European Journal of Operational Research* on applications of PSMs (Vidal 2004).

A diverse record of successful applications is an indicator of wide relevance, but a disadvantage when it comes to providing a coherent summary. A literature survey covering the period up to 1998 (summarized in Mingers and Rosenhead 2004) categorises 51 reported applications under the headings general organizational/information systems/technology, resources, planning/health services/general research. Two comments seem appropriate: (i) it is plausible to assume that reported cases are the tip of the iceberg; and (ii) 1998 was relatively early in the development of interest in PSMs.

The categories supplied in the previous paragraph are so broad as to give little flavour of the reality of PSM practice. To provide that, some short summaries of projects using PSMs that are described in Mingers and Rosenhead (2004) may be of assistance

- Organisational restructuring at Shell. SSM used to provide the basis of a reconfiguration of a central department of Shell International, in a series of workshops with senior managers
- Models to support a claim for damages. SODA (as well as System Dynamics) used to support a legal case by the Canadian-based multinational Bombardier against Trans Manche Link, for damages resulting from delays in processing designs for the Channel Tunnel shuttle wagons
- Supporting a tenants cooperative. This was an engagement over several years to help a cooperative of residents of an ex-mining village to manage their own housing. Elements of various PSMs, as well as other methods (e.g., spreadsheet financial models) were used to support strategic decisions, and help the cooperative gain confidence
- IT strategy for a supermarket chain. This study reported to the joint chief executives of the leading British supermarket chain Sainsbury's, and worked with a 16-strong senior management task force. SODA, SSM and SCA were all used at different stages, to identify IT systems that would support business objectives

- Planning for a street festival. The largest European street festival (Notting Hill Carnival) was a victim of its own success, with issues of security, congestion, cultural integrity etc. Working with representatives of the carnivalists, local government, transport and emergency services, and arts organisations, SSM and SCA were used to devise escape strategies
- National level planning in Venezuela. A version of SCA has been used at various levels of the state service in Venezuela, up to and including the Cabinet, to agree on strategic decisions in a range of areas
- Local pediatric care strategy. Health care managers and specialists in an Inner London area with some 500,000 population needed to reduce the number of inpatient paediatric care units. SCA was used in a series of workshops to produce agreement between representatives of all stakeholders on (i) how many units should remain (ii) where they should be; and (iii) what consequential changes were needed to other aspects of the health service.

This list indicates the reach of these methods, from grass-roots community groups through senior corporate management issues to the highest levels of national government. The content of many if not all of the projects would have rendered them inaccessible to conventional OR.

Using PSMs

Working with Clients

PSM practitioners have to be able to manage not only the complexity of substantive subject matter but also the dynamics of interaction among workshop participants. The dual roles of analyst and of facilitator of group process place heavy demands on the consultant, who is called upon to deploy a wider range of skills than in conventional operational research practice. When operating as facilitator she has the responsibilities of ensuring that all voices are heard (not suppressed by psychological or hierarchical effects); that apparent agreement is not based on mutual misunderstanding of key terms; and that the precious (and usually expensive) opportunity presented by the gathering of key stakeholders is exploited in a timely and effective manner. (This experience is hard to simulate 'off line', and training should, if possible, include at least a brief experience of practical apprenticeship.) It is useful to have two facilitators with differentiated roles. One of them is likely to be heavily engaged, at times leading the discussion, at others concentrating acutely on the content of the discourse and also on the interpersonal issues that it reveals. The second facilitator can be principally involved with keeping a record, perhaps by direct computer input, of the evolving model. But he will also be able to intervene with insights that his colleague might otherwise miss through following the scent too closely.

PSMs are based on the working assumption that the client is not a sole decision-maker but a client system. Organisational politics is thus an integral aspect of project process, to which the consultants must be sensitive if they are not to be derailed. In order to achieve an effective process and worthwhile outcome it is important that all relevant stakeholders are represented. This requirement may bump up against numerical constraints – most practitioners cite a group size in the range 6–10 as desirable, and 12 should be the absolute maximum for a coherent group conversation to take place. There may be pressures to add people beyond this number for reasons of organisational politics, or to exclude certain clearly relevant stakeholders. These issues of the design of the group are ones that the consultant must address.

To guide the workshop with the consent and indeed respect of the group, the consultant must be, and be seen to be, disinterested – that is, not operating on behalf of any sectional interest. Where political tensions are active, this can require both sensitivity and agility from the facilitator. In inter-organisational working (for which the multi-perspective approach of PSMs makes them particularly appropriate), the question of access to the problem domain potentially acquires an additional twist. Initial contacts with one of the organisational actors will be necessary to gain entry to the problem forum - but that entry route may itself occasion doubts among other stakeholders as to the impartiality of the facilitation that follows.

Selecting Methods

There is no established process for the selection of method or methods to use in a particular engagement. This is often done on an intuitive basis – where uncertainties are seen as particularly salient in the problematic situation, Strategic Choice or Robustness

Analysis are plausible candidate methods; an evident conflict situation may suggest drama theory; and so on. There is of course also a choice to be made between using a traditional method or a PSM of any kind.

The most widely cited and discussed framework for this higher-level choice is due to Jackson and Keys (1984). Their 'system of systems methodologies' proposed two dimensions on which to describe the context of a problem. These were the degree of agreement among participants – which can be unitary (consensus), pluralistic (several viewpoints but agreement possible), or coercive (disagreements resolved through exercise of power); and the nature of the problem - simple or complex. This yielded six cells into which OR/systems methods were placed. For example, traditional hard OR was most suitable for simple–unitary contexts, System Dynamics for complex–unitary contexts, and Problem Structuring Methods for complex–pluralistic contexts.

However the criticism has been advanced that this framework makes the (unwarranted) assumption that the nature of the problem context can unerringly be identified in advance. Commonly, however, this will not be the case in the messy situations that PSMs are appropriate for. It may well be that only *after* the investigation is underway will the view to be taken of the problem context become clear. Furthermore, since the use of PSMs is a form of organised finding out, it is quite possible that this process will change the initial understanding of the problem context. For example, what was initially perceived by the relevant actors to be pluralist in character may, as a result of the intervention be reperceived as falling elsewhere on the spectrum of degree of agreement.

Mixing Methods

Another feature undermining the simplicity of the Jackson and Keys scheme is the fact that many PSMs consist of a loosely articulated set of processes (part technical, part social), with considerable freedom to switch phase or to recycle. They therefore lend themselves to creative re-assembly, in which different methods or parts of different methods are used in conjunction. Before theoretical discussion of this potential took off in the 1990's it was already a de facto reality in practice. The most high profile of many applications was the Sainsbury's case study (Ormerod 1996) already mentioned above, in which SODA, SSM and SCA were employed on a single engagement. In fact several of the cases summarised in the Applications section of this article involved the use of parts or wholes of PSMs in combination, or indeed the joint use of a PSM with a more conventional OR technique.

This ongoing practice was systematised and given a theoretical base by multimethodology (Mingers and Gill 1997). This advocates seeking to combine together a range of methods, perhaps across the hard/soft divide, in order to deal effectively and appropriately with the qualitatively different analytic challenges which a single problem situation may pose. Based on the work of Habermas (1984, 1987), any real-world problem situation can be seen as a complex mix of the material, the social, and the personal. Different methods are appropriate for analysing and making progress in these different strata. Thus material or physical characteristics can be modelled using traditional OR techniques, but social conventions, politics and power, and personal beliefs and values need quite different, qualitative approaches.

Any practical project goes through several stages - understanding and appreciating the situation, analysing information, assessing different options, and acting to bring about change. Moving from one phase to another offers an opportunity to transfer, based on the understanding achieved up to that point, to a different level (say from the material to the social) and to a corresponding type of analysis. The appropriate use of varied methods allows the project to evolve creatively, rather than pursuing the methodology adopted at its start, regardless of the understanding which is progressively developed. These are complementary arguments for combining together different PSMs, and indeed PSMs with other methods. Multimethodology facilitates a more varied palette, to match the developing richness of problem understanding.

Software and Other Technology

Several established PSMs have associated software: examples include STRAD (for Strategic Choice) and Decision Explorer (for SODA). These packages perform a variety of functions. They may display and re-organize concepts and their inter-relationships; identify a feasible range of options for action; elicit preferences using paired comparisons; compute simple quantitative attributes of options derived from the

current problem structure, and so on. They may also perform a variety of roles in the project, from technical assistance to the facilitator between group sessions, through enabling individual participants to pursue solo investigations, to the provision of an online Group Decision Support System. The use of the software during group sessions undoubtedly has an effect on group dynamics, focusing attention and giving a degree of control to whoever is in charge of inputting data or of changing the visual display. For this reason some leading practitioners prefer not to employ computers during the actual workshop sessions. In SODA, however, the computer display of sections of strategic maps is used deliberately in order to influence the group conversation. The computer model (i.e., map) is deployed as a 'facilitative device', so that group members will more easily accept and absorb concepts that are new to them. A concept that is advanced by another group member might provoke resistance – but one which whose presentation is neutrally framed by the computer may be easier to accommodate to.

The distinctive technology of PSMs is low- rather than high-tech. Ongoing models and other notes on deliberations are recoded on A1 flipchart sheets on the meeting room walls. Oval 'postit' notes are used to capture concepts in a way which facilitates re-structuring of model relationships during the session. At the end of a workshop it is normal for these traces to be photographed, and then emailed to participants. This visual record is a vivid reminder not only of the outcome of a workshop, but also of the process by which it was reached.

Implementation

A PSM workshop should leave time at the end for the group to agree an implementation strategy. If it is an intermediate workshop with others to follow, this process will constitute the allocation of responsibilities to group members (including the facilitators) to pursue clarification or uncertainty reduction activities that have been revealed as advantageous, so that the following meeting can take off from an improved position. With some PSMs the intervening work may consist of model development by the consultants – e.g., producing revised SSM 'root definitions'; in SODA reflecting the discussion in redrawn maps; and in SCA carrying out explorations to reduce relevant uncertainties. At what is expected to be a final workshop, where some conclusions have been agreed, the implementation strategy needs to be articulated and bought into by the key players. This will require a thorough discussion to identify the tasks (including for example a dissemination strategy) necessary for sustainable action to take place, and to specify responsibilities for these.

The experience within a PSM workshop, when it is working well, is frequently intense and the sense of release and satisfaction when a breakthrough is made can be palpable. Negotiated accommodations arrived at in this way can be creative escapes from apparently irresolvable tangles. However this almost cathartic experience is not transferable to non-participants. Generally only a part of the client system will be present at the workshop, and those not present may be reluctant to take its outputs on trust. Indeed it is more likely than not that those who can actually set the wheels in motion have not been members of the workshop. A report in conventional form which presents the case for the decisions arrived at in linear fashion the may be needed. For work within a single hierarchically structured organisation, top-down authority may carry the outputs of a PSM-based process towards implementation. In the case of inter-organisational work the situation is more complex, and the generation of acceptance among the various organisational constituencies can be problematic. It is clearly advisable for these problems of multiple acceptance to be discussed by the group, and to inform the implementation strategy.

Concluding Remarks

The progress of Problem Structuring Methods - in development and sophistication of methods, in applications, and in geographic spread - since they were recognised as a category with strong family resemblances has been fairly uninterrupted. There is one exception: the United States. The development of PSMs has been virtually ignored by the US OR/MS community. This was pointed out in an unprecedented letter to *ORMS Today* (Ackerman et al. 2009) signed by 45 academics from 11 countries and four continents. They cited as a strong contributory factor the systematic exclusion of papers on this topic from

US-based academic journals. An article in the same issue (Mingers 2009) explored the phenomenon in greater depth. For a further analysis of the difference in treatment of PSMs between the U.S. and the U.K., see Paucar-Caceres (2011).

In much of the rest of the world, PSMs have effected a breakout from the well developed but relatively confined arena of technocratic solutions to consensually defined problems occupied by OR's traditional methods. This outward movement has brought decision-support modelling in touch with a range of other methods and practices designed to help groups make progress with their problems. It has been suggested elsewhere (Rosenhead and Mingers 2001) that large group methods, development planning methods and community operational research are among the areas from which PSMs can learn, and to which PSMs can contribute.

The presence of Community OR (Midgley and Ochoa-Arias 2004) in this list is due to its natural fit with PSMs. Community OR is an analytic practice aimed at extending the customers of OR to include disadvantaged and non-hierarchical groups. With few resources, many of traditional OR's resource allocation tools are irrelevant. Furthermore the weak are perhaps disproportionately confronted with 'wicked', less well-structured problems; and the bottom-up nature of the PSM approach seems appropriate for the defined clientele. Its transparent modelling approach and group orientation does not present as many obstacles to engagement as would traditional OR's more mathematical approaches. No doubt these are among the reasons for the relatively high penetration of PSMs in this area.

There is now a substantial record of achievement for PSMs. There have been a wide variety of different types of use, both in context and in content. Surveys have shown there to be a good measure of user satisfaction. And there is an exciting range of possible further developments which appear to be reachable from the base that has already been achieved.

See

- ▶ Community OR
- ▶ Practice of Operations Research and Management Science
- ▶ Robustness Analysis
- ▶ Soft Systems Methodology
- ▶ Strategic Choice Approach (SCA)
- ▶ Strategic Options Development and Analysis (SODA)
- ▶ System Dynamics
- ▶ Wicked Problems

References

Ackerman, F., Bawden, R., et al. (2009). The case for soft or letter to the editor. *ORMS Today, 36*, 20–21.

Ackoff, R. L. (1979). The future of operational research is past. *Journal of the Operational Research Society, 30*, 93–104.

Ackoff, R. L. (1981). The art and science of mess management. *Interfaces, 11*, 20–26.

Ackoff, R. L. (1999). *Re-creating the corporation: A design of organization for the 21st century*. New York: Oxford University Press.

Checkland, P. B. (1981). *Systems thinking systems practice*. Chichester: Wiley.

Checkland, P., & Poulter, J. (2006). *Learning for action: A short definitive account of soft systems methodology, and its use for practitioners teachers and students*. Chichester: Wiley.

Checkland, P., & Scholes, J. (1990). *Soft systems methodology in practice*. Chichester: Wiley.

Eden, C. (1982). Problem construction and the influence of OR. *Interfaces, 12*, 50–60.

Eden, C., & Ackermann, F. (1998). *Making strategy: The journey of strategic management*. London: Sage.

Flood, R. L., & Jackson, M. C. (1991). *Creative problem solving: Total systems intervention*. Chichester: Wiley.

Friend, J., & Hickling, A. (2004). *Planning under pressure: The strategic choice approach* (3rd ed.). Oxford: Elsevier.

Habermas, J. (1984). *The theory of communicative action. Vol. 1: Reason and the rationalization of society*. London: Heinemann.

Habermas, J. (1987). *The theory of communicative action. Vol. 2: Lifeworld and system: A critique of functionalist reason*. London: Heinemann.

Howard, N. (1999). *Confrontation analysis: How to win operations other than war, CCRP*. Washington, DC: Department of Defense.

Jackson, M. C. (1987). Present positions and future prospects in management science. *Omega, 15*, 455–466.

Jackson, M. C., & Keys, P. (1984). Towards a system of systems methodologies. *Journal of the Operational Research Society, 35*, 473–486.

Midgley, G., & Ochoa-Arias, A. E. (Eds.). (2004). *Community operational research: OR and systems thinking for community development*. New York: Kluwer.

Mingers, J. (1992). Recent developments in critical management science. *Journal of the Operational Research Society, 43*, 1–10.

Mingers, J. (2009). Taming hard problems with soft O.R. – 'Soft' methodologies tackle messy problems that traditional O.R. can't touch, so why isn't it promoted in the U.S.? *ORMS Today, 36*, 48–53.
Mingers, J., & Gill, A. (Eds.). (1997). *Multimethodology: The theory and practice of combining management science methodologies*. Chichester: Wiley.
Mingers, J., & Rosenhead, J. (2004). Problem structuring methods in action. *European Journal Operational Research, 152*, 530–554.
Ormerod, R. J. (1996). Information systems strategy development at sainsbury's supermarkets using "Soft" ORC. *Interfaces, 26*, 102–130.
Paucar-Caceres, A. (2011). The development of management sciences/operational research discourses: surveying the trends in the US and UK. *Journal of the Operational Research Society, 62*, 1452–1470.
Rittel, H. W. J., & Webber, M. M. (1973). Dilemmas in a general theory of planning. *Policy Science, 4*, 155–169.
Rosenhead, J. (1986). Custom and practice. *Journal of the Operational Research Society, 37*, 335–343.
Rosenhead, J. (Ed.). (1989). *Rational analysis for a problematic world: Problem structuring methods for complexity, uncertainty and conflict*. Chichester: Wiley.
Rosenhead, J., & Mingers, J. (Eds.). (2001). *Rational analysis for a problematic world revisited: Problem structuring methods for complexity, uncertainty and conflict*. Chichester: Wiley.
Rosenhead, J., & Thunhurst, C. (1982). A materialist analysis of operational research. *Journal of the Operational Research Society, 33*, 122–133.
Schon, D. A. (1987). *Educating the reflective practitioner: Toward a new design for teaching and learning in the professions*. San Francisco: Jossey-Bass.
Vidal, R. V. V. (2004). Special issue on applications of problem structuring methods. *European Journal Operational Research, 152*, 631–640.

Processor Sharing

A queueing discipline whereby the server shares its effort over all customers present.

See

► Queueing Theory

Product Form

► Product-Form Solution

Product Form of the Inverse (PFI)

The inverse of a matrix expressed as the product of sequence of matrices. The matrices in the product are elementary elimination matrices.

See

► Eta File
► Simplex Method (Algorithm)

Product-Form Solution

When the steady-state joint probability of the number of customers at each node (station) in a queueing network is the product of the individual probabilities times a multiplicative constant, as in $\Pr\{N_1 = n_1, N_2 = n_2, \ldots, N_J = n_J\} = K\pi(n_1)\pi(n_2)\ldots\pi(n_J)$, the network is said to have a product-form solution. Sometimes the designation of a product-form solution requires that the multiplicative constant K also decompose into separate factors for each node, as holds for open Jackson networks but not for closed Jackson networks. Variants of such product-form solutions also occur in some non-network queues, such as those with vacations.

See

► Networks of Queues
► Queueing Theory

Product-Mix Problem

► Activity-Analysis Problem
► Blending Problem

Production Function

► Economics and Operations Research

Production Management

Jaya Singhal[1], Gabriel R. Bitran[2] and Sriram Dasu[3]
[1]University of Baltimore, Baltimore, MD, USA
[2]Massachusetts Institute of Technology, Cambridge, MA, USA
[3]University of Southern California, Los Angeles, CA, USA

Introduction

Some of the important objectives of a manufacturing system are to produce in a timely manner products that conform to specifications, while minimizing costs. The strategic measures of performance of a manufacturing system are cost, quality, flexibility, and delivery. Often hundreds of products are produced by a facility, and the entire production process may span several facilities that are geographically dispersed. In many industries the production network consists of plants that are located in different countries.

Production management entails many decisions that are made at all levels of the managerial hierarchy. Manufacturing processes involve a large number of people in many different departments and organizations, and utilize a variety of resources. In addition to the quality of human resources employed, operational efficiency depends upon the location and capacity of the plants, choice of technology, organization of the production system, and planning and control systems used for coordinating the day-to-day activities. The complexity of the problems associated with effectively and efficiently utilizing all the resources — manpower, machines, materials — needed for producing goods often necessitates the development of mathematical models to aid decision making.

Manufacturing decisions can be classified into three categories: strategic, tactical and operational. Strategic decisions pertain to decisions such as degree of vertical integration, items to produced inhouse, size and location of facilities, choice of technology, nature of equipment (general versus special purpose), long-term raw material and energy contacts, skills of employees, organization design, and so forth, that have long-term consequences and can not be easily reversed. Tactical decisions have shorter horizons of 6 month to 2 years. They include decisions such as aggregate production planning (levels of production and inventory, work force, and subcontracting), facility layout, and incremental capacity expansion. Operational decisions pertaining to issues such as order processing, detailed production scheduling, follow up, maintenance routines, and inventory control rules, drive the day to day activities.

The nature of the problems faced by a production manager depends on the characteristics of the market that the facility is competing in. For this reason it is useful to distinguish between different types of manufacturing systems. The variety and volume of products produced are critical for determining the type of the manufacturing system. Manufacturing systems have been classified into job shops, batch shops, flow lines and continuous processes on the basis of the volume and variety of the product mix. Job shops produce many different products in small quantities, each with different processing requirements. Typically the products are customized and are made only after receiving an order. At the other end of the spectrum are flow lines and continuous processes that produce a limited number of products in very high volumes. Demand is met from finished goods inventories. Batch shops lie in between these two extremes. Models for aggregate production planning are described first, followed by the models for job shops, batch shops, flow lines and continuous processes. Hopp and Spearman (2000) provide a detailed coverage of these and related topics.

Aggregate Production Planning is concerned with determination of the levels of production, inventory, work force, and subcontracting to respond to fluctuating demand. With a stable work force, the level of production can be changed by using over-time or undertime. The size of work force can be varied by hiring and layoff. Fluctuating demand can also be met by accumulating seasonal inventory. An organization may also have the option of backordering or losing sales. The relevant costs are for: (1) regular payroll and overtime; (2) carrying inventory; (3) backordering or lost sales (including the possible loss of customer goodwill, lost revenue, and penalties for late delivery): and (4) hiring (including training and learning) and layoff.

Real-world production planning may involve as many as 10,000 products (Hax and Candea 1984). With 10 decision periods, this can mean more than

100,000 variables. If the number of units sold is also a decision variable, the problem may involve more than 200,000 variables. Here quadratic and linear cost models are described. Hwang and Cha (1995), Nam and Logendran (1992), Silver et al. (1997), Thomas and McClain (1993), and Venkataraman and Smith (1996) have discussed other models and methodologies. Penlesky and Srivastava (1994) described the use of spreadsheets for production planning.

Quadratic cost models — Models with quadratic costs have several major advantages. They allow for a realistic cost structure in the planning process. They also allow uncertainties to be handled directly since they minimize the expected cost if unbiased expected demand forecasts are given (Hax and Candea 1984, p. 88; Simon 1956). The resulting solution is fairly insensitive to large errors in estimating cost parameters (Hax and Candea 1984). Hax and Candea also pointed out that this is an attractive property because of the difficulty in providing accurate cost.

The production and work force smoothing model developed by Holt et al. (1960) consists of a quadratic cost function constrained by linear equations to balance production, inventory, and sales. It selects production and work force levels in each of T periods so as to satisfy demand forecast while minimizing the sum of the costs over the T periods. Let P_t, W_t, I_t, and D_t represent production volume, work force level, end of period inventory, and demand forecast for period t, where the initial inventory and work force are given. The cost in period t consists of the following components:

Regular payroll costs :	$C_1 W_t$
Hiring and layoff costs :	$C_2(W_t - W_{t-1} - C_{11})^2$
Overtime costs :	$C_3(P_t - C_4 W_t)^2 + C_5 P_t - C_6 W_t + C_{12} P_t W_t$
Inventory related costs :	$C_7(I_t - C_8 - C_9 D_t)^2$

The model may be formulated as:

$$\text{Minimize Z} = \sum_{t=1}^{T} [(C_1 - C_6)W_t + C_2(W_t - W_{t-1} - C_{11})^2 + C_3(P_t - C_4 W_t)^2 + C_5 P_t + C_{12} P_t W_t + C_7(I_t - C_8 - C_9 D_t)^2] \quad (1)$$

subject to:

$$P_t - D_t = I_t - I_{t-1} \quad (2)$$

Holt et al. focused on an infinite planning horizon with stationary costs and derived the following two linear decision rules for the first period:

$$P_1 = \theta_1 + \theta_2 I_0 + \theta_3 W_0 + \sum_{t=1}^{T} \phi_t D_t$$

$$\text{and } W_1 = \theta_4 + \theta_5 I_0 + \theta_6 W_0 + \sum_{t=1}^{T} \mu_t D_t,$$

where $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_t$, and μ_t ($t = 1, 2, \ldots, T$) are functions of the cost coefficients. The infinite series can be truncated after an appropriate number of periods T.

Singhal and Singhal (1996) developed simple computational procedures for finite horizon cases. These can be used for arbitrary time-varying cost coefficients. The complexity of the procedures grows only linearly with T. They generate the values of production, work force, and inventory levels for each period in the planning horizon. Finally, the procedures lend themselves to sensitivity analysis with respect to terminal values and to generate alternate plans.

It is beneficial to generate a collection of alternate plans on the basis of alternative terminal conditions and evaluate them more precisely according to the actual cost structure. This is usually more complex than the quadratic cost function used in the Holt et al. model. Sensitivity analysis can also be used to eliminate plans that may include negative values of P_t, W_t, or I_t. If only I_T is specified, one can compute Z as a simple quadratic function of W_T : $Z = h + kW_T + mW^2_T$ where h, k, and m are functions of the cost coefficients, ending inventory, and demand forecasts. The optimum value of W_T is then easily computed as $W_T = -k/2m$. If W_T, rather than I_T, is specified, then Z can be obtained as a quadratic function of I_T. If the terminal condition is not specified for any variable, one can obtain Z as a quadratic function of both W_T and I_T (or P_T).

One can compute optimal plans for a menu of combinations of terminal values (I_T, W_T) so as to

create a menu of alternative plans which can be evaluated in more detail with respect to alternative cost structures, constraints, and objectives. The alternate plans provide considerable flexibility to the decision maker because they can be evaluated in the context of (a) constraints not included in the model, (b) actual costs, and (c) implications beyond the planning horizon.

Constraints not included in the model — The model does not specify that P_t, W_t, or I_t be non-negative. The solution approaches developed by Holt et al. (1960) or Singhal and Singhal (1996) do not guarantee it either. However, the values of θ_1, θ_2, θ_3, θ_4, θ_5, θ_6, θ_t, and μ_t ($t = 1, 2, \ldots, T$) in the decision rules for an actual problem (Holt et al. 1960) indicate that for most problems, they will be nonnegative. For cases where a solution may include negative values of P_t, W_t, or I_t, sensitivity analysis can be used to determine the ranges of the terminal boundary conditions for which all values of P_t, W_t, and I_t are non-negative. If implementation of the optimal solution is difficult because of extremely low or extremely high levels of inventory, production, or work force in some periods, trade-offs can be made between the additional cost and the ease of implementation of alternate plans that are within the constraints on inventory, production, and work force.

Actual costs — The costs of various plans, including the optimal plan refer to the costs approximated by the linear quadratic model, not to the actual costs. In testing the model for a real-world problem, one may obtain actual costs for one or more alternate plans that are lower than those of the optimal plan.

Implications beyond the planning horizon — The organization may anticipate or plan some changes beyond the planning horizon of the model. For example, it may retire workers or introduce technology that requires fewer or more workers. If the organization plans to introduce technology that requires fewer workers, it would choose a plan that would require a smaller work force towards the end of the planning horizon. Similarly, if some workers are expected to retire in the near future, the organization would choose a plan that would require hiring more workers towards the end of the planning horizon. The exact choice of the plan will depend on the magnitude of the changes beyond the planning horizon and the cost penalty during the planning horizon. In some cases, the optimal levels of inventory and work force in the final period may be incompatible with the demand forecasts for periods beyond the planning horizon (these forecasts may be too imprecise to extend the length of the planning horizon but they may indicate the overall magnitude of demand). In such cases, trade-offs can be made by comparing the possible benefits of an alternate plan and the cost penalty associated with it.

Both the finite and infinite horizon versions can be implemented on the rolling basis. In the infinite horizon version, no consideration is given to information beyond a certain period. In the finite horizon version, the implications beyond the planning horizon are first included in the specification of the terminal conditions and then evaluated through sensitivity analysis.

Bergstrom and Smith (1970) extended the Holt et al. model to a multi-product situation. It is given as

$$\text{Minimize TC} = \sum_{t=1}^{T}\left[C_1 W_t + \sum_{i=1}^{N}\left[C_{i7}(I_{it} - C_{i8} - C_{i9}D_{it})^2\right] + C_3\left(\sum_{i=1}^{N} k_i P_{it} - C_4 W_t\right)^2 + \sum_{n=1}^{N} C_5 k_i P_{it} - C_6 W_t + C_{12} W_t\left(\sum_{i=1}^{N} k_i P_{it}\right) + C_2(W_t - W_{t-1} - C_{11})^2\right]$$

subject to:

$$I_{it} = I_{i,t-1} + P_{it} - D_{it}, \quad i = 1,2,\ldots,N; \quad t = 1,2,\ldots,T,$$

where N and T denote the number of products and periods respectively; P_{it}, D_{it}, and I_{it} represent the production, demand forecast, and inventory of product i during period n; k_i represents the standard labor time to complete one unit of product i; and W_t represents the work force during period t. The C_i are the cost coefficients. Aggregate production L_t,

aggregate inventory I_t, and aggregate demand forecast D_t can be written as

$$L_t = \sum_{i=1}^{k_i} P_{it}, \quad t = 1, 2, \ldots, T$$

$$I_t = \sum_{i=1}^{k_i} I_{it}, \quad t = 1, 2, \ldots, T$$

$$D_t = \sum_{i=1}^{k_i} D_{it}, \quad t = 1, 2, \ldots, T$$

$$I_{it} = I_{t-1} + L_t - D_t, \quad t = 1, 2, \ldots, T$$

All decision variables are unconstrained. Initial conditions I_0, W_0, and I_{0i} ($i = 1, 2, \ldots, N$) and the final conditions (work force and aggregate inventory) are specified. Singhal (1992) developed a simple and efficient non-iterative algorithm for obtaining the optimal values of the levels of production management in, inventory, and work force during the planning horizon. The efficiency is achieved by exploiting the special structure of the recurrence relations obtained by differentiating the cost function. Once the input data are developed, the computation time will remain the same irrespective of the number of products which, as noted earlier, could be as many as 100,000.

Linear cost models — Linear programming models are widely used because they can be easily tailored to a specific situation. Many constraints can be directly included in the model. A major advantage of linear programming models is the availability of computer codes that can solve very large problems. Most cost structures are generally linear within the range of interest. If they are not, one can use linear approximations. Another advantage is parametric and sensitivity analyses. The dual solution can be used to obtain the costs of constraints and one can easily perform sensitivity analysis on cost parameters and demand forecasts. For a more detailed discussion of linear programming models, see Hax and Candea (1984) and Silver et al. (1997). Hax and Candea (1984) described the following general purpose model:

$$\text{Minimize } Z = \sum_{i=1}^{N} \sum_{t=1}^{T} \left(d_{it} P_{it} + c_{it} I_{it}^{+} + b_{it} I_{it}^{-}\right) + \sum_{t=1}^{T} \left(w_t W_t + o_t O_t + h_t H_t + f_t F_t\right)$$

subject to:

$$P_{it} + I_{i,t-1}^{+} - I_{i,t-1}^{-} - D_{it} = I_{it}^{+} - I_{it}^{-} \qquad i = 1, 2, \ldots, N;$$
$$t = 1, 2, \ldots, T,$$
$$W_t - W_{t-1} = H_t - F_t \qquad t = 1, 2, \ldots, T,$$
$$O_t \le pW_t \qquad t = 1, 2, \ldots, T,$$
$$P_{it}, I_{it}^{+}, I_{it}^{-}, W_t, O_t, H_t, F_t \ge 0$$
$$i = 1, 2, \ldots, N; \qquad t = 1, 2, \ldots, T$$

P_{it} = Units of item i to be produced in period t
D_{it} = Forecast demand for item i in period t
d_{it} = Cost of producing one unit of product i in period t
c_{it} = Cost of carrying one unit of inventory of product i from period t to $t + 1$
b_{it} = Cost of backordering one unit of inventory of product i from period t to $t + 1$
w_t = Cost of one regular labor hour in period t
W_t = Regular labor hours employed in period t
o_t = Cost of one overtime labor hour in period t
O_t = Overtime labor hours used in period t
h_t = Cost of hiring one labor hour in period t
H_t = Labor hours of regular work force hired in period t
f_t = Cost of laying off one labor hour in period t
F_t = Labor hours of regular work force laid off in period t
i_{it}^{+} = Inventory of product i at the end of period t
I_{it}^{-} = Units of product i backordered at the end of period t
p = An upper bound on overtime as a fraction of regular hours

The first constraint is similar to the production-inventory balance equation in the linear-quadratic model when $I_{it} = I_{it}^{+} - I_{it}^{-}$, $t = 1, 2, \ldots, T$. The second constraint shows the changes in the level for work force due to hiring and layoff. The third constraint provides a limit on the overtime; the limit is proportional to the level of work force.

Job Shops

Job shops specialize in producing customized products, and the production process has the flexibility to produce many different products. Due to the high variety the flows in job shops are jumbled, thus making it very difficult to predict and manage the completion times of jobs. Since most of the jobs are

produced after receiving an order from a customer, very important managerial tasks are to accurately predict due dates, ensure that the quoted dates are not violated, and use resources effectively and efficiently.

Operational Problems — The challenge of managing day to day operations has given rise to a rich set of combinatorial optimization problems. The most basic operational problem is to determine a schedule that specifies when each job will be allocated different resources. Associated with each job are the arrival time, a due date and a set of operations. Each operation requires a set of resources for some duration, and there may be precedence constraints on the order in which the operations can be performed.

A variety of performance measures have been considered for evaluating alternative schedules. Common performance measures are the average or maximum time a set of jobs remains in the facility, number of jobs that are late, or the average or maximum tardiness for a set of jobs. Most of the problems of job shop schedule optimization problems, except for a small class, are computationally intractable (Lenstra et al. 1977; French 1982). Hence for most practical problems the emphasis has been on heuristics.

Researchers have successfully analyzed job shops with special structures. Many insights have been gained into the single machine and single stage, multiple machine scheduling problems. For multiple stage job shops, analysis has been possible, provided all the jobs follow the same route.

Job shop scheduling models can be classified into static and dynamic models. In static models the set of requirements including job arrival times and processing requirement are known in advance. In contrast, in dynamic job shop models new arrivals are permitted. The arrival times may be stochastic and the processing requirements may also vary dynamically.

Mathematical programming approaches have been employed to study static job shop problems. For performance measure that are non-decreasing in the completion time of the job, dynamic programming techniques have been employed to generate optimal solutions for problems of modest size. Dynamic programming based approaches have also been useful in identifying dominance criteria to reduce the number of schedules to be evaluated. Several heuristics have been developed that exploit dominance criteria. Integer programming formulations of scheduling problems have also been used to generate near optimal solutions. Typically some complicating constraints in the integer program are relaxed to yield tractable sub-problems.

While most of the theory focuses on static job shop models that assume deterministic requirements, most practical problems are dynamic and stochastic. For such complex environments analysis has largely been restricted to simulations of local dispatching rules. Each station employs a dispatching rule — for example, process jobs in increasing order of processing times — and the overall performance of the shop is evaluated via Monte Carlo simulations. Many dispatching rules have been discussed in the literature. Further details regarding scheduling algorithms are given in Conway et al. (1967), Graves (1981), and O'Eigeartaigh et al. (1985).

An important development in the area of scheduling dynamic shops has been to approximate the job shop scheduling problem by a Brownian control problem. Although the size of the networks analyzed is small, since the focus is on bottleneck stations the method is useful in many practical situations. The Brownian control problems have been useful in identifying near optimal scheduling policies for minimizing the average lead times (Wein 1990).

Strategic and tactical problems — Since most of the operational problems of sequencing and scheduling jobs through a shop floor are computationally intractable, there is a need to design the job shops such that simple real time control rules are adequate to obtain good performance. The long term performance of the shop will depend on the types of jobs processed by the facility (product mix), the capacity and technology of different stations, and the rules employed to quote due dates and manage the flow through the shop floor. Tactical and strategic decisions regarding each of these variables require models that predict the medium to long term performance of the job shops.

One approach for assessing the long term performance is to employ Monte Carlo simulations. The strength of simulation models lies in their ability to incorporate many features, such as (i) complex control rules — for example, local dispatching rules, control of input to the shop, etc.; (ii) complex arrival patterns — for example, correlated demands, non-stationary demand, etc.; and (iii) complex

resource requirements and availability — for example, multiple resources, machine failures, etc. A broad range of performance measures can also be assessed through simulation models. These models, however, are time consuming and cannot identify optimal parameters for the policies being investigated.

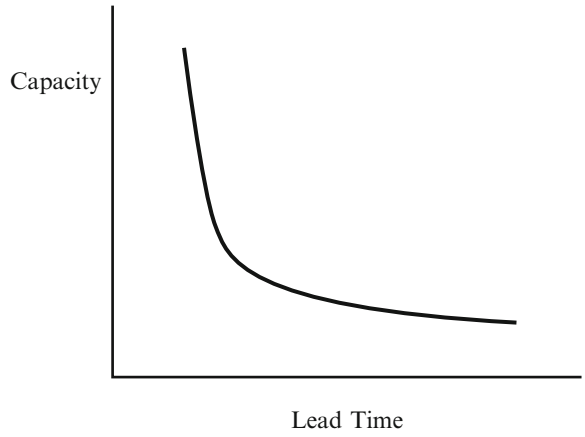
Open queueing network models have been proposed to evaluate the long term performance of job shops. Good approximation procedures have been developed to estimate the average queue lengths in networks with features such as general processing and interarrival time distributions, multiple job classes, and class dependent deterministic routing through the network.

An approximation procedure that has been frequently employed is the parametric decomposition approach (PDA). Under the PDA, each node is treated as being stochastically independent and all the performance measures are estimated based on the first two moments of the inter-arrival and service time distributions at each node. Extensive testing has shown that PDA provides accurate estimates of the average queue length at each node in very general networks. Limitations of the approach are that all the measures are for steady state, only the average queue lengths are accurately predicted, and the analysis is based on the assumption that the jobs are processed on a first come first served basis. Nevertheless, the power of this approach lies in the ease with which complex networks can be analyzed, which in turn facilitates the design of networks.

The PDA has enabled the analysis of several optimal facility design problems. One such problem is:

- Objective: Minimize total cost of equipment.
- Decision Variables: Capacity of each station in the network, and technology.
- Constraints: Upper bounds on the average lead time for different job classes.

This model addresses the relationship between average lead times and the choice of equipment. Since system design is based on multiple criteria, it is useful to develop curves that reflect the trade-off between lead times and cost of equipment. This can be done by parametrically varying the upper bound on the permissible lead times. Figure 1 provides a possible trade-off curve (Bitran and Tirupati 1989). Details regarding the application of queueing models to job shops are given in Bitran and Dasu (1992).



Production Management, Fig. 1 Illustrative trade-off curve

Batch Shops

The variety of jobs processed in a batch shop is less than that in job shops; furthermore, the set of products that are produced by the facility may be fixed. Nevertheless, the production volume of each product is such that several products may share the same equipment. Often the demand for final goods is met from finished goods inventory and production plans are based on demand forecasts. A large number of discrete part manufacturing systems can be classified as batch shops.

Operational problems: The time and cost for switching machines from one product to the next poses one of the biggest problems in managing batch shops. Although job shops can also have significant set-ups, since each job is unique the set-up time can be incorporated in that job's total processing time. On the other hand, in batch shops, the same products are produced repeatedly and there is an opportunity to mitigate the effect of set-ups by combining or splitting orders. Consequently much attention has been paid to problems of determining batch quantity of and the sequence in which each item is produced. The primary trade-offs are between inventory carrying, shortage and set-up costs.

A classic lot sizing problem is the economic lot scheduling problem (ELSP). The ELSP seeks the optimal lot size at a single production stage when the demand rate for each item is fixed and deterministic (Panwalker and Iskander 1977). The objective of the analysis is to determine the frequency with which each item is to be produced so as to minimize the average set-up and holding costs without ever stocking out.

Many of the solution procedures for ELSP consist of three steps. First, ignoring the capacity constraint, the optimal production frequency for each item is determined. Next the frequencies are rounded off to an integer multiple of a base period. In the final step a solution that specifies the sequence in which each item is produced is generated. Roundy (1986) showed that in the second step if the integer multiple is restricted to some power of 2, then a near optimal solution can be found. In recent years researchers have begun to extend the approaches developed for ELSP to multistage multi-machine problems.

ELSP is a continuous time model. In practice production plans are made on a periodic basis, prompting several researchers to develop and analyze discrete time models of the lot-sizing problems. Below a single-stage, multi-item, multi-period, capacitated lot-sizing problem is formulated:

$$\text{Minimize} \sum_{t=1}^{T} \left\{ p_t(X_t) + h_t(I_t) + \sum_{i=1}^{I} s_{it}\delta(X_{it}) \right\}$$

subject to:

$$I_{i,t-1} + X_{it} - I_{it} = D_{it} \quad t = 1,2,\ldots,T; \quad i = 1,2,\ldots,I.$$

$$\sum_{i=1}^{I} X_{it} \le X_t \quad t = 1,2,\ldots,T.$$

$$\delta(X_{it}) = \begin{cases} 1 & \text{if } X_{it} = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$I_{it}, X_{it} \ge 0 \quad t = 1,2,\ldots,T; \quad i = 1,2,\ldots,I.$$

where X_{it}, I_{it}, C_t, D_{it} and s_{it} denote respectively for period t and product i, the production quantity, the ending inventory, the capacity, the demand, and the setup cost; X_{it} and I_{it} are the only decision variables; and X_t and I_t are vector with elements $\{X_{it}\}$ and $\{I_{it}\}$, respectively. The functions $p_t(\cdot)$ and $h_t(\cdot)$ denote respectively the variable production and inventory holding costs.

Once again, except for a small class, the lot-sizing problems are NP-hard (Garey and Johnson 1979); Bitran and Yanasse 1982). The following two lot-sizing problems, however, can be solved in polynomial time and have been the basis of many approximation procedures: (a) the single item lot-sizing problem without capacity constraints, and concave variable production and inventory holding costs; and (b) single item problem, with constant capacity, and concave variable production and inventory holding costs.

Multistage systems producing multiple products with dynamic demands, usually require extensive information and considerable computational effort to find optimal solutions. For these reasons, hierarchical planning systems have been proposed. At the highest level in the hierarchy an aggregate plan with a horizon of several, usually 12, months is developed. If the demand is seasonal, the horizon should cover the full demand cycle. Over such horizons it is impractical to obtain detailed information about demand for each item and the availability of every resource. Hence, it becomes necessary to aggregate the items into families, and the machines into machine centers, etc. The aggregate plan determines the time phased allocation of aggregate resources to different part families. The plan focuses on the primary trade-offs among the cost of varying production resources employed by the firm, the costs of carrying inventory (and possibly backordering demand), and major setup costs. The extended horizon enables the facility to respond to seasonality in demand.

The aggregate plan becomes the basis for determining the detailed production schedule for each item. The detailed resource allocation decisions are constrained by the decisions made at the aggregate planning level.

The number of hierarchical planning stages, the degree of aggregation at each level, and the planning horizon lengths affect the quality of the plan and must be carefully determined for each context. Many researchers have studied hierarchical planning systems. Bitran and Tirupati (1993) and Hax and Candea (1984) contain discussions of this approach.

Once the plans have been disaggregated and the monthly requirements of each item are known, there are a number of approaches for scheduling and controlling the flow of the items through the shop. One approach is to time the release of the orders to the shop so that the required quantities of the items become available by the date specified by the hierarchical planning system. In this approach, also referred to as the push system, an estimate is made of production lead times, and order releases are offset by the lead times. The scheduling decisions at each work station may be made on the basis of the queue in front of each work station. Scheduling models developed for job shops are also useful here.

An alternate approach for operating the shop is the pull system. Under this approach the work-in-process inventory level after a production stage determines the production decisions at that stage. The buffer inventories are maintained at planned levels and a production order is triggered if the inventory level drops below the threshold.

Since the push system operates on the basis of planned lead times, OR/MS models have been developed to understand the relationship between release rules, capacity and lead times. The key decision variable in pull systems is the size of each buffer. Several researchers have examined the impact of buffer sizes on the shop performance (Conway et al. 1988).

Strategic and Tactical Problem — An approach advocated for simplifying the operations of batch shops is to partition the facility into cells. Parts produced by the facility are grouped into families and each family is assigned to a cell. Ideally all operations required for a family of parts are performed in the same cell. The advantages of cellular manufacturing systems are simplified flows, and reduced lead times and setup costs. These benefits may be partially offset by the need for additional equipment. Many different criteria — such as part geometry, production volumes, setups, and route through the shop — have been proposed for forming part families. Researchers have also investigated several algorithms for identifying alternative partitions. Typically these algorithms begin with a product-process matrix. In this matrix rows correspond to parts and columns correspond to machines. An element ij in this matrix is one if a part i requires a machine j and zero otherwise. The columns and rows of the matrix are interchanged so as to produce a block diagonal matrix. Each block identifies a set of resources and jobs that does not interact with the remaining operations, and so corresponds to a cell.

As in the case of job shops, batch shops system design can be improved if the medium to long term performance of the shop can be assessed. Closed and open queueing network models and simulation based models are useful for assessing the long term performance of batch shops. The objective of these models is to determine the relationship among capacity of different cells, lot sizes, and lead times (Bitran and Dasu 1992).

Queueing network models assume that the processing rate at each station is fixed. In practice the processing rate at each station may vary. Variations may be due either to the allocation of additional (human) resources to a stage or simply because the queue length has a motivational effect on the machine operator. Based on these observations, in recent years an alternative class of tactical models of the shop have been proposed (Graves 1986). Here the production rates are assumed to vary as a function of the size of the queue length. The processing rates at each stage are allowed to vary so as to ensure that the time spent at a station is the same for every job. The model therefore enables managers to plan the lead times for each stage.

Flow Lines and Continuous Operations

Included in this class are all systems that are dedicated to the production of (one or few) items in large volumes. Examples of such systems include assembly lines, transfer lines, and continuous operations such as cement and oil derivatives manufacture. The demand is often met from finished goods inventory and thus the main focus tends to be on the management of the corresponding inventory levels and the supply chain. The operational problems are relatively simple and are omitted.

Tactical problems — An important operational problem is to manage the trade-off between the cost of varying the production rate and the cost of finished goods inventory. The aggregate planning models discussed earlier are applicable here. Typically, all the stages of the production system have equal capacity, hence, managing the flow through the facility does not pose a significant problem. In assembly lines, the balance is achieved by carefully assigning tasks to different work stations — a complex combinatorial optimization problem. Several algorithms have been developed for assembly line balancing.

Strategic problems — High volume production systems frequently compete on the basis of low costs and supply large geographically dispersed markets. It is therefore not uncommon to have many plants that cater to different markets. OR/MS models have been developed to aid in the design of the multiplant

networks and the distribution systems (Erlenkotter 1978; Federgruen and Zipkin 1984). Here the discussion is restricted to the plant location problems.

The number of plants their capacity and location have a big effect on production and distribution costs. Models have been developed to analyze the trade-off between the fixed costs of setting up plants and the variable (transportation and production) costs of operating the plants. The models assume that a set of markets with known demands have to be supplied and the decision variables are the number of plants, their location and capacities (Erlenkotter 1978).

Concluding Remarks

Production management involves many complex trade-offs. As a result many mathematical models have been developed to aid decision makers. This is certainly not an exhaustive list and excludes many important problem areas such as inventory management, preventive maintenance, capacity expansion, and quality control. The focus has been on models that are concerned with the flow of goods through a manufacturing system. Even within this domain, in order to provide a broad overview, many important models that deal with specialized systems were not discussed, such as intelligent manufacturing systems.

The problems arising in each type of production system were described as if each plant operated in isolation. In practice, a production system is likely to consist of a network of plants. While some plants may be batch or job shops others are likely to be assembly or continuous processes. The problems of coordinating these networks was not discussed.

Most of the OR/MS models focus on managing the trade-offs among setup costs, inventory carrying costs and cost of varying production rates. On the other hand, many gains in productivity are due to the elimination (or mitigation of) the factors that give rise to these trade-offs. For example reduction in set-up costs and times reduces lead times, increases the ability of the system to produce a wider mix of products, diminishes the role of inventories and simplifies the management of batch shops. Researchers have begun to develop models that quantify the benefits of and guide such process improvement efforts (Porteus 1985; Silver 1993).

See

- ▶ Automation in Manufacturing and Services
- ▶ Dynamic Programming
- ▶ Facilities Layout
- ▶ Facility Location
- ▶ Flexible Manufacturing Systems
- ▶ Hierarchical Production Planning
- ▶ Integer and Combinatorial Optimization
- ▶ Inventory Modeling
- ▶ Job Shop Scheduling
- ▶ Location Analysis
- ▶ Operations Management
- ▶ Queueing Theory
- ▶ Supply Chain Management

References

Bergstrom, G. L., & Smith, B. E. (1970). Multi-item production planning — An extension of the HMMS rules. *Management Science, 16*, 614–629.

Bitran, G. R., & Dasu, S. (1992). A review of open queueing network models of manufacturing systems. *Queueing Systems: Theory and Applications, 12*, 95–134.

Bitran, G. R., & Tirupati, D. (1989). Trade-off curves, targeting and balancing in manufacturing networks. *Operations Research, 37*, 547–564.

Bitran, G. R., & Tirupati, D. (1993). Hierarchical production planning. In S. C. Graves, A. H. G. Rinnooy Kan, & P. Zipkin (Eds.), *Logistics of production and inventory* (Handbooks in O.R. and M.S, Vol. 4). Amsterdam: Elsevier Science Publishers.

Bitran, G. R., & Yanasse, H. H. (1982). Computational complexity of capacitated lot sizing problem. *Management Science, 28*, 1174–1186.

Burbridge, J. L. (1979). *Group technology in the engineering industry*. London: Mechanical Engineering Publications.

Conway, R. W., Maxwell, W., McClain, J. O., & Thomas, L. J. (1988). The role of work-in-process inventory in serial production lines. *Operations Research, 36*, 229–241.

Conway, R. W., Maxwell, W. L., & Miller, L. W. (1967). *Theory of scheduling*. Reading, MA: Addison-Wesley.

Erlenkotter, D. (1978). A dual-based procedure for uncapacitated facility location. *Operations Research, 26*, 992–1005.

Federgruen, A., & Zipkin, P. (1984). Approximation of dynamic multi-location production and inventory problems. *Management Science, 30*, 69–84.

French, S. (1982). *Sequencing and scheduling: An introdution to the mathematics of the job-shop*. New York: John Wiley.

Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to the theory of N.P. completeness*. San Francisco: Freeman.

Graves, S. C. (1981). A review of production scheduling. *Operations Research, 29*, 646–675.

Graves, S. C. (1986). A tactical planning model for job shop. *Operations Research, 34*, 522–533.
Hax, A. C., & Candea, D. (1984). *Production and inventory management*. New Jersey: Prentice-Hall.
Holt, C. C., Modigliani, F., Muth, J. F., & Simon, H. A. (1960). *Planning production, inventories, and work force*. Englewood Cliffs, NJ: Prentice-Hall.
Hopp, W. J., & Spearman, M. L. (2000). *Factory physics* (2nd ed.). New York: Irwin/McGraw Hill.
Hwang, H., & Cha, C. N. (1995). An improved version of the production switching heuristic for the aggregate production planning problem. *International Journal of Production Research, 33*, 2567–2577.
Lenstra, J. K., Rinnooy Kan, A. H. G., & Brucker, P. (1977). Complexity of machine scheduling problems. *Annals of Discrete Mathematics, 1*, 343–362.
Nam, S. J., & Logendran, R. (1992). Aggregate production planning — A durvey of models and methodologies. *European Journal of Operational Research, 61*, 255–272.
O'Eigeartaigh, M., Lenstra, J. K., & Rinnooy, A. H. G. K. (1985). *Combinatorial optimization — Annotated bibliographies*. New York: John Wiley.
Panwalker, S. S., & Iskander, W. (1977). A survey of scheduling rules. *Operations Research, 25*, 45–61.
Penlesky, R., & Srivastava, R. (1994). Aggregate production planning using spreadsheet software. *Production Planning & Control: The Management of Operations, 5*, 524–532.
Porteus, E. L. (1985). Investing in reduced setups in the EOQ model. *Management Science, 31*(8), 998–1010.
Roundy, R. (1986). A 98% effective lot-sizing rule for a multi-product, multi-stage production/inventory system. *Mathematics of Operations Research, 11*, 699–727.
Silver, E. A. (1993). Modeling in support of continuous improvements towards achieving world class operations. In R. Sarin (Ed.), *Perspectives in operations management: essays in honor of Elwood S. Buffa*. Norwell, MA: Kluwer.
Silver, S. A., Pyke, D. F., & Peterson, R. (1997). *Inventory management and production planning and scheduling* (3rd ed.). New York: John Wiley.
Simon, H. A. (1956). Dynamic programming under uncertainty with a quadratic cost function. *Econometrica, 24*(1), 74–81.
Singhal, K. (1992). A noniterative algorithm for the multiproduct production planning and work force planning problem. *Operations Research, 40*, 620–625.
Singhal, J., & Singhal, K. (1996). Alternate approaches to solving the Holt et al. model to performing sensitivity analysis. *European Journal of Operational Res., 91*, 89–98.
Thomas, J., & McClain, J. O. (1993). An overview of production planning. In S. C. Graves, A. H. G. Rinnoy Kan, & P. Zipkin (Eds.), *Logistics of production and inventory* (Handbooks in O.R. and M.S, Vol. 4). Amsterdam: Elsevier Science Publishers.
Venkataraman, R., & Smith, S. B. (1996). Disaggregation to a rolling horizon master production schedule with minimum batch-size production restrictions. *International Journal of Production Research, 34*, 1517–1537.
Wein, L. M. (1990). Optimal control of a two-station Brownian network. *Mathematics of Operations Research, 15*, 215–242.

Production Rule

A mapping from a state space to an action space, generally used in modular knowledge representation. With roots in syntax-directed parsing of language, production rules comprise a basic reasoning mechanism, particularly in heuristic search.

See

▶ Artificial Intelligence
▶ Expert Systems

Program Evaluation

Edward H. Kaplan and Todd Strauss
Yale University, New Haven, CT, USA

Introduction

Program evaluation is not about mathematical programming, but about assessing the performance of social programs and policies. Does capital punishment deter homicide? Which job training programs are worthy of government support? How can emergency medical services be delivered more effectively? What are the social benefits of energy conservation programs? These are the types of questions considered in program evaluation.

Notable evaluations include the Westinghouse evaluation of the Head Start early childhood program (Cicarelli et al. 1969), the Housing Allowance experiment (Struyk and Bendick 1981), the Kansas City preventive patrol experiment (Kelling et al. 1974), and evaluation of the New Haven needle exchange program for preventing HIV transmission among injecting drug users (Kaplan and O'Keefe 1993). As these examples suggest, questions and issues deserving serious evaluation often are in the forefront of social policy debates in areas such as public housing, health services, education, welfare, and criminal justice.

Closely related to program evaluation are the activities of cost-benefit and cost-effectiveness

analysis. These resource allocation methods help decision makers decide which social programs are worth sponsoring, and how much money should be invested in competing interventions. Program evaluation may be construed as an attempt to understand and estimate the benefits associated with the social program under study. While some evaluations attempt to relate these benefits to the costs of program activities, most program evaluations are viewed as attempts to measure benefits alone.

Program evaluation is often conducted by social scientists at the behest of organizations with some interest in the program, either as participants, administrators, legislators, managers, program funders, or program advocates. In such a charged atmosphere, how can OR/MS be useful? Program evaluation contributes to policy making chiefly by informing policy debate. Evaluation can be construed as an activity that produces important information for decision makers in the policy process (Larson and Kaplan 1981). Evaluation is also useful for framing issues, and for identifying and choosing among policy options. Evaluation is crucial to program administrators concerned with improving service delivery. These tasks are about gathering, analyzing, and using information. It is the orientation toward decision making that renders OR/MS particularly useful in the evaluation of public programs.

Program Components and the Scope of Evaluation

In the language of systems analysis, the components of social programs can be classified as inputs, processes, and outputs (Rossi and Freeman 1993). Inputs are resources devoted to the program, while outputs are products of the program. In this framework, program evaluation is usually about assessing a program's effects on outputs. Such evaluation is often called outcome or impact evaluation. Typically, the result of outcome evaluation is the answer to the question: Did the program achieve its goals?

In contrast to outcome evaluation, process evaluation is often referred to, perhaps pejoratively, as program monitoring. As the myriad details of real programs are classified simply as processes in monitoring studies, programs become black boxes. Such a framework is anti-operational. On the other hand, an OR/MS approach to process evaluation focuses on program operations, often with the assistance of appropriate mathematical models. Typical program evaluations too often lead to simplistic conclusions regarding which programs work. Focusing on program operations often results in understanding why some programs are successful and other programs fail. As an example, consider Larson's analysis of the Kansas City Preventive Patrol Experiment (Larson 1975). This experiment attempted to discern the impact of routine preventive patrol on important outcomes such as crime rates and citizen satisfaction, in addition to important intermediate outcomes such as response time and patrol visibility. The empirical results of this experiment resulted in several findings of "no difference" between patrol areas with supposedly low, regular, and high intensities of police preventive patrol. In contrast, Larson's application of back-of-the-envelope probabilistic models to this experiment showed that one should have expected such results due to the nature of the experimental design. He showed, for example, that one should not have expected large differences in police response times given the peculiarities of patrol assignments and call-for-service workloads evident in the experiment. The same models suggested that different experimental conditions, better reflecting police operations in other large American cities, could lead to different results.

An advantage of an OR/MS approach to program evaluation is that goals and objectives are stated as explicitly as possible. What is the purpose of the program under study, and how does one characterize good versus poor program performance? While the importance of such questions may be self-evident to OR/MS practitioners, most actors on the policy stage are not accustomed to such explicitness. The act of asking such questions is often, by itself, a contribution to policy debate. A defining feature of the OR/MS approach to problem solving is the association of one or more performance measures with program objectives. A performance measure quantifies how well a system functions. Performance measures should be measurable (computable if not actually observable), understandable, valid and reliable, and responsive to changes in program

operations. Operational modeling of public programs can even yield performance measures not apparent a priori. For example, the evaluation of the New Haven needle exchange program involved a mathematical model of HIV transmission among drug injectors as modified by the operations of needle exchange (Kaplan and O'Keefe 1993). The model revealed needle circulation time, that is, the amount of time a needle is available for use by drug injectors, as a critical performance measure. Reducing needle circulation time reduces opportunities for needle sharing on a per needle basis. This reduces both the chance that a needle becomes infected, and the chance that an injection with a used needle transmits infection. Needle exchange adjusts the distribution of needle circulation times. The model uncovered a direct link between the exchange of needles and the probability of HIV transmission.

Methodologies

Much of program evaluation is qualitative in nature. Social science methods relying on field observation, case histories, and the like are often used. However, such qualitative data often fail to satisfy critics of particular social programs. In addition, qualitative data generally allow only coarse judgments about program effectiveness. While no panacea, quantitative assessment methods have become standard in evaluating social programs and policies. Assessments of program effects are often made by statistically comparing a group participating in the program to a control group. The randomized experiment is the archetype for this kind of comparison. Since true randomized experiments may be difficult to execute under real program settings, quasi-experimental designs are often used instead. Rather than randomly assigning participants to program and control groups, quasi-experimental methods attempt to find natural or statistical controls. Multiple regression, analysis of variance, or other statistical techniques are often used; Cook and Campbell (1979) is a classic reference on quasi-experimental methods.

The model-based techniques of OR/MS are also applicable to program evaluation. Decision analysis is obviously useful in prospectively selecting among policy options. Queueing theory may be used to analyze the delivery of a wide range of programs, including public housing assignments, 911 hotlines, and dial-a-ride van services for the elderly and disabled. Applied probability models are generally useful, while statistical methods are widely valued. Techniques for multicriteria optimization, data envelopment analysis, and the analytical hierarchy process may be useful in identifying tradeoffs among multiple objectives.

While it seems that a solid understanding of OR/MS modeling is useful in conducting program evaluation, OR/MS has been underutilized. For example, basic optimization techniques such as linear programming have not been widely applied, perhaps because formulating a consensus objective function is usually very difficult. Training in OR/MS is less common than training in statistics and other social sciences. Few of those who have been trained in OR/MS have chosen to concentrate their efforts in the evaluation of public programs. Thus, social program evaluation remains an important and fertile area for further development and application of OR/MS methods.

Professional Opportunities and Organizations

Departments and agencies of federal, state, and municipal government and international organizations typically have offices that perform evaluation activities. Examples include the U.S. Environmental Protection Agency's Office of Policy Planning and Evaluation, the New York City Public School's Office of Research, Evaluation, and Assessment, and the World Bank's Operations Evaluations Unit. A few large private or non-profit organizations under-take many program evaluations. Among such organizations are The Urban Institute, Abt Associates, RAND Corporation, Mathematica Policy Research, and Westat. Much program evaluation is done by academics, largely social scientists. There are opportunities for OR/MS practitioners to get involved. One outlet is the INFORMS College on Public Programs and Processes. The American Evaluation Association is an interdisciplinary group of several thousand practitioners and academics. The journal *Evaluation Review* publishes examples of quality evaluations.

See

- Cost Analysis
- Cost-Effectiveness Analysis
- Emergency Services
- Practice of Operations Research and Management Science
- Problem Structuring Methods
- Public Policy Analysis
- RAND Corporation
- Systems Analysis
- Urban Services

References

Cicarelli, V. G., et al. (1969). *The impact of head start*. Athens, OH: Westinghouse Learning Corporation and Ohio University.

Cook, T. D., & Campbell, D. T. (1979). *Quasi-experimentation: Design and analysis issues for field settings*. Boston: Houghton Mifflin.

Kaplan, E. H., & O'Keefe, E. (1993). Let the needles do the talking! Evaluating the New Haven needle exchange. *Interfaces, 23*, 7–26.

Kelling, G. L., et al. (1974). *The Kansas city preventive patrol experiment: Summary report*. Washington, DC: The Police Foundation.

Larson, R. C. (1975). What happened to patrol operations in Kansas City? A review of the Kansas City Preventive Patrol Experiment. *Journal of Criminal Justice, 3*, 267–297.

Larson, R. C., & Kaplan, E. H. (1981). Decision-oriented approaches to program evaluation. *New Directions for Program Evaluation, 10*, 49–68.

Rossi, P. H., & Freeman, H. E. (1993). *Evaluation: A systematic approach* (5th ed.). New-bury Park, CA: Sage Publications.

Struyk, R. J., & Bendick, M., Jr. (1981). *Housing vouchers for the poor: Lessons from a national experiment*. Washington, DC: Urban Institute.

Program Evaluation and Review Technique (PERT)

A method for planning and scheduling a project which models uncertainties in activity by using optimistic, likely and pessimistic time estimates for each activity. PERT evolved when the U.S. Navy was developing a system to plan and coordinate the Polaris missile program (Malcolm et al. 1959).

See

- Critical Path Method (CPM)
- Network Planning
- Project Management
- Research and Development

References

Malcolm, D. G., Roseboom, J. H., Clark, C. E., & Fazar, W. (1959). Application of a technique for research and development program evaluation. *Operations Research, 7*, 646–669.

Project Management

Mark Westcombe and Graham K. Rand
Lancaster University, Lancaster, UK

Project management means different things to different people. Traditionally the domain of engineering, it has concerned itself with managing anything from small construction developments to large complex systems integration projects in defense, aerospace and other industries. A comprehensive survey of the development of project management since the 1940s and the issues involved in accomplishing projects is available in *The Management of Projects* (Morris 1997). In this period, OR/MS almost exclusively focused on the technical aspects of conforming to a contract using the iron triangle paradigm of management: to deliver a project to a pre-defined specification, on time, with an efficient use of resources within budget and with attention to safety. It accepted the project focus as the activities associated with the project lifecycle: defining scope; the work breakdown of the project plan; scheduling these activities; estimating costs; allocating resources and monitoring and controlling progress. OR/MS interested itself predominantly with techniques such as Program Evaluation and Review Technique (PERT) and the Critical Path Method (CPM).

Project management has since become ubiquitous within commercial and public sector organizations having been used to deliver organizational change

(see Balogun et al. 2008). Businesses might now use project management discourse and techniques to manage anything from opening a new store to the acquisition, a merger with an international corporation or to complete an urban regeneration scheme. They may conceive projects, form project teams and appoint project managers to issues that previously would have been dealt with by managers responsible for day-to-day operations. A critique of this projectification of operational management is offered by Hodgson and Cicmil (2006).

This evolution of project management has led to new ways of thinking about projects (Winter and Szczepanek 2009) and the focus of the project manager is now more concerned with defining project success (Atkinson 1999), delivering long-term project outcomes and ensuring benefits that add value to an organization's operations (Cooke-Davies 2007). Similarly OR/MS is engaging more at a strategic level of projects, offering, in particular, ideas from systems thinking for the developing of processes rather than just techniques, such as for the project front-end (Winter 2009), negotiating project objectives amongst differing stakeholder perspectives and managing stakeholder relationships. OR/MS has also contributed significantly to the risk analysis of projects (Williams 1995) as risk management has come to the fore, including: mathematical modeling (Chapman and Ward 2002); qualitative modeling of the systemic nature of risk (Ackerman et al. 2007); the cost impact of disrupted learning curves (Howick and Eden 2001); and the use of system dynamics to model disruption and delay of projects in litigation (Eden et al. 2000). It has also concerned itself with project selection, Monte Carlo simulation of projects and project portfolio management.

Outside of OR/MS, topics of current concern include: project evaluation and improvement; strategic alignment; organizational learning; program management; project leadership; sustainability issues; partnering; project governance; and procurement (see Crawford et al. 2006). A special issue of the *International Journal of Project Management* is of particular interest (Winter et al. 2006), which reviews future trends in the field as well as explores key contemporary themes in depth. A comprehensive breakdown of all the tactical elements of project management can be found in the professional Bodies of Knowledge (Association of Project Management 2006; Project Management Institute 2008), as well as from the growing industry of professional courses and certification in project management, such as PRINCE2, which is widely used in UK public sector projects and offers a particular step by step approach to project management.

Professional association in project management is available through the Association of Project Management, Ibis House, Regent Park, Summerleys Road, Princes Risborough, Buckinghamshire, UK HP27 9LE, which publishes *The International Journal of Project Management*; and the Project Management Institute, which publishes the *Project Management Journal*. Note that the term project management, or project management skills, is often misleadingly appropriated as a term in personal development to cover such transferable skills as time management, prioritization, presentation skills, etc.

See

- ▶ Critical Path Method (CPM)
- ▶ Network Planning
- ▶ Practice of Operations Research and Management Science
- ▶ Program Evaluation and Review Technique (PERT)

References

Ackerman, F., Eden, C., Williams, T., & Howick, S. (2007). Systemic risk assessment: a case study. *Journal of the Operational Research Society, 58*, 39–51.

Association of Project Management. (2006). *APM body of knowledge*. High Wycombe, Buckinghamshire: Author.

Atkinson, R. (1999). Project management: Cost, time and quality, two best guesses and a phenomenon, it's time to accept other success criteria. *International Journal of Project Management, 17*, 337–342.

Balogun, J., Hailey, V. H., Johnson, J., & Scholes, K. (2008). *Exploring strategic change*. London: FT Prentice Hall.

Chapman, C., & Ward, S. (2002). *Managing project risk and uncertainty: A constructively simple approach to decision making*. London: Wiley.

Cooke-Davies, T. (2007). Managing benefit. In J. R. Turner (Ed.), *Gower handbook of project management* (pp. 245–259). Aldershot: Gower.

Crawford, L., Pollack, J., & England, D. (2006). Uncovering the trends in project management: Journal emphases over the last 10 years. *International Journal of Project Management, 24*, 175–184.
Eden, C. E., Williams, T. M., Ackermann, F. A., & Howick, S. (2000). On the nature of disruption and delay (D&D). *Journal of the Operational Research Society, 51*, 291–300.
Hodgson, D. E., & Cicmil, S. (2006). *Making projects critical*. Basingstoke: Palgrave.
Howick, S. M., & Eden, C. (2001). The impact of disruption and delay when compressing large projects: Going for incentives? *Journal of the Operational Research Society, 52*, 26–34.
Morris, P. W. C. (1997). *The management of projects*. London: Thomas Telford.
Project Management Institute. (2008). *A guide to the project management body of knowledge*. Newtown Square, PA: Author.
Williams, T. M. (1995). A classified bibliography of research relating to project risk. *European Journal of Operational Research, 85*, 18–38.
Winter, M. (2009). Using soft systems methodology to structure project definition. In T. M. Williams, K. Samset, & K. J. Sunnevåg (Eds.), *Making essential choices with scant information: Front-end decision-making in major projects* (pp. 125–144). London: Palgrave Macmillan.
Winter, M., Smith, C., Morris, P., & Cicmil, S. (2006). Directions for future research in project management: The main findings of a UK government-funded research network. *International Journal of Project Management, 24*, 638–649.
Winter, M., & Szczepanek, T. (2009). *Images of projects*. Farnham, Surrey: Gower.

Project SCOOP

Project SCOOP (Scientific Computation of Optimal Programs) was a research program of the U.S. Air Force from the late 1940s to early 1950s whose main objective was to study and solve Air Force programming and scheduling problems. It was while working on Project SCOOP problems that George B. Dantzig formulated the linear-programming model and developed the simplex method for solving such problems.

Projection Matrix

For a given matrix $\boldsymbol{A}$, its associated projection matrix is defined as $\boldsymbol{P} = \boldsymbol{A}(\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$. The matrix $\boldsymbol{P}$ projects any vector $\boldsymbol{b}$ onto the column space of $\boldsymbol{A}$.

See

► Matrices and Matrix Algebra

Proper Coloring

An assignment of colors to nodes in a graph in which adjacent nodes are colored differently.

See

► Graph Theory

Prospect Theory

A descriptive theory of decision making under uncertainty (human choice), which attempts to explain certain deviations of observed empirical behavior from expected utility theory.

See

► Choice Theory
► Utility Theory

References

Kahneman, D., & Tversky, A. (1979). Prospect theory: An analysis of decision making under risk. *Econometrica, 47*, 263–289.

Protocols

The elicitation of an expert's procedure by asking the expert to describe aloud how he or she is solving a problem, such as making a forecast or a decision.

See

- ▶ Artificial Intelligence
- ▶ Expert Systems
- ▶ Forecasting

Pseudoconcave Function

Given a differentiable function $f(\cdot)$ on an open convex set X, the function f is pseudoconcave if $f(\boldsymbol{y}) > f(\boldsymbol{x})$ implies that $(\boldsymbol{y} - \boldsymbol{x})^T \nabla f(\boldsymbol{x}) > 0$ for all $\boldsymbol{x}, \boldsymbol{y} \in X$ where $\boldsymbol{x} \neq \boldsymbol{y}$.

See

- ▶ Concave Function
- ▶ Quasi-Concave Function

Pseudoconvex Function

Given a differentiable function $f(\cdot)$ on an open convex set X, the function f is pseudoconvex if $-f$ is pseudoconcave.

See

- ▶ Convex Function
- ▶ Pseudoconcave Function
- ▶ Quasi-Convex Function

Pseudoinverse

- ▶ Matrices and Matrix Algebra

Pseudorandom Numbers

A sequence of values coming from a mathematical algorithm, which appears to be statistically drawn independently from a uniform distribution over the unit interval [0,1].

See

- ▶ Random Number Generators

Pseudo-Polynomial-Time Algorithm

An algorithm whose running time is technically not polynomial because it depends on the magnitudes of the numbers involved, rather than their logarithms.

See

- ▶ Computational Complexity

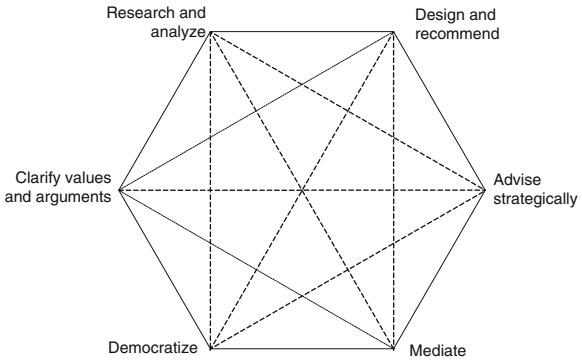
Public Policy Analysis

Warren E. Walker[1] and Gene H. Fisher[2]
[1]Delft University of Technology, Delft, The Netherlands
[2]RAND Corporation, Santa Monica, CA, USA

Introduction

Public policy analysis refers to the activities, methods, and tools that are used to give aid, advice, and support in the context of public policymaking. It covers a wide range of activities conducted with differing primary objectives and perspectives. Mayer et al. (2004) introduced a conceptual framework – the hexagon framework – that classifies the policy analysis activities in a structured manner. According to the hexagon framework, an analyst providing policy support may carry out six major clusters of activities, each having different objectives. The six objectives, represented as the vertices of the hexagon given in Fig. 1, are:

- *Research and analyze*: This type of activity aims for the generation of knowledge that can be used later for policy purposes. The major objective is to understand certain policy-relevant phenomena, and develop insights about them.
- *Design and recommend*: In certain situations the analyst can assist the decision-making process by designing alternative solutions to a problem and analyzing and possibly weighing the consequences



Public Policy Analysis, Fig. 1 Overview of objectives of policy analysis (Mayer et al. 2004)

of these alternative solutions. The main question here is more about evaluating a set of interventions, or changing the system that is related to the already known phenomena. In other words, there is a certain action orientation that ends with a policy choice or recommendation.

- *Provide strategic advice*: In certain situations, an analysis can be a strategic, client-oriented activity. The analyst can advise the client on the most effective strategy for achieving certain goals given a certain political constellation, i.e., the environment in which the client operates, the likely counter-steps of opponents, etc.
- *Mediate*: A given policy problem generally involves multiple parties that have different views and perspectives regarding the issue. Addressing the problem and coming up with an effective (i.e., accepted by all parties) policy may require the understanding of the other parties' perspectives. Hence, the task for the policy analyst may be mediating these multiple parties and promoting communication among them within a policymaking or decision-making process.
- *Democratize*: This type of policy-analytic activity aims mainly at acquiring and maintaining the involvement of all related parties in the policy process in order to make it as democratic as possible. This includes assuring the flow of proper information to all stakeholders, and the provision of opportunities for them to have their say regarding the policy issue.
- *Clarify arguments and values*: The main objective of this type of policy analysis activity is the elicitation of mindset, norms, and values of the stakeholders involved in the problem at hand. In these situations, the analyst can support or help move forward the decision-making process by analyzing the values and argumentation systems that underpin the social and political debate.

In real-life cases and projects, a policy analyst will combine one or more of these activities, albeit not all at the same time. Traditional policy analysis is focused on the 'design and recommend' vertex (see Walker 2000). The approach related to this objective is detailed below, and expanded upon in Thissen and Walker (2013). Its primary purpose is to *assist* policymakers in choosing a preferred course of action to implement in a *complex* system from among multiple alternatives under *uncertain* conditions.

The word "assist" emphasizes that policy analysis is used by policymakers as a decision aid, just as checklists, advisors, and horoscopes can be used as decision aids. Policy analysis is not meant to replace the judgment of the policymakers (any more than an X-ray or a blood test is meant to replace the judgment of medical doctors). Rather, the goal is to provide a better basis for the exercise of that judgment by helping to clarify the problem, presenting the alternatives, and comparing their consequences in terms of the relevant costs and benefits.

The word "complex" means that the system being studied contains so many variables, feedback loops, and interactions that it is difficult to project the consequences of a policy change. Also, the alternatives are often numerous, involving mixtures of different technologies and management policies, and producing multiple consequences that are difficult to anticipate, let alone predict.

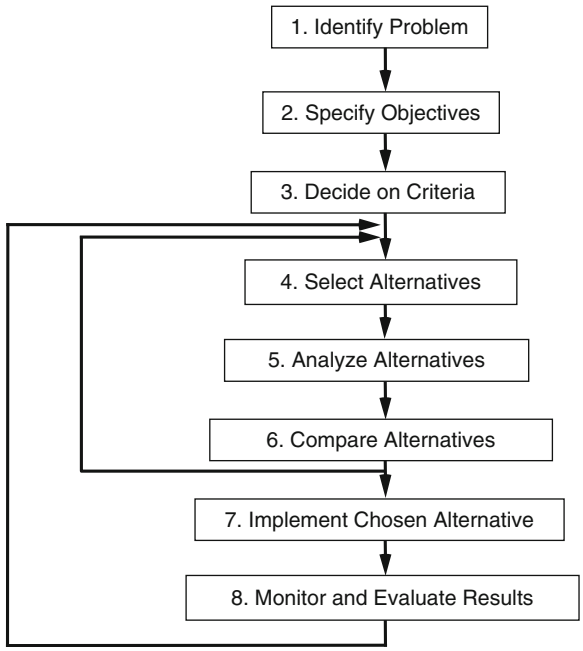
The word "uncertain" emphasizes that the choices must be made on the basis of incomplete knowledge about (a) the future world, (b) the model of the relevant system for that future world, (c) the outcomes from the system, and (d) the weights that the various stakeholders will put on the outcomes. This situation is sometimes referred to as "deep uncertainty".

Policy analysis is performed in government, at all levels; in independent policy research institutions, both for-profit and not-for-profit; and in various consulting firms. It is not a way of solving a specific problem, but is a general approach to problem solving. It is not a specific methodology, but it makes use of variety of methodologies in the context of a generic framework.

The Policy Analysis Steps

The policy analysis process generally involves performing the same set of logical steps, not always in the same order (Walker 2000; Miser and Quade 1985, p. 123). The steps, summarized in Fig. 2, are:

1. *Identify the problem*. This step sets the boundaries for what follows. It involves defining the system of interest, identifying the questions or issues involved, fixing the context within which the issues are to be analyzed and the policies will have to function (this is often done by using "scenarios"), clarifying constraints on possible courses of action, identifying the people who will be affected by the policy decision (the "stakeholders"), and discovering the major operative factors.
2. *Identify the objectives of the new policy*. Loosely speaking, a policy is a set of actions taken to solve a problem. The policymaker(s) and stakeholders have certain objectives that, if met, would "solve" the problem. In this step, the policy objectives are determined. (Most public policy problems involve multiple objectives, some of which conflict with others.)
3. *Decide on criteria (measures of performance and cost) with which to evaluate alternative policies*. Determining the degree to which a policy meets an objective involves measurement. This step involves identifying consequences of a policy that can be measured (either quantitatively or qualitatively) and that are directly related to the objectives. It also involves identifying the costs (negative benefits) that would be produced by a policy, and how they are to be measured.
4. *Select the alternative policies to be evaluated*. This step specifies the policies whose consequences are to be estimated. It is important to include as many as stand any chance of being worthwhile. If a policy is not included in this step, it will never be examined, so there is no way of knowing how good it may be. The current policy should be included as the "base case" in order to determine how much of an improvement can be expected from the other alternatives.
5. *Analyze each alternative*. This means determining the consequences that are likely to follow if the alternative is actually implemented, where the consequences are measured in terms of the criteria chosen in Step 3. This step usually involves using a model or models of the system.
6. *Compare the alternatives in terms of projected costs and benefits*. This step involves ranking the alternatives in order of desirability and choosing the one preferred. If none of the alternatives examined so far is good enough to be implemented (or if new aspects of the problem have been found, or the analysis has led to new alternatives), return to Step 4.
7. *Implement the chosen alternative*. This step involves obtaining acceptance of the new procedures (both within and outside the government), training people to use them, and performing other tasks to put the policy into effect.
8. *Monitor and evaluate the results*. This step is necessary to make sure that the policy is actually accomplishing its intended objectives. If it is not, the policy may have to be modified or a new study performed.



Public Policy Analysis, Fig. 2 Steps in a policy analysis study (Source: Walker 2000)

The individual steps in the process are described in detail by Miser and Quade (1985, Chap. 4), Quade (1989, Chap. 4), Walker (2000), and Enserink et al. (2010).

OR/MS and Public Policy Analysis

Policy analysis is closely related to operations research; in fact, in many respects it grew out of operations research as it was being applied at the RAND Corporation and other applied research organizations in the 1960s and 1970s. Miser (1980) and Majone (1985) describe this evolution. In the beginning, operations research techniques had been applied primarily to problems in which there were few parameters and a clearly defined single objective function to be optimized (e.g., aircraft design and placement of radar installations). Gradually, the problems being analyzed became broader and the contexts more complex. Health, housing, transportation, and criminal justice policies were being analyzed. Single objectives (e.g., cost minimization or single variable performance maximization) were replaced by the need to consider multiple (and conflicting) objectives (e.g., the impacts on health, the economy, and the environment and the distributional impacts on different social or economic groups). Non-quantifiable and subjective considerations had to be considered in the analysis (Schlesinger 1967, provided an early discussion of this issue). Optimization was replaced by satisficing.

Simon (1969, pp. 64–65) defined satisficing to mean finding an acceptable or satisfactory solution to a problem instead of an optimal solution. He said that satisficing was necessary because "in the real world we usually do not have a choice between satisfactory and optimal solutions, for we only rarely have a method of finding the optimum."

Operations research techniques are among the many tools in the policy analyst's took kit. The analyses and comparisons of alternative policies are usually carried out with the help of mathematical and statistical models. Simulation, mathematical programming, and queueing theory are among the many tools that are used in policy analysis study. But modeling is just one part of the process; all of the steps are important.

The policy analysis process has been applied to a wide variety of problems. Miser and Quade (1985, Chap. 3) provide examples of some of these, including improving blood availability and utilization, improving fire protection (for this, see also Walker et al. 1979), protecting an estuary from flooding, and providing energy for the future. More generally, the policy analysis approach has been used in the formulation of policies at the national level, including national security policies, transportation policies, and water management policies (e.g., Goeller and the PAWN Team 1985). Other examples that illustrate the approach can be found in a variety of publications, including Drake et al. (1972), House (1982), Mood (1983), Pollock et al. (1994), Miser (1995), and Walker et al. (2008).

See

- ▶ Choice Theory
- ▶ Cost Analysis
- ▶ Cost-Effectiveness Analysis
- ▶ Decision Analysis
- ▶ Decision Making and Decision Analysis
- ▶ Deep Uncertainty
- ▶ Exploratory Modeling and Analysis
- ▶ Multi-attribute Utility Theory
- ▶ Practice of Operations Research and Management Science
- ▶ RAND Corporation
- ▶ Satisficing
- ▶ Systems Analysis

References

Drake, A. W., Keeney, R. L., & Morse, P. M. (Eds.). (1972). *Analysis of public systems*. Cambridge, MA: MIT Press.

Enserink, B., Hermans, L., Kwakkel, J., Thissen, W., Koppenjan, J., & Bots, P. (2010). *Policy analysis of multi-actor systems*. Den Haag: Boom Lemma Publishers.

Findeisen, W., & Quade, E. S. (1985). The methodology of systems analysis: An introduction and overview, Chap. 4. In H. J. Miser & E. S. Quade (Eds.), *Handbook of systems analysis: Overview of uses, procedures, applications, and practice*. New York: Elsevier.

Goeller, B. F., & the PAWN Team. (1985). Planning the Netherlands' water resources. *Interfaces, 15*(1), 3–33.

House, P. W. (1982). *The art of public policy analysis, Sage Library of Social Research* (Vol. 135). Beverly Hills, CA: Sage Publications.

Majone, G. (1985). Systems analysis: A genetic approach, Chapter 2. In H. J. Miser & E. S. Quade (Eds.), *Handbook of systems analysis: Overview of uses, procedures, applications, and practice*. New York: Elsevier.

Mayer, I. S., van Daalen, C. E., & Bots, P. W. G. (2004). Perspectives on policy analyses: A framework for understanding and design. *International Journal of Technology, Policy and Management, 4*(2), 169–190.

Miser, H. J. (1980). Operations research and systems analysis. *Science, 209*, 139–146.
Miser, H. J. (Ed.). (1995). *Handbook of systems analysis: Cases*. Chichester: Wiley.
Miser, H. J., & Quade, E. S. (Eds.). (1985). *Handbook of systems analysis: Overview of uses, procedures, applications, and practice*. Chichester: Wiley.
Mood, A. M. (1983). *Introduction to policy analysis*. New York: North-Holland.
Pollock, S. M., Rothkopf, M. H., & Barnett, A. (Eds.). (1994). *Operations research and the public sector* (Handbooks in operations research and management science, Vol. 6). New York: North-Holland.
Quade, E. S. (1989). *Analysis for public decisions*. New York: Elsevier.
Schlesinger, J. R. (1967). *On relating non-technical elements to system studies, P-3545*. Santa Monica, CA: The RAND Corporation.
Simon, H. A. (1969). *The sciences of the artificial*. Cambridge, MA: MIT Press.
Thissen, W. A. H. & Walker, W. E. (2013). Public policy analysis: new developments. New York: Springer.
Walker, W. E. (2000). Policy analysis: A systematic approach to supporting policymaking in the public sector. *Journal of Multicriteria Decision Analysis, 9*(1–3), 11–27.
Walker, W. E., Chaiken, J. M., & Ignall, E. J. (Eds.). (1979). *Fire department deployment analysis: A public policy analysis case study*. New York: Elsevier North Holland.
Walker, W. E., van Grol, R., Rahman, S. A., van de Voort, M., Röhling, W., & Burg, R. (2008). Policy analysis of sustainable transport and mobility: The SUMMA project, Chapter 13. In A. Perrels, V. Himanen, & M. Lee-Gosselin (Eds.), *Building blocks for sustainable transport: Obstacles, trends, solutions* (pp. 73–102). Bingley, UK: Emerald Group.

Pull System

Production system in which work is released into the production facility based on the current state of the facility, which includes information such as available inventory, work in process, and realized demand.

See

- ▶ CONWIP
- ▶ Kanban
- ▶ Production Management

Pure-Integer Programming Problem

A mathematical programming problem in which all variables are restricted to be integer. Usually refers to a problem in which the constraints and the objective function are linear.

See

- ▶ Mixed-Integer Programming Problem (MIP)

Push System

Production system in which work is released into the system according to forecasted demand, usually based on a schedule prepared in advance.

See

- ▶ Production Management

Q

QC

Quality control; quality circles

See

▶ Quality Control

Q-Gert

Queue graphical evaluation and review technique.

See

▶ GERT
▶ Network Planning
▶ Project Management
▶ Research and Development

QP

▶ Quadratic Programming

Quadratic Assignment Problem

Alla Kammerdiner[1], Theodoros Gevezes[2], Eduardo Pasiliao[3], Leonidas Pitsoulis[2] and Panos M. Pardalos[4]
[1]New Mexico State University, Las Cruces, NM, USA
[2]Aristotle University of Thessaloniki, Thessaloniki, Greece
[3]Air Force Research Laboratory (AFRL) Munitions Directorate, Eglin Air Force Base, FL, USA
[4]University of Florida, Gainesville, FL, USA

Introduction

The quadratic assignment problem (QAP) is one of the most computationally challenging and well-known problems in the area of combinatorial and integer optimization. In general terms, also known as in the Lawler form (Lawler 1963), the QAP can be be stated as

$$\min_{\pi \in \Pi_n} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij\pi(i)\pi(j)} \tag{1}$$

where Π_n denotes the set of all possible permutations of n elements. In its original introduction due to Koopmans and Beckmann (1957), the statement of the QAP assumes that the cost coefficients c_{ijpq} have the following simple structure:

$$c_{ijpq} = \begin{cases} f_{ij} d_{pq} & \text{if } i \neq j \text{ or p} \neq \text{q}, \\ f_{ii} d_{kk} + a_{ik} & \text{otherwise.} \end{cases} \tag{2}$$

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

This less general version of the problem is known as the QAP in the Koopmans-Beckmann form.

In economic practice, the QAP arises naturally as a class of facility location and layout problems in the following way. Given

- two sets of the same cardinality n, namely, a set of objects (e.g., facilities) and a set of positions (e.g., locations),
- an $n \times n$ matrix $\boldsymbol{D} = \{d_{pq}\}$ of distances between two positions,
- an $n \times n$ matrix $\boldsymbol{F} = \{f_{ij}\}$ of flows from one object to another (e.g., amount of goods transferred from location i to location j), and
- an $n \times n$ matrix $\boldsymbol{A} = \{a_{ip}\}$ of costs of placing a specific object at a given location,

find a bijective assignment of the objects to the respective positions such that the total cost is minimized. In this context, it is commonly assumed that $\boldsymbol{F}$ and $\boldsymbol{D}$ are symmetric matrices with all zeros in the diagonal and nonnegative elements and that matrix $\boldsymbol{A}$ is also nonnegative.

In addition to the aforementioned area, the QAP has applications in various other fields as diverse as ergonomics, electronics and computer manufacturing, telecommunications, sports, archeology, and organic chemistry. In particular, the QAP can be used to place interconnected control and display devices on the backboard panel in such a manner that minimizes the total wire length; to design an ergonomic typewriter keyboard; to create an efficient layout for a hospital; to rank teams in a relay race; to analyze chemical reactions between various organic compounds; and to study archeological data. Other QAP applications include discriminative word alignment used in statistical machine translation systems, and symbol mapping diversity design for optimal retransmission of multiple data packets in wireless communications. For more information regarding applications, see the surveys by Burkard et al. (1999); Loiola et al. (2007); and Pitsoulis and Pardalos (2009).

Computational Complexity

The QAP is famous for its computational complexity, which may be one of the main reasons why the problem has received a considerable amount of attention from the operations research and management science research community. In fact, the QAP was shown by Sahni and Gonzalez (1976) to be strongly NP-hard. This means that the optimal solution for the QAP cannot be computed efficiently, and there are no computationally efficient algorithms that are able to find an approximate solution that is within some constant ratio to the optimal. Formally speaking, the QAP is an NP-hard problem (i.e., the existence of a polynomial time algorithms for solving the QAP implies $\mathcal{P} = \mathcal{NP}$), and furthermore, for an arbitrary $\varepsilon > 0$, a polynomial time ε-approximation algorithm for the QAP does not exist unless $\mathcal{P} = \mathcal{NP}$.

A stronger result obtained by Queyranne (1986) states that, unlike in the case of the traveling salesman problem (TSP), which is also NP-hard and represents a special case of the Koopmans-Beckmann QAP that is polynomially approximable within $3/2$ when the distance matrix is symmetric and satisfies the triangle inequality, there is no polynomial approximation algorithm (unless $\mathcal{P} = \mathcal{NP}$) for finding a feasible solution within some finite approximation ratio to the optimal, even for the QAPs in the Koopmans-Beckmann form with $\boldsymbol{D}$ representing distances of the set of points on a line, a symmetric block diagonal matrix F of flows, and zero linear terms (i.e., $\boldsymbol{A} = 0$). This result can be viewed as a reinforcement for a common belief that the QAP appears to be computationally more challenging as compared to other NP-hard problems in combinatorial and integer optimization. In addition to the TSP, other NP-hard problems such as the maximum clique problem, the subgraph isomorphism problem, the minimum weight feedback arc set problem, the linear arrangement problem, and the graph packing problem can be viewed as a special case of the QAP.

The problems of finding locally optimal solutions for several neighborhood structures (e.g., a Kernighan-Lin-type structure and a pair exchange or 2-opt structure) were investigated for the QAP. Similar to the way the polynomial time decision problems are said to be in $\mathcal{P}$, the problems of this kind, for which polynomial time algorithms exist, are known as the polynomial time local search (PLS) problems. By analogy to NP-complete problems, the PLS-complete problems are the most computationally challenging among the PLS problems. Both Kernighan-Lin-type neighborhoods for the QAP and a commonly used 2-opt neighborhood structure are shown to be the PLS-complete. Burkard et al. (1999) state that there are no known local search criteria for comparing

the quality of the locally optimal solution to the global solution. Furthermore, no computationally efficient algorithms exist for deciding whether a locally optimal solution is also globally optimal for a given QAP instance.

Since in general the QAP is NP-hard and non-approximable, it is of special interest to find some more restricted versions of this problem for which computationally efficient algorithms exist. The motivation behind this is a natural expectation that a better understanding of what makes the QAP so computationally difficult in its general case may be achieved by comparison of the general QAP structure with the problem structure in the polynomially-solvable versions. The approaches commonly used to find new polynomially-solvable cases of the QAP can be divided into two categories:

1. Determine a suitable set of assumptions to be imposed on the structure of coefficient matrices of the QAP in order to get polynomially-solvable instances of the QAP.
2. Analyze other relevant problems that can be viewed a special case of the QAP (e.g., the TSP or the linear arrangement problem) in order to identify their polynomially-solvable versions. Then reformulate these polynomially-solvable versions of the other relevant problems as the QAP instances.

For further details on the computational complexity of the QAP (including discussions of the PLS complexity issues and the polynomially-solvable special cases) see, e.g., Chapters 3 and 10 in Burkard et al. (1999), and Chapter 3 in Pitsoulis and Pardalos (2009).

Formulations

Similarly to many other problems in combinatorial optimization (CO), the QAP can be equivalently reformulated using several alternative representations. Besides the purely combinatorial formulation (1) that represents feasible solutions of the QAP as permutations, the problem can be written as a quadratic integer programming program and as a concave quadratic optimization problem. The QAP can also be given a trace formulation. Alternatively, it can be formulated using the Kronecker product. Moreover, there is a graph-theoretic formulation of the QAP, just like there is one for the linear assignment problem (LAP). Each of these alternative formulations brings with itself a unique set of solution methods and analytical techniques developed for that specific area. For instance, the graph reformulation allows for application of graph-theoretic reasoning to the QAP, whereas a reformulation using semidefinite programming relaxation leads to very different lower bounds than the Gilmore-Lawler-type lower bounds, which are based on the Kronecker product formulation. This section briefly describes some of the most commonly used QAP formulations. For more information on various QAP formulations, see, e.g., Pardalos and Wolkowicz (1994); Burkard et al. (1999); and Loiola et al. (2007).

Since every permutation $\pi \in \Pi_n$ of the n-element set can be equivalently represented by an $n \times n$ 0-1 matrix $\boldsymbol{X} = \{x_{ip}\}$, which is called a permutation matrix and has the following elements:

$$x_{ip} = \begin{cases} 1 \text{ if } \pi(i) = p \\ 0 \quad \text{otherwise,} \end{cases} \tag{3}$$

then the QAP (1) can be stated as the following quadratic 0-1 integer programming problem:

$$\min \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{p=1}^{n} \sum_{q=1}^{n} c_{ijpq} x_{ip} x_{jq} \tag{4}$$

$$\text{subject to} \quad \sum_{\mathrm{i}=1}^{\mathrm{n}} x_{ip} = 1, \quad p = 1, 2, \ldots, n, \tag{5}$$

$$\sum_{p=1}^{n} x_{ip} = 1, \quad i = 1, 2, \ldots, n, \tag{6}$$

$$x_{ip} \in \{0, 1\}, \; i, p = 1, \ldots, n. \tag{7}$$

The above formulation very naturally facilitates linearization of the quadratic objective function through introduction of new 0-1 integer variables, and so the application of this formulation is a common first step in solving the QAP using several linearization techniques that are discussed in more detail later. Additionally, by defining an inner product of matrices $\boldsymbol{F} = \{f_{ij}\}$ and $\boldsymbol{Y} = \{y_{ij}\}$ as

$$\langle \boldsymbol{F}, \boldsymbol{Y} \rangle = \sum_{i=1}^{n} \sum_{j=1}^{n} f_{ij} y_{ij},$$

and then observing that for any $n \times n$ permutation matrix $\boldsymbol{X}$ corresponding to $\pi \in \Pi_n$ and any $n \times n$ matrix $\boldsymbol{D}$, the following is true:

$$\boldsymbol{XDX^T} = \left\{ \sum_{p=1}^{n} \sum_{q=1}^{n} x_{ip} d_{pq} x_{jq} \right\} = \left\{ d_{\pi(i)\pi(j)} \right\},$$

the QAP in the Koopmans-Beckmann form is compactly written as the following quadratic integer program:

$$\begin{aligned} \min \quad & \left\langle \boldsymbol{F}, \boldsymbol{XDX^T} \right\rangle + \langle \boldsymbol{A}, \boldsymbol{X} \rangle \\ \text{subject to} \quad & \boldsymbol{X} \in \mathbf{X}_n, \end{aligned} \tag{8}$$

where $\mathbf{X}_n$ denotes a set of all possible $n \times n$ permutation matrices, i.e., matrices that satisfy conditions (5), (6), and (7).

Since a number of problems on graphs, including the subgraph isomorphism and the graph partitioning problems, can be viewed as special cases of the QAP, it comes as no surprise that the QAP itself can be given a graph-based formulation. Specifically, given two undirected weighted complete graphs G_F and G_D both having n vertices and $n(n-1)/2$ edges with the weights represented by matrices F and D, respectively, which are symmetric and have all zeros on the diagonal, then a Koopmans-Beckmann QAP with zero linear terms can be formulated as the problem of aligning the vertex sets of the two graphs in such a way as to minimize the sum of products of the aligned edges. As noted in Loiola et al. (2007), this graph formulation of the QAP also leads to the characterization of the feasible solutions as the line-graph automorphisms of the complete graph K_n on n vertices. For a general graph-based formulation of a Lawler QAP, see, e.g., Kaibel (2000).

A very convenient cost structure of a Koopmans-Beckmann QAP also permits its alternative reformulation via trace function. Indeed, since the trace of an $n \times n$ matrix $\boldsymbol{A}$ is defined as:

$$\mathrm{tr}(\boldsymbol{A}) = \sum_{i=1}^{n} a_{ii}$$

then the QAP in Koopmans-Beckmann form can be written as:

$$\begin{aligned} \min \quad & \mathrm{tr}\left(\left(\boldsymbol{FXD^T} + \boldsymbol{A} \right) \boldsymbol{X^T} \right) \\ \text{subject to} \quad & \boldsymbol{X} \in \mathbf{X}_n. \end{aligned} \tag{9}$$

The trace formulation can be utilized to introduce eigenvalue-based lower bounds for the QAP instances, where at least one of the matrices $\boldsymbol{F}$ and $\boldsymbol{D}$ is symmetric. In such case, the nice properties of the trace function actually allow transforming the problem into a QAP where both matrices become symmetric. For example, if F is symmetric and $\boldsymbol{D}$ is not, then introducing a new $\overline{\boldsymbol{D}} = \frac{1}{2}(\boldsymbol{D} + \boldsymbol{D^T})$ does the trick.

The QAP formulation, originally due to Bazaraa and Sherali (1982), as the following quadratic concave minimization problem:

$$\begin{aligned} \min \quad & \boldsymbol{x^T Q x} \\ \text{subject to} \quad & \boldsymbol{x} = \mathrm{vec}(\boldsymbol{X^T}),\ \boldsymbol{X} \in \mathbf{X}_n, \end{aligned} \tag{10}$$

(where $\boldsymbol{Q}$ is an $n^2 \times n^2$ symmetric, negative definite matrix) can be used in cutting plane procedures. Notice that $\boldsymbol{x}$ in (10) represents an $n^2 \times 1$ vector, which is formed by arranging all the rows of some $n \times n$ permutation matrix $\boldsymbol{X}$ successively into a single row of size n^2 and then transposing this long row into a n^2-dimensional column vector. The reformulation (10) is obtained from (4) by first combining the cost coefficients c_{ijpq} into an $n^2 \times n^2$ matrix $\boldsymbol{B} = \{b_{rs}\}$ with $b_{rs} = c_{ijpq}$, for $r = (i-1)n + p$, $s = (j-1)n + q$, then transforming $\boldsymbol{X}$ into $\boldsymbol{x}$ as described above, and lastly defining the $n^2 \times n^2$ matrix $\boldsymbol{Q} = \boldsymbol{B} - \beta \boldsymbol{I}$, where $\boldsymbol{I}$ denotes an $n^2 \times n^2$ identity matrix and β is a positive real number such that $\beta > \max\left\{ \sum_{s=1}^{n} |b_{rs}|,\ 1 \le r \le n^2 \right\} =: \| \boldsymbol{B} \|_\infty$.
Analogously, the QAP can be easily formulated as a quadratic convex minimization problem, by simply adding the $\beta \boldsymbol{I}$ term to $\boldsymbol{B}$ instead of subtracting it.

By introducing an $n^2 \times n^2$ cost matrix $\boldsymbol{B}$ (defined above), using the inner product of matrices, and incorporating the Kronecker product into constraints of the problem, the QAP in the Lawler form can be compactly written as:

$$\begin{aligned} \min \quad & \langle \boldsymbol{B}, \boldsymbol{Z} \rangle \\ \text{subject to} \quad & \boldsymbol{Z} = \boldsymbol{X} \otimes \boldsymbol{X},\ \boldsymbol{X} \in \mathbf{X}_{\boldsymbol{n}}, \end{aligned} \tag{11}$$

where the Kronecker product of $n \times m$ matrix $\boldsymbol{X}$ and $p \times q$ matrix $\boldsymbol{Y}$ is given by an $np \times mq$ matrix

$$\boldsymbol{X} \otimes \boldsymbol{Y} = \begin{bmatrix} x_{11}\boldsymbol{Y} & \dots & x_{1m}\boldsymbol{Y} \\ \vdots & \ddots & \vdots \\ x_{n1}\boldsymbol{Y} & \dots & x_{nm}\boldsymbol{Y} \end{bmatrix}.$$

The formulation (11) represents the QAP as an LAP of size n^2 with the additional quadratic constraint that an $n^2 \times n^2$ permutation matrix $\boldsymbol{Z}$ must be a Kronecker product of a permutation matrix $\boldsymbol{X}$ with itself. Because of the new constraint, the resulting problem cannot be solved efficiently. The trace and the Kronecker product are also used in semidefinite programming relaxations of the QAP, discussed below.

Relaxations

Two approaches commonly used to tackle the QAP are linear programming (LP) and semidefinite programming (SDP) relaxations, which attempt to address the difficulties associated with the quadratic objective function by reformulating the problem based either on linearization techniques or by SDP-based transformations of "stretching" and "lifting" the original matrices into the Euclidean space of high-dimensional square matrices. The resulting significant increase in dimensionality of the problem is the common trade-off with these approaches. In particular, the consequent high dimensionality of the linearized versions of the QAP represents a considerable challenge for efficiently solving the corresponding linear programs. However, the relaxations of the linearized versions or the SDP-based reformulations (especially for QAPs with additional constraints on the costs structure) can be solved to compute lower bounds for the QAP. Therefore, better lower bounds are likely to be produced with a tighter relaxation.

The four fundamental QAP linearization techniques are given by Lawler (1963), Kaufmann and Broeckx (1978), Frieze and Yadegar (1983), and Adams and Johnson (1994). The Lawler linearization represents the QAP as the following 0-1 integer LP with $n^4 + n^2$ variables and $n^4 + 2n^2 + 1$ constraints:

$$\min \sum_{i,j,p,q} c_{ijpq} z_{ijpq} \tag{12}$$

$$\begin{aligned} \text{subject to } & \{x_{ij}\} \in \boldsymbol{X}_n, \\ & \sum_{i,j,p,q} z_{ijpq} = n^2 \\ & x_{ij} + x_{pq} - 2z_{ijpq} \geq 0, \ \forall i,\ j,\ p,\ q, \\ & z_{ijpq} \in \{0,1\}, \ \forall i,j,p,q, \end{aligned}$$

by replacing all the products of the form $x_{ij}x_{pq}$ in (4) with new binary variables $z_{ijpq} = x_{ij}x_{pq}$. Understandably, a large number of variables and constraints in (12) is a disadvantage of this linearization.

Kaufmann and Broeckx (1978) proposed a linearization with the smallest number of variables and constraints. In their linearization, the QAP is equivalently expressed as the following mixed integer linear programming (MILP) problem with n^2 real variables, n^2 binary variables and $n^2 + n$ constraints:

$$\min \sum_{i,j} y_{ij} \tag{13}$$

$$\begin{aligned} \text{subject to } & \{x_{ij}\} \in \mathbf{X}_n, \\ \forall i,j, \quad & \alpha_{ij}x_{ij} + \sum_{p,q} c_{ijpq}x_{pq} - y_{ij} \ \leq \ \alpha_{ij} \\ & y_{ij} \in \{0,1\}, \ \forall i,j, \end{aligned}$$

by introducing new real-valued variables $y_{ij} = x_{ij} \sum_{p,q=1}^{n} c_{ijpq}x_{pq}$ and constants $\alpha_{ij} = \sum_{p,q=1}^{n} c_{ijpq}$.

Similarly to the Lawler's technique, the linearizations by Frieze and Yadegar (1983) and by Adams and Johnson (1994) both introduce new variables $z_{ijpq} = x_{ij}x_{pq}$, but unlike in Lawler's case, the Frieze-Yadegar and the Adams-Johnson linearizations relax the integrality condition by allowing $\{z_{ijpq}\}$ to be real-valued. Therefore, the latter two linearizations result in MILP problems with the same objective function but different constraints. Specifically, the Frieze-Yadegar linearization reformulates the QAP as:

$$\min \sum_{\mathrm{i,j,p,q}} c_{ijpq} y_{ijpq} \tag{14}$$

$$\begin{aligned} \text{subject to } & \{x_{ij}\} \in \mathbf{X}_n, \\ & \sum_{i} y_{ijpq} = \sum_{i} y_{jipq} \\ & = \sum_{i} y_{pqij} = \sum_{i} y_{pqji} = x_{pq}, \ \forall j,p,q, \\ & y_{ijij} = x_{ij}, \ \forall i,j, \\ & 0 \leq y_{ijpq} \leq 1, \ \forall i,j, \end{aligned}$$

whereas the Adams-Johnson linearization rewrites the QAP as:

$$\min \sum_{\mathrm{i,j,p,q}} c_{ijpq} y_{ijpq} \tag{15}$$

$$\begin{aligned} \text{subject to} \quad & \{x_{ij}\} \in \mathbf{X}_n, \\ \forall j,p,q, \quad & \sum_i y_{ijpq} = \sum_i y_{jipq} = x_{pq} \\ & y_{ijpq} = y_{pqij}, \ \forall i,j,p,q, \\ & y_{ijpq} \geq 0, \ \forall i,j,p,q. \end{aligned}$$

As mentioned above, useful QAP relaxations can also be obtained using semidefinite programming (SDP). Interestingly, SDP can be thought as a generalization of LP with the variables that belong to the Euclidean space of matrices, where the inner product between the elements is given by the trace of the product of matrices. Interior point methods or cutting plane procedures are commonly applied for solving the SDP relaxations, and the obtained solutions represent valid lower bounds for the original QAP. In fact, the lower bounds produced by the SDP relaxations are typically rather strong, but they often are computationally very expensive due to high dimensionality of the problems resulting from the SDP relaxations. A thorough explanation of the application of SDP to the QAP can be found in Zhao et al. (1998), where it is shown that based on the trace formulation of the problem below:

$$\min \ \operatorname{tr}\left(\boldsymbol{FXDX}^T - 2\boldsymbol{AX}^T\right) \tag{16}$$

$$\begin{aligned} \text{subject to} \quad & \boldsymbol{XX^T} = \boldsymbol{X^TX} = \boldsymbol{I}, \\ & \boldsymbol{Xe} = \boldsymbol{X^Te} = \boldsymbol{e} = (1,\ldots,1)^T, \\ & x_{ij}^2 - x_{ij} = 0, \end{aligned}$$

the following SDP relaxation of the QAP can be obtained:

$$\min \ \operatorname{tr}(\boldsymbol{KY}) \tag{17}$$

$$\begin{aligned} \text{subject to} \quad & \boldsymbol{b}^0\mathrm{diag}(\boldsymbol{Y}) = \mathbf{o}^0\mathrm{diag}(\boldsymbol{Y}) = \boldsymbol{I}, \\ & \mathrm{arrow}(\boldsymbol{Y}) = \mathbf{e}_0, \\ & \operatorname{tr}(\boldsymbol{RY}) = 0, \\ & \boldsymbol{Y} \succcurlyeq 0, \end{aligned}$$

where

$$K = \left[0 \ \& \ -\mathrm{vec}(\boldsymbol{A})^T - \mathrm{vec}(\boldsymbol{A}) \ \& \ \boldsymbol{D} \otimes \boldsymbol{F}\right],$$

$$\boldsymbol{R} = \left[2n \ \& \ -2\boldsymbol{e}^T \otimes \boldsymbol{e}^T - 2\boldsymbol{e} \otimes \boldsymbol{e} \ \& \ \boldsymbol{I} \otimes \boldsymbol{E} + \boldsymbol{E} \otimes \boldsymbol{I}\right],$$

and $\mathbf{E}$ is an $n \times n$ matrix of all ones, $\preccurlyeq$ denotes the Löwner partial order (i.e., $\boldsymbol{A} \preccurlyeq \boldsymbol{B}$ if and only if $\boldsymbol{B} - \boldsymbol{A} \succcurlyeq 0$, which means that $\boldsymbol{B} - \boldsymbol{A}$ is positive semidefinite). Note that the rows and columns of an $(n^2+1) \times (n^2+1)$ matrix $\boldsymbol{Y}$ are assumed to be numbered starting from 0 to n^2. Then $\mathbf{b}^0\mathrm{diag}$ and $\mathbf{o}^0\mathrm{diag}$, respectively, are the block-0-diagonal and off-0-diagonal operators from the space of $(n^2+1) \times (n^2+1)$ matrices into the space of $n \times n$ matrices given by:

$$\mathrm{b}^0\mathrm{diag}(\boldsymbol{Y}) = \sum_{k=1}^{n} \boldsymbol{Y}_{(k,\cdot)(k,\cdot)},$$

$$\mathrm{o}^0\mathrm{diag}(\boldsymbol{Y}) = \sum_{k=1}^{n} \boldsymbol{Y}_{(\cdot,k)(\cdot,k)}.$$

Here $\boldsymbol{Y}_{(k,\cdot)(k,\cdot)}$ denotes the k-th $n \times n$ matrix on the diagonal of the $n \times n$ array of $n \times n$ matrices, all of which together with the first row $\boldsymbol{Y}_{(0,\cdot)}$ and the first column $\boldsymbol{Y}_{(\cdot,0)}$ compose the $(n^2+1) \times (n^2+1)$ original matrix $\boldsymbol{Y}$. In a similar fashion, $\boldsymbol{Y}_{(\cdot,k)(\cdot,k)}$ denotes an $n \times n$ matrix of the diagonal elements in the position (k,k) of the $n \times n$ matrices, which together form the $n^2 \times n^2$ lower right block of $\boldsymbol{Y}$. Also, arrow$(\cdot)$ in (17) denotes the arrow operator from the space of $(n^2+1) \times (n^2+1)$ matrices into the space of (n^2+1)-dimensional vectors defined as:

$$\mathrm{arrow}(\boldsymbol{Y}) = \mathrm{diag}(\boldsymbol{Y}) - \left(0, \boldsymbol{Y}_{(0,1:n^2)}\right),$$

where $\mathrm{diag}(\boldsymbol{Y}) = \left(\mathbf{y}_{00},\ldots,\mathbf{y}_{\mathrm{n^2\,n^2}}\right)^{\mathrm{T}}$ and $\left(0, \boldsymbol{Y}_{(0,1:n^2)}\right) = (0, y_{01},\ldots,y_{0n^2})^T$ is an (n^2+1)-dimensional vector with zero in the first position and the rest of the elements given by the elements in the first row (i.e., row 0) of $\boldsymbol{Y}$ starting from the position 1 all the way to the last position n^2. Finally, $\boldsymbol{e}_0 = (1,0,\ldots,0)^T$ is a (n^2+1)-dimensional unit vector with a one in the first dimension (i.e., row 0).

For formulations of many additional SDP relaxations, theoretical relationships between them and

a general approach of devising strong SDP relaxations for the QAP and other CO problems using a Lagrangian framework, see, e.g., the survey by Roupin (2009).

Polytopes

Although polyhedral combinatorics has been around since the 1950s and proved quite successful in application to many NP-hard problems in combinatorial optimization (CO), including the TSP, the stable set problem, and the maximum cut problem, polyhedral theory was not applied to the QAP until the 1990s (Kaibel 2000). Generally speaking, polyhedral combinatorics is concerned with understanding the geometric structure imposed by the constraints of CO problems with linear objective functions. Moreover, polyhedral methods can be extended to study the linearized versions of CO problems with nonlinear objectives, such as the QAP.

Since the space of all feasible solutions of a CO problem is represented geometrically as a convex hull called the associated polytope of the problem, and because the polytopes can equivalently be described via the finite systems of inequalities that produce bounded solution spaces and are referred as linear descriptions, polyhedral combinatorics can basically be thought of as the branch of CO that is concerned with finding such linear descriptions for the polytopes arising from CO problems (Kaibel 2000). Investigation of the linear descriptions helps gain structural insights into the corresponding problems and their solution algorithms. Furthermore, by finding a complete linear description for a CO problem and then proving that the so-called separation problem for that linear description is solvable in polynomial time, one can prove that the correspondent problem is in P. Of course, because the QAP is well-known to be NP-hard, instead of searching for a complete linear description, which is unlikely to exist, the focus of the polyhedral methods for the QAP is on finding its partial linear descriptions.

Partial linear descriptions of polytopes based on the linearized versions of the QAP can be used in particular to compute lower bounds on the optimal solution value of an instance via cutting plane algorithms. Obviously, for reasons of efficiency of cutting plane algorithms, it is important that the obtained partial linear descriptions are non-redundant, i.e., no inequality in such systems can be represented as a linear function of the other inequalities with non-negative weights. In particular, this leads to the questions about the polytope's dimension and the dimensional gap. Other important characteristics of the polytopes are its faces of different dimensions, including vertices, edges, ridges, and facets.

As Kaibel (2000) shows, a convenient way to derive a number of important results regarding the polytopes associated with the QAP is by considering a graph formulation of the QAP. In particular, this approach can be used to derive a characterizations for the vertices of the symmetric QAP polytope and the polytope of a general QAP with m objects and n locations. Also the graph-based formulation can be utilized to establish an affine isomorphism between the QAP polytopes for the QAP with $n-1$ objects and n locations and the QAP with n objects and n locations. An analogous result for the symmetric QAP polytopes are also true.

Several interesting results have already been established about the connection between the QAP polytopes and other polytopes (Kaibel 2000). The polytope of a general QAP with m objects and n locations can be thought as a specific face of a cut polytope. This important fact allows utilizing existing extensive knowledge regarding the structure of the cut polytopes. Not very surprisingly (since the TSP is a certain special case of the QAP), the TSP polytope can be viewed as a simple orthogonal projection of the QAP with the equal number of objects and locations. A similar fact is also true for the linear ordering polytope. The change of coordinate representation, called the star-transformation, has been proved exceptionally useful for reducing the dimensional gap and for proving the results about facial descriptions of the QAP polytopes, the polytopes' dimensions, and the so-called box-inequalities for the QAP.

Also it is noteworthy that there exists an interesting connection between the QAP polytopes and the representation theory of the symmetric group, which was exploited to obtain one of the earliest results on the dimension of the QAP polytope. For further references, formulations of theoretical results, and in-depth discussion of the subject of the polyhedral methods for the QAP, see, e.g., the survey by Kaibel (2000).

Lower Bounds

Lower bounds play a fundamental role in combinatorial and integer optimization in general, and are especially useful for solving the QAP. Not only do they serve as a key instrument in evaluating the quality of solutions obtained by heuristic algorithms (i.e., procedures that do not necessarily return optimal solutions), but also they can be very useful in reducing the search space of the implicitly enumerative algorithms (e.g., branch-and-bound-type procedures).

In terms of usability, the two crucial characteristics of lower bounds for the QAP are the tightness of a bound and the computational efficiency. As applied to implicit enumeration methods, lower bounds can provide a guarantee that a given subset of the solution space does not contain the global optimal solution and therefore does not need to be searched. Hence, given that a stronger lower bound lies closer to the optimum, such a lower bound would likely be capable of producing a greater reduction of the search space. Similarly, in application to heuristics, stronger lower bounds are able to provide a better estimate of the quality of the solution found by the heuristic. However, computing a strong lower bound for the QAP typically can be very challenging. For the general QAP, there seems to be a trade-off between the tightness and the computational efficiency of its lower bounds, although results by Mittelmann and Peng (2010) show that for the QAP instances that are based on the Hamming and Manhattan distance matrices, the SDP relaxation can produce lower bounds that are both strong and relatively easy to compute.

Lower bounds for the QAP can loosely be subdivided into the following seven categories:

- Gilmore-Lawler type lower bounds;
- Lower bounds based on LP relaxations;
- Lower bounds based on SDP relaxations;
- Variance reduction lower bounds;
- Eigenvalue-based lower bounds;
- Decomposition-based improved lower bounds for specially structured QAPs;
- Lower bounds based on polyhedral methods.

Gilmore-Lawler type lower bounds (GLTB) are derived using the formulation (11), which represents the QAP as an LAP of size n^2. To compute GLTB, a solution matrix $\boldsymbol{Z}$ is constructed by solving a sequence of LAPs. If the computed matrix $\boldsymbol{Z}$ is a permutation matrix, then it is also a feasible solution of the QAP, and therefore, $\boldsymbol{Z}$ gives the optimal solution of the QAP; otherwise the value of the optimal solution of the QAP is bounded from below by $\langle \boldsymbol{B}, \boldsymbol{Z} \rangle$.

Several lower bounds utilize this basic principle, including the original Gilmore-Lawler bound (GLB). This bound is simple to compute, but the gap between GLB and the QAP optimal value increases very quickly with the size of the problem. One way to improve the quality of GLB is by transforming a given problem so that some of the weight of the cost coefficients in the quadratic terms is moved into the corresponding linear cost coefficient. This can be achieved by means of reduction methods, which work by first suitably decomposing the cost coefficients in the quadratic term and then moving some of their value into the linear term, which, in fact, results in a stronger lower bound. An alternative approach for improving GLB is by using a reformulation. The idea behind this approach is to devise a series of successive reformulations constructed in such a way as to guarantee that the sequence of GLBs computed for the consecutive reformulations is monotonically nondecreasing. Although lower bounds based on reformulation procedures typically have good quality, the tradeoff is that they are time-consuming.

Some GLB-inspired approaches incorporate a dual formulation and LP relaxations. A number of these bounds are based on the reformulation-linearization (RLT) technique and are considered some of the tightest and most computationally efficient lower bounds for the QAP; see, e.g., Loiola et al. (2007). RLT procedures were originally introduced for solving mixed-integer 0-1 programming problems. For a given instance with n 0-1 variables, RLT constructs an n-level hierarchy of relaxations starting from the LP relaxation and ending with the convex hull of feasible solutions. This is essentially an iterative procedure that incorporates two key phases. In the first phase, the problem is reformulated by introducing redundant nonlinear constraints that are obtained by multiplying each of the defining constraints by the product factors. Next, in the linearization phase, each distinct variable in the nonlinear terms of the objective or constraints is replaced by the corresponding single continuous variable. As a result, the problem is formulated as mixed-integer 0-1 LP problem in

a higher-dimensional space. Furthermore, at each level of the hierarchy in RLT, the resulting continuous relaxation is at least as tight as the preceding one, and the highest n-th level gives the convex hull of the entire feasible region of the original QAP. Therefore, the facets of this hull in the variables of the original QAP can be obtained as projections of the final relaxation. Importantly, the RLT procedure is able to exploit the special block-diagonal structure of the Lagrangian dual. This allows to use the dual-ascent procedure to solve the dual of the relaxation and compute lower bounds.

RLT bounds can be viewed as the bounds obtained via the so-called lift-and-project method, which represents the QAP polytope as the projection of some higher-dimensional polytope. Other bounds that also utilize this general method are SDP-based lower bounds. The major difference between the RLT and SDP-based bounds in terms of their application of the lift-and-project method is that the SDP involves semi-definite relaxation in place of the linearization that is used in RLT.

The SDP-based bounds are among those with the smallest gap to the optimal value for many QAP instances from the QAP library (Burkard et al. 1997). Typically, the SDP lower bounds are computed by solving the SDP relaxation by means of cutting plane algorithms or interior point methods. Application of cutting plane algorithms for computing the SDP-based bounds often can speed up the computation, which allows to use the SDP in exact methods by computing bounds for weaker relaxations in a shorter time. Typically, the task of solving the SDP relaxations by interior point methods represents a significant computational challenge, although in some cases it may be possible to exploit group symmetry in the structure of QAP cost coefficients. In fact, using the representation theory of symmetric groups, it was shown that when the QAP coefficient matrices have sufficiently large automorphism groups, it becomes possible to solve the SDP relaxations using interior point methods and obtain extremely strong lower bounds. Roupin (2009) used the connections between partial Lagrangian and SDP relaxations of 0-1 quadratic programs to analytically compare some of the SDP-based lower bounds for the QAP and show theoretical equivalence between the R3 SDP relaxation by Rendl and Sotirov (2007) and the SDP relaxation in the lift-and-project procedure by Burer and Vandenbussche (2006). However, because of their dependence on SDP solvers, these two equivalent SDP relaxations produce lower bounds that differ in their usefulness and numerical behavior, indicating that the results of numerical comparison of the QAP lower bounds' performance may depend heavily on the implementation. Roupin (2009) explains how partial Lagrangian relaxations can be used to easily develop various strong SDP relaxations. Overall, due to their tightness and despite significant computational challenges involved, the approaches utilizing the SDP bounds appear to be very promising for solving large instances of the QAP. For additional information on the SDP relaxations and the performance of their lower bounds for the QAP, see Loiola et al. (2007), Rendl and Sotirov (2007), and Roupin (2009).

Variance reduction lower bounds can be used for the QAP in the Koopmans-Beckmann form (2). These bounds perform well when the flow and distance matrices exhibit high variances. Unfortunately, when the variance of the coefficient matrices is small, they are rather weak, with performance comparable to that of GLB. Eigenvalue-based lower bounds capitalize on the connection between the QAP objective in the trace formulation (9) and the eigenvalues of its coefficient matrices. Although the quality of these bounds is better than GLB, they are disadvantageous in terms of computational time, and their usefulness for reducing the search space diminishes with each subsequent level of the branch and bound. Several QAP lower bounds are based on the idea of decompositions, which involves exploiting special structure of some restricted QAP instances. The approach works well for the rectilinear QAPs called grid QAPs, where the flow and distance matrices are given by the distances of the points on the rectangular grid. Finally, as described by Kaibel (2000), polyhedral methods can used to compute good quality lower bounds for the QAP.

Exact Solution Approaches

Various algorithms for solving the QAP to optimality have been developed. Many of the exact algorithms proposed for the QAP are some type of branch-and-bound (B&B) procedures. Other exact approaches used for solving the QAP are traditional cutting plane methods and polyhedral cutting plane

(or branch-and-cut) methods. B&B algorithms appear to be the most efficient exact procedures for solving the QAP. The performance of B&B algorithms on large-sized QAPs strongly depends on the quality and computational efficiency of the lower bounds employed by such algorithms. As discussed above, as well as in the surveys by Loiola et al. (2007) and Hahn et al. (2010), B&B approaches based on the RLT lower bounds are among the best and are capable of solving difficult large QAP instances.

In terms of the rules used in the construction of a tree or a forest of trees of the branching process, the B&B type procedures can be grouped into three categories:

- Algorithms based on single assignment;
- Algorithms based on pair assignment;
- Relative positioning algorithms.

Single assignment algorithms appear to be the most efficient.

Since the decisions made at the early branching stages clearly play an enormous role in the consequent evolution of the search tree, the idea of strong branching was introduced in an attempt to reduce the computation for the QAP and other hard CO problems. Strong branching utilizes dual information associated with the bounds. Strong branching works by computing bounds either partially or fully for the candidate child nodes in the search tree, and selecting the final branching variables so that the resulting search space is reduced as much as possible. Strong branching appears to be very effective at reducing the size of the B&B tree, which is especially useful for solving large QAPs as illustrated in Anstreicher (2003).

A number of cutting plane algorithms for finding the optimal solution of the QAP exist (Burkard et al. 1999). Traditional cutting plane algorithms for the QAP incorporate MILP formulations, which allow the use of Bender's decomposition. In the application of cutting plane algorithms to the QAP, the performance of these procedures is usually rather slow, due to significant amount of time needed for the upper and lower bounds to converge. Consequently, traditional cutting plane algorithms can solve only QAP instances of a small size. On the other hand, the heuristics derived from these algorithms are able to produce good quality feasible solutions early in the search (Burkard et al. 1999).

Polyhedral cutting plane methods, also known as branch-and-cut algorithms (B&C), can be used for solving the QAP to optimality. Cutting plane algorithms based on the polyhedral results in Kaibel (2000) have several advantages over traditional cutting plane approaches. In particular, the former generate the cuts valid for the entire polytope of feasible solutions, in contrast to traditional techniques. Hence, the polyhedral techniques do not require complete recomputation for different cuts, which means that they need less running time and memory than the traditional cutting plane algorithms. Computational studies indicate B&C methods employing the box inequalities for computing the lower bounds are promising for computing tight lower bounds and even finding optimal solutions. However, the running times of these algorithms on larger QAP instances are considerably slow because of significant computational requirements for computing the bounds. A possible way to improve running times of the polyhedral cutting plane algorithms is to exploit special cost structures, e.g., sparsity of the QAP objective function.

Heuristics

Because solving the QAP has proved to be so hard for even moderately-sized problem instances, a large body of research is devoted to the development of approaches for finding good quality feasible solutions. These heuristic algorithms for the QAP can be categorized into the following groups:

- Constructive heuristics;
- Heuristics derived based on limited enumeration methods;
- Improvement methods;
- Genetic algorithms (GA);
- Simulated annealing (SA);
- Ant colony optimization (ACO) methods;
- Greedy randomized adaptive search procedures (GRASP);
- Tabu search (TS);
- Memetic algorithms (MA);
- Path relinking (PR);
- Artificial neural networks (ANN);
- Hybrid algorithms.

Constructive heuristics are one of the earliest approximate methods for the QAP. They work iteratively often starting with an empty permutation and gradually building partial permutations into

a feasible solution of the QAP, e.g., by assigning a given object to an available location. Heuristics based on limited enumeration methods typically exploit the fact that enumeration methods, e.g., B&B, can often produce a good quality solution in early stages of the search. The disadvantage of these procedures is that the stopping criteria employed by limited enumeration methods tend to eliminate the optimal solution (Loiola et al. 2007).

Improvement methods are local search (LS) algorithms, and as such they depend on the definition of neighborhood structure. Some of the neighborhood structures commonly used for the QAP are the pair-exchange and cyclic triple-exchange neighborhoods, which are based on the permutation representation of a feasible solution. Another important characteristic of LS procedures is the so-called update rule for the selection of the next current solution. The rules commonly used by the improvement methods to search the QAP solution space are the first improvement, the best improvement, and the Heider's rule (Burkard et al. 1999).

Because the QAP has many local optima, the improvement methods are likely to terminate in a local optimum. Therefore, they are usually performed multiple times starting with a new initial solution. Some interesting theoretical results relevant to the LS methods for special cases and general QAP are obtained in Barvinok and Stephen (2003). Several metaheuristic approaches applied to the QAP, such as SA, GRASP, and TS, incorporate LS procedures.

A number of different algorithms based on simulated annealing (SA) exist for the QAP. All of these procedures use the pair-exchange neighborhood, but integrate distinct cooling and thermal equilibrium schemes. Numerical experiments indicate that the performance of SA is significantly impacted by the choice of the cooling scheme and other control parameters. Since SA can be viewed as a non-homogeneous ergodic Markov chain, under suitable conditions SA converges asymptotically to the QAP optimal solution; however, the SA convergence speed for the QAP is hard to analyze theoretically (Burkard et al. 1999).

The GRASP metaheuristic has been applied successfully for solving various hard CO problems, including the QAP. It combines the greedy improvement phase with the random search phase that allows to explore the space of the feasible solutions.

Tabu search (TS) is another LS-based metaheuristic that has been shown to be useful for solving the QAP. It facilitates the efficient exploration of the solution space by placing some solutions in the tabu list and using the aspiration criterion to override the tabu status of the solution. Different implementations of TS for the QAP exist, including the TS with fixed tabu list, the robust TS, and the reactive TS. Based on numerical studies reported in Burkard et al. (1999), the latter TS implementation appears to outperform other traditional TS schemes. Exponentially decreasing tabu effects also appears to improve the performance of TS on the QAP. Useful strategic diversifications can be incorporated into TS for the QAP both with and without hybridization. TS algorithms for the QAP are shown to benefit from hybridization with other heuristics. One example is TS with mutation (where the idea of mutation is borrowed from GA).

Genetic algorithms (GAs) are another metaheuristic that have been applied to solving the QAP. Standard GA procedures perform poorly even on small to moderate size instances of the QAP, so to improve the performance of GAs on the QAP, several hybridization schemes have been proposed, including a method combining GA with TS and another hybrid procedure incorporating a greedy scheme into GA. In particular, the greedy GA performed well on large-scale instances from QAPLIB (Burkard et al. 1997). An alternative improved GA technique is a two-phase approach that works, essentially, by hybridizing GA with itself, since in each phase different GA-based approaches are used. Another improved GA method takes advantage of massive parallelization on GPUs (graphic processing units).

In addition to GAs, another evolutionary-based technique applied to the QAP is a genetic LS, better known as memetic algorithms (MA). These procedures combine recombination of the good solutions with LS algorithms. Numerical experiments show that MA can outperform reactive TS, robust TS and fast ACO. Parallel MA implementations for large QAPs also exist.

Path-relinking (PR) is another approach used for the QAP. Its main advantage is that PR can often quickly find new local optima on its path connecting two high-quality solutions. Such paths may have many non-improving solutions that are typically barriers for standard LS-based procedures. Both serial and parallel PR implementations for the QAP exist.

An interesting alternative to more commonly used methods for solving CO problems is an approach based on artificial neural networks (ANN or simply NN). ANN implementation for the QAP, based on the Hopfield NN, cannot by itself compete with more traditional methods, but can serve to obtain initial solutions for further application of improvement methods. An improved performance of ANN for the QAP can also be achieved by incorporating chaotic behavior and burst noise instead of uniform noise in the Hopfield NN.

Ant colony optimization (ACO) procedures have also been applied to the QAP. These are nature-inspired multi-agent search algorithms that imitate the behavior of ants during the search for food. Ants initially start searching for food randomly, then as they gradually mark the trail leading to promising solutions with the pheromone, the colony's search begin to gravitate more towards following the trails where the good quality solutions were often found at earlier stages. To avoid search stagnation, the pheromone strength is allowed to decrease with time. Although early ACO implementations for the QAP were not competitive with other metaheuristics, numerical results show that ACO appears to be one of the best available procedures for real-life, structured QAP instances.

Hybridizing algorithms for the QAP appears to be a useful and popular technique. In addition to aforementioned hybrid algorithms, a number of hybrid procedures for the QAP appear in the literature, including algorithms combining several methods, e.g., LS and ACO with TS, or LS and ACO with SA, or LS and ACO with GA. Other QAP solution algorithms hybridize GA with LS, GA with TS, GRASP with PR, and even TS and ANN.

Distributed Computing

Parallel implementations for the QAP have appeared since the 1990s. In fact, many large difficult instances of the QAP were solved using distributed algorithms. In distributed computing, a network of connected machines is utilized to perform computationally expensive tasks. The size of such network may vary considerably from small local area networks to the Internet. As described in Anstreicher (2003), parallel implementations of QAP solution algorithms share common difficulties with other computational tasks performed using distributed computations, including load balancing and reliability issues. In fact, worker processes are inherently unreliable, as the availability of CPUs (central processing units) is typically dynamic and the master process utilizes worker processes as they become available. Clearly, fault tolerance is very important for master and worker processes.

In terms of the application of metacomputing and grid computing, which involve geographically distributed computer networks, QAP solution methods based on computationally inexpensive lower bounds (e.g., GLB) appear to be easier to implement in parallel as compared to the more computationally demanding lower bounds (e.g., many SDP-based bounds). Some approaches, such as ACO or TS, are inherently easier to implement in parallel, as they may allow, as in the case of ACO and TS, the search to be easily divided among multiple processes. For further discussion of the application of distributed computing to the QAP, see Anstreicher (2003) and Pardalos and Pitsoulis (2000).

Landscapes and Asymptotics

The investigation of the fitness landscape and the asymptotic properties of the QAP has led to a number of important results. The study by Barvinok and Stephen (2003) investigated the distribution of the objective function values with respect to the Hamming distance on permutations for a general QAP and some special cases, such as TSP, using the representation theory of the symmetric group. These results have interesting interpretations in terms of performance of the LS algorithms.

In CO, the fitness landscape is used to study the performance of LS-based heuristics. The landscape of a given QAP instance can be thought as an ordered quadruple $(S, \mathcal{N}, f, d)$ of the space S of all feasible solutions of the QAP, a neighborhood structure $\mathcal{N}$ on S, a distance d between the pairs of solutions from S, and the solution fitness f. The objective function value is usually referred as the solution fitness. A number of approaches, including fitness-distance correlation coefficient and correlation length, have been applied to analyze the so-called landscape ruggedness. A flat landscape, where the costs between the neighboring solutions are very close on average, appears to be better suited for the LS algorithms as compared to

a more rugged landscape. It turns out that the fitness landscape properties of the QAP depend considerably on the degree of correlation in the structure of cost coefficients as well as the magnitude of the flow dominance characteristic (Krokhmal and Pardalos 2009). The flow dominance is essentially defined as the ratio of the standard deviation of the coefficients in the flow matrix to their average. The numerical analysis of various QAP instances indicates that many QAPs have rather unstructured landscapes, with the exception of the randomly generated QAP instances with cost matrices satisfying the triangle inequality. In fact, it has been observed that the LS-based heuristics appear to perform better on the structured QAPs with low flow dominance.

In general, the asymptotic properties of the QAP with randomly generated cost coefficients differ considerably from the asymptotic behavior exhibited by its linear counterparts, such as the LAP or the multi-dimensional assignment problem (MAP). An interesting property of the randomly generated QAPs is that the ratio of the objective function values between the optimal solution and an arbitrary (including the worst) feasible solution of the QAP converges to one in probability as the size of the problems approaches infinity. However, the convergence rate is rather slow. This convergence implies that despite the complexity of the QAP, for extremely large problems any algorithm can produce a solution of very good quality. In fact, it is possible to prove that a similar result holds not only in probability but also almost surely. As explained in Krokhmal and Pardalos (2009), investigation of the asymptotic behavior of the QAP has led to discovery of a more general class of CO problems with analogous asymptotic properties. Another noteworthy analytic result proves that a relationship similar to the well-known Chebyshev's inequality holds for the optimal solution value of random QAPs.

Additional details on the asymptotic properties and fitness landscape analysis of the QAP, as well as comparison of these two characteristics for other assignment problems, can be found in the survey by Krokhmal and Pardalos (2009).

Generalizations and Related Problems

The relationship of the QAP to many other related problems have been known for some time (Pardalos and Pitsoulis 2000). Not only can numerous hard CO problems (e.g., the TSP, the linear arrangement, the graph partitioning, the maximum clique, the minimum weight feedback arc set, and the graph packing problems) be viewed as special cases of the QAP, but also the QAP itself can be considered as a special case of other CO problems, e.g., the quadratic three-dimensional assignment problem (Q3AP), which can be viewed as a problem of minimizing the quadratic and linear costs associated with finding a tripartite matching. Mathematically, the Q3AP can be compactly formulated as follows:

$$\min_{\pi,\sigma\in\Pi_n} \sum_{i=1}^{n}\sum_{j=1}^{n} c_{i\pi(i)\sigma(i)j\pi(j)\sigma(j)} \tag{18}$$

where as before Π_n denotes the set of all possible permutations of n elements. The Q3AP arises as a problem of minimizing a transmission error bound in the design of the wireless communication systems where a digital message is automatically repeated twice to improve wireless transmission quality. During each message repeat the data are mapped into symbols for a transmission. To minimize the likelihood of a transmission error, the two mappings of the data into transmitted symbols must be as independent as possible. Several solution procedures for the Q3AP have been proposed, including B&B algorithms based on the RLT techniques similar to those used for solving large QAPs. Implementing the exact algorithms for the Q3AP requires solving a three-dimensional extension of the LAP, itself an NP-hard problem (Hahn et al. 2010). A number of heuristic approaches originally developed for the QAP, including stochastic LS, have also been applied for solving the Q3AP. Serial implementations of solution algorithms for the Q3AP have only been able to solve the Q3APs of a moderate size. Overall, solving larger instances of the Q3AP may require development of parallel algorithms.

The survey by Hahn et al. (2010) presents the QAP as a fundamental problem in a growing class of practically important and computationally difficult assignment problems that in addition to the Q3AP, includes the linear three-dimensional assignment problem (3AP), the cubic and generalized cubic assignment problems (CAP and GCAP, respectively), the biquadratic assignment problem (BiQAP), the generalized linear and generalized quadratic assignment problems (G3AP and GQ3AP, respectively), the multi-story

space assignment (MSAP), the cross-dock door assignment problem (CDAP), and the stochastic quadratic assignment problem (SQAP). Other APs related to the QAP include a polynomially-solvable LAP and several NP-hard problems, such as MAP (which is a generalization of the LAP and 3AP), the bottleneck QAP, and the quadratic semi-assignment problem (QSAP). Using the totally ordered commutative semigroups, the so-called algebraic LAP and algebraic QAP are defined as algebraic generalizations of the LAP and the QAP, respectively.

See

- ▶ Computational Complexity
- ▶ Convex Optimization
- ▶ Evolutionary Algorithms
- ▶ Facilities Layout
- ▶ Genetic Algorithms
- ▶ Heuristics
- ▶ Integer and Combinatorial Optimization
- ▶ Metaheuristics
- ▶ Neural Networks
- ▶ Simulated Annealing
- ▶ Tabu Search

References

Adams, W. P., & Johnson, T. A. (1994). Improved linear programming-based lower bounds for the quadratic assignment problem. In P. M. Pardalos & H. Wolkowicz (Eds.), *Quadratic assignment and related problems* (DIMACS series on discrete mathematics and theoretical computer science, Vol. 16, pp. 43–75). Providence, RI: AMS.

Anstreicher, K. M. (2003). Recent advances in the solution of the quadratic assignment problems. *Mathematical Programming, 97*, 27–42.

Barvinok, A., & Stephen, T. (2003). The distribution of values in the quadratic assignment problem. *Mathematics of Operations Research, 28*(1), 64–91.

Bazaraa, M. S., & Sherali, H. D. (1982). On the use of exact and heuristic cutting plane methods for the quadratic assignment problem. *Journal of Operations Research Society, 33*, 991–1003.

Burer, S., & Vandenbussche, D. (2006). Solving lift-and-project relaxations of binary integer programs. *SIAM Journal on Optimization, 16*(3), 726–750.

Burkard, R., Çela, E., Pardalos, P., & Pitsoulis, L. (1999). The quadratic assignment problem. In D.-Z. Du & P. M. Pardalos (Eds.), *Handbook of combinatorial optimization* (pp. 75–149). Boston: Kluwer.

Burkard, R., Dell'Amico, M., & Martello, S. (2009). Assignment problems. In *SIAM monograph, SIAM books*. Philadelphia, PA: SIAM.

Burkard, R., Karisch, S. E., & Rendl, F. (1997). QAPLIB – a quadratic assignment problem library. *Journal of Global Optimization, 10*, 391–403.

Dyer, M. E., Frieze, A. M., & McDiarmid, C. J. H. (1986). On linear programs with random costs. *Mathematical Programming, 35*, 3–16.

Frieze, A. M., & Yadegar, J. (1983). On the quadratic assignment problem. *Discrete Applied Mathematics, 5*, 89–98.

Hahn, P. M., Zhu, Y.-R., Guignard, M., & Smith, J. M. (2010). Exact solution of emerging quadratic assignment problems. *International Transactions in Operations Research, 17*, 525–552.

Kaibel, V. (2000). Polyhedral methods for the quadratic assignment problem. In P. Pardalos & L. Pitsoulis (Eds.), *Nonlinear assignment problems: Algorithms and applications* (pp. 109–141). Boston: Kluwer Academic Publishers.

Kaufmann, L., & Broeckx, F. (1978). An algorithm for the quadratic assignment problem using Bender's decomposition. *European Journal of Operations Research, 2*, 204–211.

Koopmans, T. C., & Beckmann, M. J. (1957). Assignment problems and the location of economic activities. *Econometrica, 25*, 53–76.

Krokhmal, P., & Pardalos, P. (2009). Random assignment problems. *European Journal of Operational Research, 194*(1), 1–17.

Lawler, E. L. (1963). The quadratic assignment problem. *Management Science, 9*, 586–599.

Loiola, E. M., de Abreu, N. M. M., Boaventura-Netto, P. O., Hahn, P. M., & Querido, T. (2007). A survey for the quadratic assignment problem. *European Journal of Operational Research, 176*, 657–690.

Mittelmann, H., & Peng, J. (2010). Estimating bounds for quadratic assignment problems associated with the Hamming and Manhattan distance matrices based on semidefinite programming. *SIAM Journal on Optimization, 20*, 3408–3426.

Pardalos, P. M. and Pitsoulis, L., Co-editors. (2000). *Nonlinear assignment problems: Algorithms and applications*. Boston: Kluwer.

Pardalos, P. M. and Wolkowicz, H., Co-editors. (1994). *Quadratic assignment and related problems*. DIMACS series (Vol. 16). Providence, RI: American Mathematical Society.

Pitsoulis, L., & Pardalos, P. M. (2009). Quadratic assignment problem. In C. A. Floudas & P. M. Pardalos (Eds.), *Encyclopedia of optimization* (pp. 3119–3149). New York: Springer.

Queyranne, M. (1986). Performance ratio of heuristic for triangle inequality quadratic assignment problem. *Operations Research Letters, 4*, 231–234.

Rendl, F., & Sotirov, R. (2007). Bounds for the quadratic assignment problem using the bundle method. *Mathematical Programming, Series B, 109*, 505–524.

Roupin, F. (2009). Semidefinite relaxations of the quadratic assignment problem in a Lagrangian framework. *International Journal of Mathematics in Operational Research, 1*(1/2), 144–162.
Sahni, S., & Gonzalez, T. (1976). P-complete approximation problems. *Journal of the Association of Computing Machinery, 23*, 555–565.
Zhao, Q., Karisch, S. E., Rendl, F., & Wolkowicz, H. (1998). Semidefinite relaxations for the quadratic assignment problem. *Journal of Combinatorial Optimization, 2*, 71–109.

Quadratic Form

A function that can be written as $\boldsymbol{x^T\ Cx}$, where the $n \times n$ matrix $\boldsymbol{C}$ is a matrix of known coefficients and $\boldsymbol{x}$ is a column vector. Matrix $\boldsymbol{C}$ is usually assumed to be symmetric or can be transformed into a symmetric matrix. The form is said to be positive definite if $\boldsymbol{x^T\ Cx} > 0$ for $\boldsymbol{x} \neq 0$. The form is positive semidefinite if $\boldsymbol{x^T\ Cx} \geq 0$ for all $\boldsymbol{x}$. Negative definite and negative semidefinite forms are defined by appropriate reversal of the inequality signs in the preceding definitions.

See

► Quadratic Programming

Quadratic Programming

Katta G. Murty
University of Michigan, Ann Arbor, MI, USA
Kaing Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

Introduction

Quadratic programming (QP) deals with a special class of mathematical programs in which a quadratic function of the decision variables is required to be optimized (i.e., either minimized or maximized) subject to linear equality and/or inequality constraints.

Let $\boldsymbol{x} = (\boldsymbol{x_1}, \ldots, \boldsymbol{x_n})^T$ denote the column vector of decision variables. In mathematical programming it is standard practice to handle a problem requiring the maximization of a function $f(\boldsymbol{x})$ subject to some constraints, by minimizing $-f(\boldsymbol{x})$ subject to the same constraints. Both problems have the same set of optimum solutions. Because of this, the discussion is restricted to minimization problems, without any loss of generality.

A quadratic function of decision variables $\mathbf{x}$ is a function of the form

$$Q(\boldsymbol{x}) = \sum_{i=1}^{n} \sum_{j=i}^{n} q_{ij} x_i x_j + \sum_{j=1}^{n} c_j x_j + c_0.$$

Define $\boldsymbol{c} = (c_1, \ldots, c_n)$, and a square symmetric matrix $\boldsymbol{D} = (d_{ij})$ of order n, where

$$d_{ii} = 2q_{ii} \text{ for all } i = 1 \text{ to } n \triangleleft d_{ij} = d_{ji} = q_{ij} \text{ for } j > i$$

Then in matrix notation, $Q(\boldsymbol{x}) = \frac{1}{2}\mathbf{x}^T D\mathbf{x} + \mathrm{c}\mathbf{x} + c_0$. Here $\boldsymbol{D}$ is the Hessian matrix (i.e., the matrix of second order partial derivatives) of $Q(\boldsymbol{x})$.

As an example, consider $n = 3$, $\mathbf{x} = (x_1, x_2, x_3)^T$, and $h(\boldsymbol{x}) = 81x_1^2 - 7x_2^2 + 5x_1x_2 - 6x_1x_3 + 18x_2x_3$. This quadratic function $h(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{D}\mathbf{x}$ where

$$\boldsymbol{D} = \begin{pmatrix} 162 & 5 & -6 \\ 5 & -14 & 18 \\ -6 & 18 & 0 \end{pmatrix}.$$

A quadratic function is the simplest nonlinear function, and hence they have always served as model functions for approximating general nonlinear functions by local models.

A square matrix $\boldsymbol{D}$ of order n is said to be

Positive semidefinite (PSD) if $\boldsymbol{x^TDx} \geq 0$ for all $\boldsymbol{x} \in R^n$;

Positive definite (PD) if $\boldsymbol{x^TDx} > 0$ for all nonzero $\boldsymbol{x} \in R^n$.

These matrix-theoretic concepts are important in the study of QP because the quadratic function $Q(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x^TDx} + \boldsymbol{cx} + \mathrm{c}_0$ is a convex function over R^n iff the matrix $\boldsymbol{D}$ is PSD.

Superdiagonalization Algorithm to Check Whether a Given Square Matrix is PD, PSD

Let $\boldsymbol{M}_0$ be the square matrix of order n to be checked. Let $\boldsymbol{D}^n = \boldsymbol{M}_0 + \boldsymbol{M}_0^T$. $\boldsymbol{D}^n$ is symmetric. $\boldsymbol{M}_0$ is PSD,

PD iff $\boldsymbol{D}^n$ is. The algorithm consists of at most n steps carried out on $\boldsymbol{D}^n$. In each step, one Gaussian pivot step is carried out; and some rows t and the corresponding columns t are deleted, so the remaining matrix is a symmetric square matrix of smaller order. When the remaining matrix is of order s, denote it by $\boldsymbol{D}^s$. The algorithm begins with $s = n$.

General Step: Let $\boldsymbol{D}^r$ be the current matrix of order r.

If any principal diagonal entry of $\boldsymbol{D}^r$ is negative, $\boldsymbol{M}_0$ is not even PSD; terminate with this conclusion.

If any diagonal entry of $\boldsymbol{D}^r$ is 0, the row and column of $\boldsymbol{D}^r$ of that entry must be 0-vectors; otherwise $\boldsymbol{M}_0$ is not even PSD; terminate with this conclusion. If that row and column are 0-vectors, eliminate them, and continue with the remaining matrix.

Now all diagonal entries of the remaining matrix are positive. Perform a Gaussian pivot step on this remaining matrix in column 1 with its diagonal entry as the pivot element, to convert all nondiagonal entries in this column to 0. After the pivot step, eliminate row 1, column 1 of the resulting matrix, and with the remaining matrix go to the next step.

If the conclusion that $\boldsymbol{M}_0$ is not PSD is never reached, and all the steps are carried out, the algorithm will terminate with a 1×1 matrix, i.e., a positive number. in this case terminate with the conclusion that $\boldsymbol{M}_0$ is PSD. It is also PD if no 0-diagonal entry appeared during the algorithm.

For numerical examples, see Sect. 9.2 in Murty (2010).

Classification of Quadratic Programs

QPs can be classified into the following types.

Unconstrained quadratic minimization problem: minimization of a quadratic function $Q(\boldsymbol{x})$ over the entire space.

Equality constrained quadratic minimization problem: minimization of a quadratic function $Q(\boldsymbol{x})$ subject to linear equality constraints on the variables, $\boldsymbol{Ax} = \boldsymbol{b}$. These equations can be used to eliminate some variables by expressing them in terms of the others, and thereby transform the problem into an unconstrained one in the remaining variables. Thus these problems are mathematically equivalent to (and can be solved by techniques similar to those of) unconstrained quadratic minimization problems.

Inequality constrained quadratic minimization problem: minimization of a quadratic function $Q(\boldsymbol{x})$ subject to linear inequality constraints $\boldsymbol{Bx} \geq \boldsymbol{d}$, and possibly bounds on individual variables $\ell \leq x \leq u$, and may be some equality constraints $\boldsymbol{Ax} = \boldsymbol{b}$.

Quadratic network optimization problem: quadratic program in which the constraints are flow conservation constraints on a pure or generalized network.

Bound constrained quadratic minimization problem: minimization of a quadratic function subject only to bounds (lower and/or upper) on the variables.

Convex quadratic program (**CQP**): any of the above problems in which the objective function to be minimized, $Q(\boldsymbol{x})$, is convex.

Nonconvex quadratic program: any of the above problems in which the objective function to be minimized, $Q(\boldsymbol{x})$, is nonconvex.

Linear complementarity problem (**LCP**): special problem dealing with a system of equations in nonnegative variables in which the variables are formed into various pairs called complementary pairs. A feasible solution in which at least one variable in each pair is zero is desired. There is no objective function to be minimized in this problem. The first order necessary optimality conditions for a QP are in the form of an LCP. And in turn every LCP can be posed as a QP.

Nearest point problems: special type of QPs that require finding the nearest point in the set of feasible solutions of a system of linear constraints, by Euclidean distance, to a given point (Liu and Fathi 2011).

Unconstrained Quadratic Minimization in Classical Mathematics

Historically, quadratic functions became prominent because they provide simple local models for general nonlinear functions. A quadratic function is the simplest nonlinear function, and when used as a local approximation for a general nonlinear function, it can capture the important curvature information that a linear approximation cannot.

The use of quadratic approximations to handle general nonlinear functions goes back a very long time. Some important instances are the following.

1. **Newton's method** Newton developed the celebrated method for finding an unconstrained

minimum of a twice continuously differentiable function, $f(\boldsymbol{x})$, by constructing the local model for $f(\boldsymbol{x}^r + \boldsymbol{y})$ at the current point $\boldsymbol{x}^r$ to be the quadratic function $Q(\boldsymbol{y}) = f(\boldsymbol{x}^r) + \nabla f(\boldsymbol{x}^r)\boldsymbol{y} + \frac{1}{2}\boldsymbol{y}^T H(f(\boldsymbol{x}^r))\boldsymbol{y}$, where $\nabla f(\boldsymbol{x}^r)$ is the row vector of the first order partial derivatives of $f(\boldsymbol{x})$ at $\boldsymbol{x}^r$, and $H(f(\boldsymbol{x}^r))$ is the Hessian matrix of $f(\boldsymbol{x})$ at $\boldsymbol{x}^r$. $Q(\boldsymbol{y})$ is the 2nd order Taylor series approximation for $f(\boldsymbol{x})$ at $\boldsymbol{x}^r$. The method finds the minimizer $\boldsymbol{y}^r$ of the model function $Q(\boldsymbol{y})$ (assuming that $H(f(\boldsymbol{x}^r))$ is PD, then $\boldsymbol{y}^r = -(H(f(\boldsymbol{x}^r)))^{-1}(\nabla f(\boldsymbol{x}^r))^T$), and the next point to be $\boldsymbol{x}^{r+1} = \boldsymbol{x}^r + \boldsymbol{y}^r$.

Thus Newton's method solves an unconstrained quadratic minimization problem in each step. Starting from an initial point $\boldsymbol{x}^0$, it generates the sequence $\{\boldsymbol{x}^r\}$, which under certain conditions can be shown to converge to the minimum of the original function $f(\boldsymbol{x})$.

To treat the case where the Hessian $H(f(\boldsymbol{x}^r))$ may not be PD, several modified Newton methods based on quadratic models different from the 2nd order Taylor series approximation at $\boldsymbol{x}^r$, have been developed.

Also, the mathematically beautiful theory of quasi-Newton methods for unconstrained minimization has also been developed through the study of quadratic models (Bazaraa et al. 1993).

2. **Conjugate gradient method** There is the very efficient Gaussian elimination method for solving a square nonsingular system of linear equations, $\boldsymbol{Ax} = \boldsymbol{b}$ say, of order n. However, when n is very large, this method becomes unwieldy and difficult to implement. The least squares formulation of this system of equations is the unconstrained quadratic minimization problem

$$\text{Minimize } (\boldsymbol{Ax} - \boldsymbol{b})^{\mathrm{T}}(\boldsymbol{Ax} - \boldsymbol{b})$$

and Hestenes and Stiefel developed the conjugate gradient method for solving this problem in the 1950s. Subsequently, through the study of the quadratic model, several researchers have extended this method directly into a variety of conjugate gradient methods for the unconstrained minimization of general nonlinear functions.

3. **Linear least squares** Consider a large system of linear equations (typically overdetermined, i.e., where the number of equations exceeds the number of variables), say $\boldsymbol{Ax} = \boldsymbol{b}$, which has no exact solution. A common approach for handling such a system is to look for a least squares solution, i.e., an optimum solution of the unconstrained quadratic minimization problem

$$\text{Minimize } (\boldsymbol{Ax} - \boldsymbol{b})^{\mathrm{T}}(\boldsymbol{Ax} - \boldsymbol{b})$$

This problem is known as the linear least squares problem. Powerful numerical linear algebra techniques such as singular value decomposition (SVD) have been developed to solve large scale versions of this special class of QPs. Statisticians have been using the linear least squares model for computing the estimates of the coefficients in a linear regression model for a long time.

Types of Solutions

Linear programming (LP) deals with only one type of optimum solution, but not about different types of optima such as local and global optima. That is because every local optimum, and every point satisfying the first order necessary optimality conditions for an LP, is also a global optimum. Unfortunately this is not the case in general QPs.

For a QP, or any mathematical program in which an objective function $\theta(\boldsymbol{x})$ is required to be minimized, there are the following types of optimum solutions:

Local minimum — a feasible solution $\bar{\boldsymbol{x}}$ for which there exists an $\epsilon > 0$ such that $\theta(\boldsymbol{x}) \geq \theta(\bar{\boldsymbol{x}})$ for all feasible solutions within a Euclidean distance of at most ϵ from $\bar{\boldsymbol{x}}$

Global minimum — a feasible solution $\hat{\boldsymbol{x}}$ satisfying $\theta(\boldsymbol{x}) \geq \theta(\hat{\boldsymbol{x}})$ for all feasible solutions $\boldsymbol{x}$

Stationary point or **KKT point** — a feasible solution satisfying the first order necessary optimality conditions (also called the KKT (Karush, Kuhn, Tucker) optimality conditions) for the problem.

In a convex QP, every stationary point (KKT point), or a local minimum, is a global minimum; hence all these concepts converge in a convex QP. The same may not be true in nonconvex QPs, i.e., there may be local minima which are not global minima, and stationary points which are neither global nor local minima. Also, the problem may have some local minima, even when the objective function is unbounded below on the set of feasible solutions.

The first order (KKT) necessary optimality conditions for a QP will be referred to as its KKT system.

What Types of Solutions Can Be Computed Efficiently by Existing Algorithms?

Like LPs, QPs have the property that when the set of feasible solutions is nonempty, either a global minimum exists, or the objective function is unbounded below on the set of feasible solutions. And for both convex and nonconvex QPs, there exist finite algorithms for checking whether the objective function is unbounded below on the set of feasible solutions, and for computing a global optimum solution when one exists.

For convex QPs there are very efficient algorithms for computing a global minimum when it exists, and very high quality software implementing these algorithms is available commercially.

For nonconvex QPs, even though finite algorithms for computing a global minimum are available, they are impractical, because the computational effort needed by them grows exponentially with the size of the problem being solved. Nonconvex QP is NP-hard, and so far there is no algorithm known that is guaranteed to find a global minimum for it within a reasonable time.

Can at least a local minimum for a nonconvex QP be computed efficiently? Unfortunately, even the problem of checking whether a given feasible solution is a local minimum for a nonconvex QP may be a hard problem. In Murty and Kabadi (1987), it has been shown that the problem of checking whether 0 is a local minimum in the following simple QP

$$\begin{aligned} &\text{Minimize} && \boldsymbol{x}^{\mathrm{T}}\boldsymbol{D}\boldsymbol{x} \\ &\text{subject to} && \boldsymbol{x} \geq 0 \end{aligned}$$

is a co-NP-complete problem when $\boldsymbol{D}$ is not PSD. In this paper, it has been explained that when dealing with a nonconvex QP, a reasonable practical goal is to look for an algorithm that produces a descent sequence (i.e., a sequence of feasible points along which the objective value strictly decreases) converging to a KKT point. Some of the algorithms discussed below have this property.

Some Important Applications of QP

Finance Analysis using QP models is an established part of selecting optimum investment strategies. Perhaps Markowitz (1959) is the first published book in this area. The Markovitz model employs the variation in return as measured by the quadratic function $\boldsymbol{x}^T\boldsymbol{D}\boldsymbol{x}$, where $\boldsymbol{D}$ is the variance/covariance matrix, and $\boldsymbol{x}$ is the vector of stock investments; as a measure of the risk. This risk is the objective function to be minimized. Constraints in the model guarantee conservation on the flow of funds, and a lower bound on the expected returns from the portfolio. There may also be bounds placed on the investments in particular sectors of the economy (such as utilities, etc.) to make sure that the model does not put too many eggs in any basket, thus achieving diversification. Many other practical aspects of investing can easily be included by either adding appropriate constraints or modifying the objective function by including quadratic penalty terms.

Portfolio management The portfolio optimization problem discussed above is a widely studied static problem since it only determines the optimum investment amounts in various stocks at one point of time. But the real problem in managing investments, known as portfolio management, is dynamic, as yield and risk data keep changing over time randomly. Many authors (e.g., Mulvey 1987) have designed multiperiod quadratic generalized network flow models in which interest, dividends, and loans are modeled by means of arc multipliers.

Taxation QP models play a very important role in the analysis of tax policies. Political leaders at the national and state levels are relying more and more on such analyses to forecast growth rates in tax revenues, and to set various taxes at levels that are likely to ensure growth at desired rates. White (1983) gives a detailed description of such an analysis carried out for the state of Georgia.

National and state government taxes such as sales tax, motor fuels tax, alcoholic beverages tax, personal income tax, etc. are all set at levels to ensure a healthy economic growth. Government finance is based on the assumption of predictable and steady growth of each tax over time.

If s is the tax rate for a particular tax and S_t the expected tax revenue for this tax in year t, then a typical regression equation used to predict S_t as

a function of s and t is: $\log_e S_t = a + bt + cs$ where a, b, c are parameters to be estimated from past data to give the closest fit by the least squares method. The annual growth rate in this tax revenue is then the regression coefficient b multiplied by 100 to convert it to percent.

The decision variables in the model are: $s_j =$ the tax rate for tax j in the base year (0th year) as a fraction. From the known tax base for tax j in the 0th year, the revenues from tax j in this year can be obtained as: s_j (tax base for tax j) $= x_j$. The instability or variability in this revenue is measured by the quadratic function $Q(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{V} \boldsymbol{x}$, where $\boldsymbol{V}$ is the variance/covariance matrix estimated from past data and $Q(\boldsymbol{x})$ is to be minimized. The constraints in the model consist of bounds on the x_j, and a condition that $\sum x_j = T$, the total expected tax revenue in the 0th year. And there is an equation that the overall growth rate which can be measured by the weighted average of the growth rates of the various taxes j, $\sum(x_j b_j)/T$ should be equal to the desired growth rate λ. Any other linear constraints that the decision variables are required to satisfy can also be included. In fact λ can be treated as a parameter, and the whole model solved as a parametric QP model. Exploring the optimum solution for different values of λ in the reasonable range yields information for the political decision makers to determine good values for the various tax rates that are consistent with expected growth in tax revenues.

Equilibrium models Economists use equilibrium models to analyze expected changes in economic conditions, predict prices, inflation rates, etc. These models often involve QPs. As an example, in Glassey (1978) a simple equilibrium model of interregional trade in a single commodity is described. He considers N regions, and the following data elements and variables.

Data:	$a_i > 0$ the equilibrium price in the ith region in the absence of imports and exports.
	$b_i > 0$ the elasticity of supply and demand in the ith region.
	c_{ij} the cost/unit to ship from i to j.
Variables:	p_i equilibrium price in the ith region.
	y_i net imports into the ith region (may be > 0, or 0, or < 0)
	x_{ij} actual exports from region i to region j.

If $p_i > a_i$, supply locally exceeds demand in the ith region, the difference being available for export. From this follows $p_i = a_i - b_i y_i$. Also, the y_i and x_{ij} are linked through flow conservation equations. The interregional trade equilibrium conditions are

$$p_i + c_{ij} \geq p_j \text{ for all i, j}$$
$$(p_i + c_{ij} - p_j) x_{ij} = 0 \text{ for all i, j}$$

If the first condition above does not hold, exports from i to j will increase until the elasticity effects in markets i and j rise, and prices will adjust so that additional profit from export no longer exists. Also, if $x_{ij} > 0$, then $p_i + c_{ij} - p_j = 0$.

It can be verified that these conditions are the first-order necessary optimality conditions for a quadratic network flow problem in which the quadratic objective function can be interpreted as a net social payoff function. Using this observation, Glassey (1978) describes a procedure for computing the equilibrium prices and flows based on solving the QP.

In the same way, traffic engineers use traffic equilibrium models solved by quadratic network flow algorithms for road and communication network planning. These traffic equilibrium models typically have hundreds of thousands of variables and constraints, and are probably the largest QP models solved on a regular basis.

Electrical networks Even during the physicist J. C. Maxwell's time in the second half of the 19th century, it has been well recognized that the equilibrium conditions of an electrical or a hydraulic network are attained at the point where the total energy loss is minimized. Dennis (1959) has formally shown that the sum of the energy losses in the resistors and at the voltage sources in an electrical network, is a quadratic function of the branch currents, if all devices in the network are of a linear (i.e., ohmic) nature. Using this he formulated the problem of determining the branch currents at equilibrium in an electrical network connecting various devices, voltage sources, diodes, and resistors, as a QP. He then showed that the optimality conditions for this QP are precisely the Kirchoff laws governing the equilibrium conditions of the network, with the Lagrange multipliers representing node potentials. In the distribution of electrical power, this QP model is used to solve the load flow problem concerned with the

flow of power through the transmission network to meet a given demand.

Power system scheduling problem The economic dispatch problem in an electrical power system operation deals with the problem of allocating the demand for power – or system load – among the generating units in operation at any point of time. The optimal allocation of load among the units to achieve a least cost allocation, depends on the relative efficiencies of the units; and can be modeled as a QP; see Wood (1984). In power system operation, this model is usually solved many times during the day with appropriate load adjustments.

Application in solving general nonlinear programs One of the most popular algorithms for solving general nonlinear programming problems is the SQP (sequential or recursive quadratic programming) method. It is an iterative method which in each iteration solves a convex QP to find a search direction, and a line search problem (one dimensional minimization problem for a merit function) in that direction. The original concepts of this method are outlined in the Harvard Ph. D. thesis of R. B. Wilson in 1963, and now it has been developed into a successful approach through the work of many researchers; see Bazaraa, Sherali and Shetty (1993), and Murty (1988). The success of these methods has made QP a very important topic in mathematical programming. A nice software package for nonlinear programs based on this approach is FSQP (Zhou and Tits 1992).

Studies involving Support Vector Machines (SVMs) SVMs are learning methods used for classification and regression. Given training examples of two categories of objects, each object represented as a vector in R^n (this vector may represent the measurements of various characteristics of the object), SVM training algorithm builds a model to construct a hyperplane in R^n (represented by a linear equation in which the coefficients are the parameters to be determined using the model) that separates the sets of points of the two categories; with the largest possible distance to the nearest training data points of any category. This hyperplane is later used to classify new examples into one of the categories based on which side of the hyperplane they fall on. The model to determine the hyperplane is a QP model. Thus algorithms for QP play an important role in applications involving SVMs.

A version of SVM for regression was proposed by Vapnik and others, this method is now called support vector regression. SVMs find many diverse applications in character recognition, image classification, clustering, machine learning, neural networks, statistics, data mining, biosequence analysis, and bioinformatics (Steinwart and Christmann 2008).

Algorithmic Developments

(a) **Frank-Wolfe method** One of the first methods for QP developed in 1950s is that of Frank and Wolfe. It is an iterative method which in each iteration solves an LP to find a search direction, and a line search problem in that direction. It produces a descent sequence such that every limit point of this sequence is a KKT point. However, the method has slow convergence, and is not popular except on problems with special structure that makes it possible to solve the LP in each iteration by an extremely fast special method taking advantage of the structure.

(b) **Reduced gradient methods** The simplex method for LP has been extended to solve problems involving the minimization of a quadratic (or in general a smooth nonlinear) function subject to linear constraints. The method is called the reduced gradient method and is discussed by P. Wolfe in 1959. The name reduced gradient method refers to any method which uses the equality constraints to eliminate some variables (called the dependent or basic variables) from the problem, and treats the remaining problem in the space of the independent (or nonbasic variables) only, either explicitly or implicitly. The reduced gradient is the gradient of the objective function in the space of independent variables. The method is quite popular. The OSL software package uses this method for solving QPs. The MINOS software package uses this method for minimizing a smooth nonlinear function subject to equality constraints.

This method has been generalized directly into the GRG (generalized reduced gradient) method for solving nonlinear programs involving nonlinear constraints. The GRG is a popular

method on which several successful nonlinear programming software packages are based.

(c) **Methods based on the LCP** In the 1950s and 60s several researchers proposed schemes for solving the QP by solving its KKT system. Lemke formulated the KKT system for a QP as an LCP and developed a beautiful algorithm for it called the complementary pivot algorithm. The data for an LCP of order n consists of a square matrix $\boldsymbol{M}$ of order n, and a column vector $\boldsymbol{q} \in R^n$; and it is to find a $\boldsymbol{w} = (w_j) \in R^n$ and a $\boldsymbol{z} = (z_j) \in R^n$ satisfying

$$\begin{aligned} \boldsymbol{w} - \boldsymbol{M}\boldsymbol{z} &= \boldsymbol{q} \\ \boldsymbol{w}, \boldsymbol{z} &\geq 0 \\ w_j z_j &= 0 \text{ for all j} \end{aligned}$$

Checking whether the general LCP has a solution is an NP-complete problem, and there are no efficient algorithms known for it. But the complementary pivot algorithm is a finite path following method for finding a solution, when one exists, to a class of LCPs which includes the KKT systems corresponding to convex QP.

The development of the complementary pivot method is a nice theoretical breakthrough for which Lemke received the von Neuman Theory Award of ORSA/TIMS in 1978. However, the complementary pivot method, and several other methods developed for the LCP, are not preferred for solving even convex QPs, because a QP involving m inequality constraints in n nonnegative variables leads to an LCP of order $m + n$, blowing up the size of the model.

For tackling nonconvex QPs, the complementary pivot approach is clearly unsuitable, as it focusses attention purely on the KKT system, and never even computes the objective value; and if it leads to a KKT point at termination, that point may not even be a local minimum.

However, the formulation of the LCP and the complementary pivot method constituted great contributions to theory. The LCP has a fascinating geometrical interpretation. The study of the geometry of LCP was initiated in the 1968 Ph. D. thesis of Murty and continues to be a very active area of research. And the mathematical principle behind the complementary pivot method has been used to develop simplicial methods (which are also called complementary pivot methods) to solve systems of nonlinear equations and fixed point problems. See Murty (1988) and Cottle, Pang and Stone (1992).

(d) **Active set methods** A popular method for solving QP is based on a combinatorial approach to iteratively determine the set of active constraints at the optimum. This type of strategy for handling inequality constrained optimization problems is called the active set strategy. The method solves a sequence of equality constrained QPs by treating some of the inequality constraints as equations (the active set) and temporarily ignoring the others. Several rules are employed to modify the active set from one iteration to the next, to guarantee finite convergence of the procedure. Several researchers have extended this method to minimize a smooth nonlinear function subject to linear equality and inequality constraints; see Bazaara, Sherali and Shetty (1993).

(e) **Interior point methods** Since the development of a very successful interior point method for LP by Karmarkar (1984), a variety of interior point methods have been developed for convex QPs and the LCPs associated with them. These methods are polynomially bounded, and some versions of them give excellent computational performance on large sparse problems. The monograph (Kojima et al. 1991) establishes the theoretical foundations for primal-dual interior point methods for LP and LCP. The authors won the 1992 Lanchester award for this monograph. Some other references on these methods are Ye (1991), Wright (1997), Vanderbei (2008).

(f) **Sphere methods** These are also interior point methods. They find a largest inscribed sphere with center having an objective value $\leq$ that at the current interior feasible solution, approximately. The problem of minimizing the objective function on that sphere, is known as a trust region problem, for which there are efficient algorithms. Also, good software implementations of these algorithms are available. Sphere methods use the direction from the center of inscribed sphere to the point minimizing the objective function on this sphere as a descent direction. For details, see Chapter 9 in (Murty 2010).

(g) **Methods for nonconvex QP** Efficient polynomially bounded algorithms that are guaranteeed to find a local minimum for some special classes of nonconvex QP have been developed by Vavasis.

The sphere methods discussed above have also been adopted to solve nonconvex QPs, and even 0-1 interger programs through nonconvex QP formulations (Murty 2010).

Software

There are several commercially available software packages for solving QPs, including CPLEX, MINOS (available from The Scientific Press as part of either of the algebraic modeling systems AMPL or GAMS), IBMs OSL, MOSEK, and LINDO (Schrage 1987). Also, the online NEOS server for optimization provides a list of QP software, and solvers offering them.

AMPL (Fourer et al. 1993) is a modeling language for mathematical programming which provides a natural form of input for linear, integer, and nonlinear mathematical models besides QP models. The book is accompanied by a PC student version of AMPL and representative solvers, enough to easily handle problems of a few hundred variables and constraints. Versions that support much larger problems are available from the publisher. AMPL uses either MINOS, OSL, or CPLEX solvers for solving QP models.

GAMS (Brooke et al. 1988) is a high-level language that is designed to make the construction and solution of large and complex mathematical programming models straightforward for programmers, and more comprehensible to users of models. It uses the MINOS or the CPLEX solvers for solving QPs, it has also solvers for linear, integer, and nonlinear programming problems. A student version and a professional version are available.

IBM's OSL is a collection of high performance mathematical subroutines for solving linear, integer and quadratic programming models.

MINOS is a Fortran-based computer system designed to solve large scale linear, quadratic, and nonlinear models developed by Murtagh and Saunders in the Department of OR at Stanford University.

The books by Wright (1997) and Vanderbei (2008) provide details about sources for QP software systems.

See

- ▶ Algebraic Modeling Languages for Optimization
- ▶ Complementarity Applications
- ▶ Complementarity Problems
- ▶ Convex Optimization
- ▶ Economics and Operations Research
- ▶ Financial Engineering
- ▶ Interior-Point Methods for Conic-Linear Optimization
- ▶ Linear Programming
- ▶ Nonlinear Programming
- ▶ Portfolio Theory: Mean-Variance Model

References

Bazaraa, M. S., Sherali, H. D., & Shetty, C. M. (1993). *Nonlinear programming theory and algorithms*. New York: Wiley-Interscience.

Brooke, A., Kendrick, D., & Meeraus, A. (1988). *GAMS: A user's guide*. Redwood: The Scientific Press.

Cottle, R. W., Pang, J. S., & Stone, R. E. (1992). *The linear complementarity problem*. Boston: Academic Press.

Dennis, J. B. (1959). *Mathematical programming and electrical networks*. New York: Wiley.

Fourer, R., Gay, D. M., & Kernighan, B. W. (1993). *AMPL A modeling language for mathematical programming*. San Francisco: The Scientific Press.

Glassey, C. R. (1978). A quadratic network optimization model for equilibrium single commodity trade flows. *Mathematical Programming, 14*, 98–107.

Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica, 4*, 373–395.

Kojima, M., Megiddo, N., Noma, T., & Yoshise, A. (1991). *A unified approach to interior point algorithms for linear complementarity problems* (Lecture notes in computer science, Vol. 538). New York: Springer-Verlag.

Liu, Z., & Fathi, Y. (2011). An active index algorithm for the nearest point problem in a polyhedral cone. *Computational Optimization and Applications, 49*, 435–456.

Markovitz, H. M. (1959). *Portfolio selection: Efficient diversification of investments*. New York: Wiley.

Mulvey, J. M. (1987). Nonlinear network models in finance. *Advances in Mathematical Programming and Financial Planning, 1*, 253–271.

Murty, K. G. (1988). *Linear complementarity, linear and nonlinear programming*. Berlin: Heldermann Verlag (Available for public download author's Web site).

Murty, K. G. (2010). *Optimization for decision making: Linear and quadratic models*. New York: Springer.

Murty, K. G., & Kabadi, S. N. (1987). Some NP-complete problems in quadratic and nonlinear programming. *Mathematical Programming, 39*, 117–129.
Schrage, L. (1987). *User's manual for LINDO*. Redwood City, CA: The Scientific Press.
Steinwart, I., & Christmann, A. (2008). *Support vector machines*. New York: Springer.
Vanderbei, R. J. (2008). *Linear programming: Foundations and extensions* (3rd ed.). New York: Springer.
White, F. C. (1983). Trade-off in growth and stability in state taxes. *National Tax Journal, 36*, 103–114.
Wood, A. J. (1984). *Power generation, operation, and control*. New York: Wiley.
Wright, S. J. (1997). *Primal-dual interior point methods*. Philadelphia: SIAM.
Ye, Y. (1991). Interior point algorithms for quadratic programming. In S. Kumar (Ed.), *Recent developments in mathematical programming* (pp. 237–261). Philadelphia: Gordon and Breach.
Zhou, J. L., & Tits, A. L. (1992). *User's guide to FSQP* [Version 3.1]. SRC TR-92-107r2, Institute for Systems Research, University of Maryland, College Park.

Quadratic-Integer Programming

A mathematical program involving the minimization of a quadratic function subject to linear constraints and integer variables.

See

► Quadratic-Programming Problem

Quadratic-Programming Problem

A mathematical-programming problem with a quadratic objective function to be minimized subject to a set of linear constraints.

See

► Mathematical-Programming Problem
► Quadratic Programming
► Wolfe's Quadratic-Programming Problem Algorithm

Quality Control

Francis B. Alt[1] and Scott D. Grimshaw[2]
[1]University of Maryland, College Park, MD, USA
[2]Brigham Young University, Provo, UT, USA

Introduction

While interest in quality goes back to the Middle Ages, quality control as a technical and managerial discipline started to become accepted and widely practiced only in the 1940s and 1950s. Statistical methods of quality control, though developed in the United States and Britain, found their most ardent followers among Japanese businessmen and managers in the post-War decades. Statistical quality control (SQC) consultants such as W.E. Deming became household names in Japan although they were scarcely known in their own countries. During the 1980s, however, there was a renewed interest in quality control in the West, spurred no doubt by the globalization of competition and increasing customer awareness of quality. The Baldrige Performance Excellence Program is one continuous improvement program that has had a great impact on putting quality on top managements' agendas throughout the nation. Even those companies that do not apply for the Malcolm Baldrige Award (MBA) are using its criteria to assess the quality management system of their company, identify any existing gaps and determine necessary improvements.

Organizations establish and employ quality management systems to satisfy the requirements of their customers. A Quality Management System (QMS) is the collection of interrelated activities that a firm uses to define and implement its quality policies and attain its quality objectives. The ISO 9000 standard, the Baldrige Criteria for Performance Excellence and programs such as Six Sigma can be used to provide an infrastructure for implementing a QMS.

To ensure that an organization's suppliers will deliver products and services that conform to its requirements in an era of growing international trade, the International Organization for Standardization (ISO) developed and issued a set of international standards called ISO 9000. The International Organization for Standardization is

a non-governmental organization founded in 1947 and is located in Geneva, Switzerland.

The ISO 9000 series first published in 1987 consisted of ISO 9000 (Guidelines for Selection and Use), three 9001 standards and the 9004 standard. The Series has been revised several times since then. One update, designated ISO 9000:2000, was released in late 2000. In this revision, the three ISO 9001 standards were merged into ISO 9001 and renamed "Quality management systems — Requirements." Although an organization is certified using the requirements given in ISO 9001, the organization frequently states that it is ISO 9000 certified.

The latest revision occurred in 2008 and includes ISO 9000:2008 (Quality management systems — Fundamentals and vocabulary) and ISO 9001:2008 (Quality management systems — Requirements). The revisions to the 2008 version are minor compared to the 2000 version and usually clarify terminology. A listing of the specific changes between the 2000 and 2008 versions of ISO 9001 can be found in Annex B of ISO 9001:2008.

ISO 9001:2008 is based on eight quality management principles: customer focus, leadership, involvement of people, process approach, system approach to management, continual improvement, factual approach to decision making and mutually beneficial supplier relationships. The close relationship between these eight principles and those underlying Total Quality is found in Goetsch and Davis (2009).

ISO 9004:2009 is now called "Managing for the sustained success of an organization — A quality management approach." This standard shows how to continually improve the performance of an organization, thereby leading to its long-term success.

An organization wanting to obtain the registration and approval of its quality system needs to go through extensive documentation and assessment and periodic audits by a registrar. An overview of the registration process is found in Foster (2010). The ISO9000 certification has become essential for companies interested in doing business globally.

The 2000 and 2008 revisions of ISO 9000 provide for improved alignment between ISO 9001 and ISO 14001, where ISO 14000 is directed towards the implementation of an effective environmental system. However, as Foster (2010) states: "ISO 14000 is very riskyfor U.S. firms." The reason is that firms are required to report environmental violations to the U.S. Environmental Protection Agency, thereby exposing themselves to fines and penalties.

The use of statistical techniques plays a key role in the ISO 9001:2008 standard and includes those used for process monitoring and the analysis of supplier and customer satisfaction data. To assist in the selection of the appropriate statistical technique, ISO has issued ISO/TR 10017:2003 (Guidance on Statistical Techniques for ISO 9001:2000).

The Baldrige Criteria for Performance Excellence provide another infrastructure for implementing a quality management system. The criteria also allow an organization to determine the current status of its quality journey so that it can improve its performance and competitiveness. The award was named after Malcolm Baldrige, the 26th Secretary of Commerce. The Malcolm Baldrige National Quality Improvement Act was signed into law in 1987. One statement in the Findings and Purposes Section of the law is: "strategic planning for quality and quality improvement programs, through a commitment to excellence in manufacturing and services, are becoming more and more essential to the well-being of our Nation's economy and our ability to compete effectively in the global marketplace."

Applicants for the Malcolm Baldrige Award are graded on the following seven criteria: Leadership (120), Strategic Planning (85), Customer Focus (85), Measurement, Analysis and Knowledge Management (90), Workforce Focus (85), Process Management and Results(450). The point value for each criterion is in parentheses where the total point value is 1,000. The National Institute of Standards and Technology (NIST), which manages the program with assistance from the American Society for Quality (ASQ), states "[The Criteria] have evolved from having a specific focus on manufacturing quality to a comprehensive strategic focus on overall organizational competitiveness and sustainability." The Criteria are built on a set of interrelated Core Values and Concepts including visionary leadership, customer-driven excellence, and a systems perspective.

The number of categories eligible for the award was increased in 2007, and the categories now comprise: manufacturing businesses (2), service companies, small businesses (3), educational organizations (1), health care organizations (1), and nonprofit organizations such as charities, trade and professional

associations and governmental agencies. The numbers in parentheses are the number of recipients in each category for 2010 which marked the first year in the award's history with three small business recipients. Information about the criteria and the award can be found at the NIST/Baldrige Award Web site.

A third infrastructure for quality improvement is Six Sigma, which originated at Motorola in the mid-1980s. However, its adoption by General Electric, led by former CEO Jack Welch, brought Six Sigma to the forefront as a quality framework that merited serious consideration. Six Sigma has at least three perspectives.

One is a level of process performance such that if the process mean shifts by 1.5 standard deviations, the process will not generate more than 3.4 defects per million opportunities (DPMO), where an opportunity is defined as a chance for nonconformance. Allowing a shift of 1.5 standard deviations recognizes that it is very difficult to hold a process on target. The number of defects per million opportunities (DPMO) can be written as

$$\text{DPMO} = 10^6 \cdot \frac{\text{Number of Defects Observed in a Fixed Period of Time}}{(\text{Number of Units Inspected})(\text{Opportunities for Error})}.$$

To determine the sigma level of a process, calculate $z = 1 - [(\text{DPMO})/10^6]$ and then find $\Phi^{-1}(z) + 1.5$, where Φ^{-1} is the inverse cumulative distribution function of the standard normal. This assumes the critical to quality characteristic (measures what is important to customers) has a normal distribution and that the shift is 1.5 standard deviations.

There is nothing sacred about a shift of 1.5 standard deviations and a 6-sigma quality level. Other shifts and quality levels can also yield 3.4 DPMO. Specifically,

Magnitude of shift	Sigma quality level	DPMO
0.00	4.50	3.4
0.50	5.00	3.4
1.00	5.50	3.4
1.50	6.00	3.4

A DPMO other than 3.4 can also be used. The specific combination that is used depends on the objectives of the organization, the resources available to achieve those objectives and the potential benefit. As Evans and Lindsay (2011) state, "... a change from 3- to 4- sigma represents a 10-fold improvement; from 4- to 5- sigma, a 30-fold improvement; and from 5- to 6- sigma, a 70-fold improvement—difficult challenges for any organization."

A second perspective of Six Sigma is that it can be thought of as a data-driven, highly structured approach for improving processes using the DMAIC methodology, where DMAIC is an acronym for the five sequential phases: Define, Measure, Analyze, Improve and Control.

Suppose that a project has been selected and embraced by a champion: usually a senior manager who owns the project and "provides continuing support for the project and validates the results at the end of the project" (Foster 2010). Project selection could involve developing a business case for each project and assessing its risk and return.

In the Define phase, the problem is clearly defined including its goals and customers. The project team is formed in which a team includes master black belts, black belts, green belts, and others. A detailed description of the typical six sigma roles is found in Munro et al. (2008). The team articulates the project's charter, develops a communication plan for the project's stakeholders and a preliminary timeline to monitor the project's progress. Furthermore, a process map or SIPOC diagram is prepared toidentify suppliers (S), inputs (I), the process (P), outputs (O) and customers (C). Preliminary information is obtained on the voices of the customer (VOC), the business (VOB) and the employee (VOE)—what is important to the customer, business and employees.

The Measure phase involves collecting accurate and reliable data to address the problem being studied. This includes identifying key output variables (Y) and input variables (X) where the expression $Y = f(X)$ is used to denote the process by which the X's are transformed into Y's. The Measure phase includes developing data collection plans for all variables and providing their operational definitions to remove ambiguity. Sometimes, a measurement systems analysis is conducted to determine the consistency of measurements, usually via a gage repeatability and reproducibility (Gage R&R) analysis. Repeatability and reproducibility capture the equipment and operator measurement variation,

Q

respectively. Detailed examples for conducting a Gage R&R analysis can be found in Burdick, Borror, and Montgomery (2005). A major goal of the measure phase is to determine the current performance level and capability of the process that can be used as a baseline for future performance after the Improve phase.

In the Analyze phase, the data is methodically examined to identify sources of variation affecting the key input and output variables and the root causes underlying those sources.

The Improve phase entails acting on the data to bring about lasting process improvement by identifying and addressing the root causes and generating potential solutions. The set of solutions is evaluated and prioritized by using such criteria as the time and cost of implementation and the probability of successful implementation. Design of Experiments is frequently used to determine the levels or settings of the X's that yield the optimal values of the Y's. Once improvements/countermeasures are implemented, the results are analyzed and compared to the baselines from the Measure phase.

In the Control phase, procedures are put in place to ensure that the improvements implemented in the Improve phase are sustained. It is also necessary to determine whether cost savings are being realized. Control charts are frequently used to monitor the modified process. The project is completed by returning the improved process back to the process owner who now assumes responsibility for the improved process.

Foster (2010) presents quality tools that could be used at each phase. The American Society for Quality provides formal certification of green, black and master black belts.

A third perspective of Six Sigma is that it can be thought of as an infrastructure for quality improvement throughout an organization: the relentless and vigorous pursuit of the reduction of variation in all critical processes to achieve continuous and breakthrough improvements that positively impact the profit of the organization and increase customer satisfaction.

Lean Six Sigma combines Six Sigma with Lean, where Lean is also known as Lean operations, Lean manufacturing and Lean production. Lean is embodied in the management philosophy and practices of the Toyota Production System. Two key philosophies underlying Lean are the systematic elimination of waste (muda) and the respect for people. Ohno (1988) identified seven sources of waste: Correction or rework, Motion, Overproduction, Inventory, Conveyance, Overprocessing and Waiting. These seven types of waste can be viewed as opportunities for improvement when non-value added activities are identified. A downside of Lean is the possible occurrence of major disruptions in the supply chain. Lean embraces such principles as one-unit-at-a-time flow versus large batches, as well as demand-driven pull systems versus forecast-based push systems. The concept of Statistical Thinking underlies Six Sigma where Statistical Thinking is based on three principles: (i) all work occurs in a system of interconnected processes; (ii)variation exists in all processes; and (iii) understanding and reducing variation are keys to success (Hoerl and Snee 2003). It is not a question of whether to choose between Lean or Six Sigma, because an organization can use both since they focus on different issues. As Meisel et al. (2007) state: "Flow is negatively affected by excessive variation and rework; quality is negatively affected by unnecessary complexity in a process. The ability to go back and forth between the two methodologies, in a Lean Six Sigma culture, is a real plus and results in accelerated improvement."

The development of the Toyota Production System can be found in Ohno (1988) and Womack, Jones, and Roos (1991). Womack and Jones (2003) provide guidelines for implementing Lean in the context of a lean supply chain while Meisel et al. (2007) provide an overview of Lean Six Sigma.

Many organizations have benefited greatly from employing such QMS as ISO 900, Baldrige and Six Sigma. While some organizations have decided to use only one or two of the three, others have employed all three. The QMS chosen and utilized depends on the needs of an organization at a particular point in its quality passage. Although all three QMSs have common elements, they are different. A comparison of the three QMS can be found at the NIST/Baldrige Web site.

While basic statistical process control (SPC) techniques have remained unchanged for over fifty years, developments are taking place, such as the robustness of existing methods, the application of Bayesian decision theory to control charts, the extension to the multivariate case and the relationship between SQC and engineering control. After a brief

introduction, the basic concepts of control charts and classical SPC methods (Shewhart control charts) are discussed. Cumulative sum and moving average quality control procedures, which are more suitable for detecting small persistent changes in a process, are then described.

History of SQC Procedures

Over the years, SQC techniques have found literally thousands of applications in manufacturing, service, and health care organizations, as well as government and education. The basic SQC techniques are relatively simple to use and are often applied by shop floor personnel with little training in statistical methods. SQC techniques can be broadly divided into two categories: statistical process control and acceptance sampling.

Statistical process control (SPC) involves the use of control charts for monitoring a process at regular intervals of time to detect any problems that may develop in the process and to take corrective action if necessary. Control charts provide a clear and visual representation of the status of the process. Often, problems are identified early and corrective action is taken — thus minimizing economic losses.

Montgomery (2009b) listed five main reasons for the popularity of control charts. Control charts are a proven technique for improving productivity, preventing defects and unnecessary process adjustments, and providing diagnostic and process capability information.

The quality of inputs into a process has a significant bearing on the quality of its output. Raw materials from a supplier or semifinished output from a work station may be the input for a particular process. In such a situation, one may receive a batch of products/materials from a source external to the process and one has to then decide whether to accept or reject the batch based on the quality of a representative sample taken from it. This type of quality control where the decision to accept or reject a batch is based on inspection of a sample of incoming/outgoing goods is termed acceptance sampling. Although this is a well-developed branch of SQC, its use has been criticized because quality cannot be inspected into a product. By the time that a sampling plan is used, the possible production of nonconforming items from an out-of-control process has already occurred. A sampling plan will merely detect with a certain probability, the presence of nonconforming product. Quality control is most effective when it is preventive in nature rather than curative. A comprehensive overview of acceptance sampling is found in Schilling and Neubauer (2009).

Deming (1986) was critical of standard sampling plans because they fail to take into account the cost of inspection and the cost associated with the failure to detect a defective item. This has led to the development of Deming's *kp* rule for a stable process, which calls for either 0% inspection or 100% inspection.

The techniques of statistical process control and acceptance sampling have been around since the 1920s. Walter A. Shewhart of the Bell Telephone Laboratories developed control charts in 1924. In the late 1920s, Harold F. Dodge and Harold G. Romig developed the concept of acceptance sampling, again at the Bell Telephone Laboratories. The development and use of SQC techniques grew rather slowly initially. The exigencies created by the defense requirements of World War II provided an impetus to the use of SQC techniques in industry. The late 1940s and 1950s were a period characterized by the consolidation of technical gains in SQC methodology achieved during the war. While the use of SQC spread in the industrialized and industrializing nations, Japan embraced these techniques with a missionary zeal and showed their potential to the world.

The 1950s saw the use of experimental designs for making sequential product and process improvements. Box (1957) suggested an innovative industrial application of (statistical) design of experiments and termed it Evolutionary Operation (EVOP). EVOP involves introducing small, deliberate variations in a process according to a systematic plan (i.e., according to a designed experiment). After a certain number of trials, sufficient information becomes available to guide future trials (in an evolutionary manner) in order to improve productivity, reduce costs, or both. The key idea being that process improvement can be effected right in the plant during regular production runs rather than in a research lab. The concept of designed experiments was championed in Japanese manufacturing through Taguchi's efforts. Taguchi's rationale was that quality has to be built into the product, and any deviation from the target leads to

quality losses that can and should be minimized. Bendell, Disney, and Pridmore (1989) contains many case studies and Ross (1996) gives an overview of the Taguchi approach. Montgomery (2009a) gives a thorough treatment of experimental design. Refer to Myers, Montgomery, and Anderson–Cook (2009) for a general treatment of response surfaces.

Basic SPC Concepts

In this section, some of the key ideas on which SPC procedures (control charts) were developed are presented. A key concept is the realization that two items manufactured on the same machine under nearly identical conditions may nevertheless have different values for the quality characteristic. This is to say, variability is inherent in all processes and that it is impossible to eliminate all variability in manufactured products irrespective of the precision of the process used. Some processes may exhibitless variability than others, but variability is present nonetheless.

There are two broad sources of process variation: Natural variation, and variation produced by assignable causes. Natural variation is the sum total of the effects of numerous factors impacting on a process, each of which has too small an impact to be identified individually. A stable process is one that exhibits only natural variation, which can be monitored by a control chart. This inherent variation in a process is suggested when design engineers provide specification limits instead of a single precisevalue. When a stable process is operating as planned, it will produce products with the desired quality characteristics within specification limits. In such a situation, the process is said to be in control. Deming (1986) referred to natural variation ascommon cause variation.

The other source of variation in processes is that due to one or more assignable causes. When an assignable cause is present in a manufacturing process (such as wear and tear of a tool, a displaced setting, temperature change, introduction of poor qualityraw materials, or even an inspection gauge needing recalibration), the process is said to be out of control. The presence of assignable causes can be identified through the use of control charts and the process returned to stable in-control operation by removing the assignable cause(s). Deming (1986) referred to assignable cause variation as special cause variation.

When a stable process is in control, measurements on the quality characteristic tend to exhibit behavior which may be taken as the model of satisfactory process behavior. A normal distribution is frequently used as a model of satisfactory process behavior. Thus, a stable, in control process has predictable behavior; that is, its measurements follow the normal distribution. If measurements on the quality characteristic and the corresponding value of some statistic indicate that they do not follow the predictable pattern, the implication is that there must be an assignable cause for this out-of-control status. The purpose of SPC is to detect an assignable cause as early as possible so that corrective action may be taken.

A process is monitored by regularly taking random samples of size n from the output, taking measurements on the selected items, computing a relevant statistic (such as the mean, standard deviation, range, proportion nonconforming, etc.), and plotting the summary statistic for each sample on a chart. The Shewhart (1931) control chart has a center line and control limits and is used for monitoring a single quality characteristic.

If the sample statistic falls between the control limits, the process is stable—that is, there is no signal that special causes are present and the variation observed between time periods is due to natural variation. On the other hand, if the statistic falls outside the control limits or if the statistic shows some kind of pattern, the data suggests that an assignable cause has changed the process and the resulting production does not meet the customer's expectations. The process should be stopped and adjusted to remove this assignable cause before restarting.

For the Shewhart charts, the specification of the probability of false alarms (Type I error) associated with an in-control process can be used to determine the control limits. A false alarm occurs when a statistic plots outside the control limits but the process is actually in control. The probability of a false alarm has to be balanced against the probability of the failure to detect an out-of-control process (Type II error). The forms of the control limits are presented later for a variety of control charts.

The relevant sampling issues in the use of control charts include the determination of sample size,

frequency of sampling, and sampling technique. In terms of the sample size, the larger the sample, the higher the probability of detecting a shift in the process mean. In practice, however, small samples ($n = 4$ or 5) are used. In terms of the frequency of sampling, it would be desirable to specify the frequency depending on how fast a process change could occur. If a process could change rapidly, samples would be taken more frequently. Taking smaller samples more frequently is more common than taking large samples infrequently.

Since one is generally interested in monitoring a process over time to detect any process shifts, it makes sense to use the time order of production as a logical basis for drawing samples. Shewhart suggested that samples (subgroups) should be drawn rationally so that, if assignable causes are present, the chance for within group differences is minimized while the chance for between group differences is maximized. The most commonly used approach for rational subgrouping is to include items produced around the same time in a sample. The proper selection of samples is absolutely essential for gaining as much useful information as possible from control chart analysis. Outputs from different machines, work centers, shifts, or operators should not be pooled together to form a rational subgroup because that would make it impossible to pinpoint assignable causes.

The performance of a control chart is judged in terms of its average run length (ARL), which is the mean number of samples taken before an out-of-control signal is observed. An out-of-control signal for an in-control process is nothing but a false alarm. Thus, the ARL of an in-control process should be large while the ARL of an out-of-control process should be small. It can be shown that the in-control ARL for a Shewhart chart equals $1/\alpha$, where α is the probability of Type I error associatedwith the control chart. The out-of-control ARL is $1/(1-\beta)$, where β is the probability of Type II error. The ARL can also be used to determine the sample size and sampling frequency.

Shewhart charts, based only on the latest sample, are effective at detecting large shifts in the process but are insensitive to small and medium shifts as well as to incremental shifts over time. Other quality control charts, specifically the cumulative sum (CUSUM) chart and the exponentially weighted moving average (EWMA) chart are, however, useful for detecting small to moderate shifts in the process though they are less sensitive for detecting large shifts.

Finally, a word about how the control charts are implemented. Consider the case where the quality characteristic is measured (depth, contents). If the use of the control chart is to detect deviation from a target mean (μ_0) and standard deviation (σ_0), the design of control chart procedures is relatively straightforward. If μ_0 and σ_0 are unknown, they are estimated by sampling a process under allegedly stable conditions of production. Data from 20 or more samples (rational subgroups) are frequently used for estimating the process mean and standard deviation.

Using these estimates, trial limits are computed and the process is retrospectively tested to see whether it was in control when the samples were taken. If necessary, the trial limits are revised by excluding samples that resulted in out-of-control signals using the trial limits and for which a root cause was identified and eliminated. Then, the remaining subgroups are iteratively reviewed using the revised limits to determine if they come from a stable process. The process continues to be monitored in the short-term future using the revised limits.

Univariate control charts can be classified into two groups. If the quality characteristic of interest is a continuous variable, control charts based on measurements of that characteristic are referred to as control charts for variables. If the data being collected come from a discrete variable, the relevant control charts are termed control charts for attributes. Each of these groups contains three main types of control charts, namely, Shewhart, CUSUM, and EWMA charts. The next section presents the univariate Shewhart charts, which is followed by a section describing the univariate CUSUM and EWMA charts.

Shewhart Charts

For a continuous variable with a probability distribution (assumed to be normal in this discussion), generally both the central tendency and the variability of the process should be monitored. The central tendency can be monitored using either individual observations (X chart) or sample means ($\overline{X}$ chart) based on grouped observations. Measures typically used for monitoring variability of grouped

observations are the sample range (R chart) and sample standard deviation (S chart). Typically, control charts are used in pairs such as $\overline{X}$ and R charts (or $\overline{X}$ and S charts). It is recommended that process variability be monitored while monitoring the processaverage and that one analyze the R or S chart first.

Monitoring some quality characteristics may involve the classification of each inspected item as conforming or nonconforming to the specifications, such as can be determined by using a go/no-go gauge. Quality characteristics of this type are referred to as attributes, and the control chart is either based on proportion nonconforming (p chart) or number nonconforming (np chart). The basis for these charts is the binomial distribution. It is assumed that the probability (p) of obtaining a nonconforming unit is constant.

Another set of attribute control charts is based on counting nonconformities in a unit. If each sample unit, say, a square yard of fabric, or a roll of wire of given length, can have a number of different nonconformities, a control chart that monitors thenumber (or average number) of nonconformities per unit (c chart or u chart) may be appropriate. These charts are relevant if the probability of occurrence of a nonconformity is constant and very low relative to the opportunities for its occurrence. These charts are based on the Poisson distribution.

Control Chart for Sample Means ($\overline{X}$ Chart)

The control limits for a chart based on sample means are developed using the result that if observations from a process are normally distributed with mean μ_0 and standard deviation σ_0, then the sample mean based on n observations from the same process is normally distributed with mean μ_0 and standard deviation $\sigma_0/\sqrt{n}$. The control limits are:

$$(LCL, UCL) = \mu_0 \pm z_{\alpha/2}\frac{\sigma_0}{\sqrt{n}} = \mu_0 \pm A\sigma_0$$

where the values of A for $z_{\alpha/2} = 3$ and $n = 2, 3, \ldots, 25$ can be found in Montgomery (2009b). The commonly used $z_{\alpha/2} = 3$ results in $\alpha = 0.0027$ and an in-control ARL of 370. For successive random samples of size n, this control chart can be viewed as repeated significance tests of $H_0 : \mu = \mu_0$ vs. $H_1 : \mu \neq \mu_0$ where α is the Type I error associated with the test.

In many cases the in-control values μ_0 and σ_0 are unknown and replaced with unbiased estimates from a sample believed to be representative of the process during stable in-control operation. The estimates are found by using the sample mean $\overline{X}$ and sample standard deviation S for each of m subgroups to compute the overall mean $\overline{\overline{X}}$ and average standard deviation $\overline{S}$. If the range is used instead of the standard deviation, then R is computed for each subgroup, and average range is denoted by $\overline{R}$. Note that $\overline{S}$ and $\overline{R}$ are biased estimators of the process standard deviation, but unbiased estimators are obtained by using $\overline{S}/c_4$ or $\overline{R}/d_2$. The control limits for $\overline{X}$ charts when $\overline{S}$ is used are:

$$(LCL, UCL) = \overline{\overline{X}} \pm 3\frac{\overline{S}}{c_4\sqrt{n}} = \overline{\overline{X}} \pm A_3\overline{S}$$

and if $\overline{R}$ is used the control limits are:

$$(LCL, UCL) = \overline{\overline{X}} \pm 3\frac{\overline{R}}{d_2\sqrt{n}} = \overline{\overline{X}} \pm A_2\overline{R}$$

where the values for A_2 and A_3 for $n = 2, 3, \ldots, 25$ are given in Montgomery (2009b). For a set of m rational subgroups the control limits can be applied retrospectively to check whether the process was stable. If there are any rational subgroups that resulted in out-of-control signals, the estimates $\overline{\overline{X}}$ and $\overline{S}$ or $\overline{R}$ can be recomputed.

Control Charts for Process Variation (R and S Charts)

If samples of size n are taken at regular intervals, the variation or dispersion of the process is monitored using either the range (R chart) or the standard deviation (S chart).

The range is $R = X_{\max} - X_{\min}$, which has $E(R) = d_2\sigma_0$ and $\sqrt{V(R)} = d_3\sigma_0$ where d_2 and d_3 are constants tabled in Montgomery (2009b) for $n = 2, 3, \ldots, 25$. The control limits for the R chart are

$$\begin{aligned}(LCL, UCL) &= E(R) \pm 3\sqrt{V(R)} = d_2\sigma_0 \pm 3d_3\sigma_0 \\ &= (d_2 \pm 3d_3)\sigma_0 = (D_1\sigma_0, D_2\sigma_0)\end{aligned}$$

where the values of D_1, D_2 for $n = 2, 3, \ldots, 25$ are given in Montgomery (2009b).

Since R is not normally distributed, the control limits for the R chart are based on the premise that most of the probability distribution of R is within three standard deviations of its mean. Thus, if a value of R plots outside the control limits, it is highly likely that the process variability has shifted from the target value σ_0.

Because the computation of the sample standard deviation S is not as straightforward as the range, R charts have been used more often than S charts for monitoring the process variation. However, with modern computing replacing hand calculation even on the shop floor, the use of S charts has increased. The efficiency of the S chart relative to the R chart was discussed in Lowry, Champ, and Woodall (1995). While the S charts are superior, the difference isn't dramatic until $n \geq 10$. The target value for the S chart is $c_4\sigma_0$ and the control limits are $(LCL, UCL) = (B_5\sigma_0, B_6\sigma_0)$, with the values of c_4, B_5, B_6 for $n = 2, 3, \ldots, 25$ found in Montgomery (2009b). As was discussed above with R charts, S is not normally distributed and S charts are based on the notion that most of its probability distribution lies within three standard deviations of its mean.

When σ_0 is unknown, the control limits can be constructed using $\overline{S}$ or $\overline{R}$ using the appropriate constant to create an unbiased estimator. In S charts, the control limits become

$$(LCL, UCL) = (B_5\overline{S}/c_4, B_6\overline{S}/c_4) = (B_3\overline{S}, B_4\overline{S})$$

and the control limits for R charts become

$$(LCL, UCL) = (D_1\overline{R}/c_4,\ D_2\overline{R}/c_4) = (D_3\overline{R},\ D_4\overline{R})$$

where values of B_3, B_4, D_3, D_4 are in Montgomery (2009b). Control charts for S, S^2, and R using probability based limits are found in Ryan (2000).

Control Chart for Individual Observations (*X* and *MR* Charts)

In this case each subgroup is of size $n = 1$ and an unbiased estimator of the in-control mean is obtained by averaging m historical observations from the process during in-control operation. A procedure for estimating the in-control variance is based on the average value of the moving ranges where

$$\overline{MR} = \frac{1}{m-1}\sum_{i=1}^{m-1} |X_{i+1} - X_i|.$$

An unbiased estimator of σ is given by $\overline{MR}/d_2 = \overline{MR}/1.128$, using the value for d_2 when $n = 2$. Using this estimate of σ, the control limits for monitoring the process mean are

$$(LCL, UCL) = \overline{X} \pm 2.66\overline{MR}$$

since $3/1.128 = 2.66$.

To monitor process variability, compute the moving range $MR = |X_{i+1} - X_i|$ and compare to the upper control limit $UCL = 3.267\overline{MR}$. An interesting feature of the moving range chart is that if MR exceeds the upper control limit this could actually reflect a change in the level of the process mean between the two adjacent observations. Amin and Ethridge (1998) discussed the value added by using both X and MR charts versus only the X chart. The ARL of the X and MR charts was investigated by Crowder (1987a).

Control Chart for Proportion of Nonconforming Units (*p* Chart)

Suppose that a process is operating in a stable manner and that the probability of producing a nonconforming unit is p_0, the nominal value. If this process is monitored by taking samples of size n at regular time intervals and counting the number (Y) of nonconforming units in each sample, then the random variable Y (the number of nonconforming units) follows a binomial distribution with parameters n and p_0. The mean of Y is np_0 and its variance is $np_0(1 - p_0)$. If the statistic fraction of nonconforming units (Y/n) is used, its mean is p_0 and its variance is $p_0(1 - p_0)/n$. Plugging these values into the structure of the Shewhart chart yields the following control limits:

$$(LCL, UCL) = p_0 \pm 3\sqrt{\frac{p_0(1-p_0)}{n}}$$

where $LCL = 0$ if the computed value is negative.

If this chart is used retrospectively, the process proportion (p) of nonconforming units is estimated by using data from, say, 20 to 30 rational subgroups. The sample fraction of nonconforming units from each of these subgroups is computed and the average of these sample fractions, $\overline{p}$, is used as an estimate of p_0. Replacing p_0 by $\overline{p}$ results in trial control limits for the p chart. If necessary, the trial limits may need to be revised.

The control limits given above apply only if the sample size n is constant. If the sample size changes from subgroup to subgroup, then the upper and lower control limits will change from subgroup to subgroup by using the relevant sample size (n_t) in place of n in the control limits formulae.

As mentioned earlier, sample sizes required for fraction nonconforming units charts are much larger than the sample sizes used in variables control charts. A rule of thumb is that the sample size should be such that the probability of detecting a specified shift, d, on the next sample is 0.5. Using the normal approximation to the binomial distribution and 3σ limits, the minimum sufficient sample size n is given by:

$$n = \frac{9p(1-p)}{d^2}.$$

Montgomery (2009b) gives a method for choosing n such that $LCL>0$.

Control Chart for Number of Nonconforming Units (*np* Chart)

The control limits of this chart are easily obtained from those of the p chart by noting the number of nonconforming units (Y) in a sample of size n has mean np_0 and variance $np_0(1-p_0)$, where p_0 is the in-control probability of a nonconforming unit.

Control Chart for Number of Nonconformities (*c* Chart)

In some processes the inspection unit or area of opportunity is a single item (e.g. a bolt of fabric) or a group of items (e.g. 50 DRAM chips), but the inspection unit is constant. A common assumption is that the random variable counting the number of nonconformities follows a Poisson distribution where c is the mean and variance of the number of nonconformities.

Thus, 3σ control limits for the c chart with in-control nonconformity rate c_0 are:

$$(LCL, UCL) = c_0 \pm 3\sqrt{c_0}.$$

If LCL is negative, it is set equal to zero. Again, every effort should be made to reduce the level of c. If no standard is given, c_0 is estimated by the average number of nonconformities, $\overline{c}$, in a preliminary sample of size m and trial limits are computed using $\overline{c}$ instead of c_0.

Control Chart for Average Number of Nonconformities per Unit (*u* Chart)

The u chart is used in place of the c chart when areas of opportunity vary in size. For example, in monitoring the daily number of medication errors (wrong patient, amount, time, etc.) at a hospital, the area of opportunity (number of patients) would vary day to day. The plotted statistic is $u = c/(\text{area of opportunity})$, and the control limits are

$$(LCL, UCL) = \overline{u} \pm 3\sqrt{\frac{\overline{u}}{n_i}}$$

where $\overline{u} = \sum c_i / \sum n_i$ and n_i is the size of the ith sample.

Interpreting Control Chart Patterns

If a process is operating only under natural or common causes of variation, the points on the control chart should be randomly scattered around the center line. Nonrandom control chart patterns of any kind signal the possibility of assignable causes beingpresent. So, in addition to the 3σ limits, one can use additional rules to identify nonrandom patterns on the control chart.

In addition to the rule of one point falling outside the three-sigma control limits, Western Electric's *Statistical Quality Control Handbook* (1956) recommends three other rules to identify nonrandom patterns based on dividing the region between the LCL and UCL into six zones each with a width of one

standard deviation. The rules given in the handbook are such that the probability of occurrence of those patterns is approximately equal to the probability of a point falling outside the 3σ limits. These rules are: Two out of three successive points outside 2σ limits (same side), or four out of five successive points outside 1σ limits (same side), or eight successive points on the same side of the center line. The occurrence of any of these patterns indicates the possible presence of an assignable cause. Popular software for quality improvement allows one to incorporate these and other additional rules.

If one uses more than one rule for monitoring a process, the probability of detection of special causes increases but the probability of false alarms increases. Refer to Nelson (1984, 1985), Champ and Woodall (1987) and Walker, Philpot, and Clement (1991) for the effects of using these rules. More information on the interpretation of these and other other patterns is available in Western Electric's *Statistical Quality Control Handbook* (1956).

CUSUM and EWMA Charts

The CUSUM chart is based on a statistic that reflects the cumulative sums of the deviations of each sample statistic relative to the target value. The EWMA chart is based on exponentially weighted moving averages of the current and all past observations. The CUSUM procedure gives equal weight to all of the observations. The EWMA procedure, on the other hand, gives the most weight to the latest observation and exponentially decreasing weights to prior observations.

A CUSUM procedure combines information both from previous samples and the current sample to detect process shifts. CUSUM procedures were first proposed by Page (1954, 1961) and are based on repeated applications of the sequential probability ratio test (see Wald 1947; Johnson and Leone 1962). While CUSUM procedures can be derived for many sample statistics, this section will discuss only the use of CUSUM for individual values.

Since the shift in the process mean may be an increase or a decrease, a CUSUM can be constructed using either a mobile V-mask or an equivalent tabular procedure with a decision interval. Suppose the individual values are independent and normally distributed with mean μ_0 and variance σ_0^2. Then, at time t, compute two cumulative sums, L_t and U_t, for detecting shifts in the process mean:

$$L_t = \min\{0, L_{t-1} + (X_t - \mu_0) + K\}$$
$$U_t = \max\{0, U_{t-1} + (X_t - \mu_0) - K\}$$

where $L_0 = U_0 = 0$, and $K \geq 0$ is referred to as the reference value. The value for K is frequently chosen as halfway between the in-control mean μ_0 and the out-of-control mean μ_1, or $K = |\mu_1 - \mu_0|/2$. An out-of-control signal is given atthe first t or which either $L_t < -H$ or $U_t > H$, where Hs called the decision interval. Typically, the use of $H = 4\sigma_0$ or $H = 5\sigma_0$ yields charts with good ARL properties for detecting shifts of one standard deviation in the mean. If the out-of-control signal is the result of $L_t < -H$ there has been a downward shift in the process mean, whereas if $U_t > H$ signals an upward shift in the process mean. The determination of when the process mean has shifted is accomplished by using counters N^+ and N^-, respectively. In the case of an upward shift, N^+ counts the number of consecutive periods from when U_t first became nonzero at period t^* until $U_t > H$. The shift occurred between periods $(t^* - 1)$ and t^*.

Although CUSUM charts are more effective than Shewhart charts in detecting small to moderate shifts in the process mean, these charts are not as effective in detecting large shifts. Hawkins and Olwell (1998) provide details on choosing H and K based on the desired in-control and out-of-control ARLS, as well as modifications like the fast initial response to improve the sensitivity of the CUSUM at process start-up.

The EWMA chart was introduced by Roberts (1959) and a detailed exposition was given by Hunter (1986). Suppose individual observations X_t are made on a process that is normally distributed with in-control mean μ_0, in-control standard deviation σ_0, and that observations are independent over time. At time t, the EWMA statistic is

$$Z_t = \lambda X_t + (1 - \lambda) Z_{t-1}$$

where $Z_0 = \mu_0$ and $0 < \lambda \leq 1$. Montgomery (2009b) recommended values for λ in the interval $[0.05, 0.25]$. It can be shown that Z_t is a weighted average of all

previous observations where the weights add to 1 and decrease geometrically with the age of the sample. The variance of Z_t is given by

$$V(Z_t) = \sigma_0^2 \frac{\lambda[1-(1-\lambda)^{2t}]}{(2-\lambda)}$$

and as t increases $V(Z_t)$ approaches $\sigma_0^2\lambda/(2-\lambda)$. Therefore, the control limits for the EWMA chart are

$$(LCL, UCL) = \mu_0 \pm 3\sigma_0\sqrt{\frac{\lambda}{2-\lambda}}$$

for moderately large t. For small t, control limits should be based on the exact variance of Z_t.

The EWMA chart is very effective in detecting small shifts in the process mean but is not as effective as the Shewhart chart for detecting large shifts in the process mean. Crowder (1987b, 1989) and Lucas and Saccucci (1990) have provided tables for designing optimal EWMA procedures. An optimal EWMA chart is defined as one with a fixed in-control ARL which has the smallest out-of-control ARL for a specified process shift. For example, by using 2.962σ control limits (rather than 3) and $\lambda = 0.20$, the EWMA chart has an in-control ARL of 500 and an out-of-control ARL of 10.5 for a shift of one σ.

Specification Limits and Process Capability

Once the specification limits for a quality characteristic have been established, one needs to make sure that a stable process is capable of meeting these requirements. A commonly used measure of process capability relative to specification limits is called the Process Capability Index. One such index is C_p, which is defined as

$$C_p = \frac{U-L}{6\sigma},$$

where (L, U) denote the lower and upper specification limits for the quality characteristic and σ is the standard deviation of the distribution of the process. Therefore, C_p is a comparison of the specification limits to the natural variation of the process. The 6σ value represents the process has 99.73% of production within three standard deviations of the mean. If the process has been improved so that C_p is increased, the process has become "more capable" of production within the specification limits. The C_p index is frequently written as $C_p = \text{VOC}/\text{VOP}$, where VOC is the voice of the customer and VOP is the voice of the process.

The C_p index assumes the process is centered at the target mean. An index that penalizes for a mean different from the midpoint of the specification limits is

$$C_{pk} = \min\left\{\frac{U-\mu}{3\sigma}, \frac{\mu-L}{3\sigma}\right\}$$

where μ is the mean of the process distribution. C_{pk} is the more commonly applied summary statistic comparing process capability to specification limits. For a Six Sigma process that is centered at the target mean, the distance from the mean to either specification limit is 6σ and $C_{pk} = 2$. If the process mean shifts by 1.5σ, $C_{pk} = 1.5$ and the DPMO = 3.4. In practice, one needs to estimate μ and σ^2, which can be used to obtain a confidence interval for the capability index. A comprehensive treatment of process capability indices and their statistical properties is in Kotz and Johnson (1993).

See

- ▶ Multivariate Quality Control
- ▶ Total Quality Management

References

Amin, R. W., & Ethridge, R. A. (1998). A note on individual and moving range control charts. *Journal of Quality Technology, 30*, 70–74.

Bendell, A., Disney, J., & Pridmore, W. A. (1989). *Taguchi methods: Applications in world industry*. New York: Springer.

Box, G. E. P. (1957). Evolutionary operation: A method of increasing industrial productivity. *Ó Applied Statistics, 6*, 81–101.

Burdick, R. K., Borror, C. M., and Montgomery, D. C. (2005). *Design and analysis of gauge R&R studies: Making decisions with confidence intervals in random and mixed ANOVA models*, *ASA–SIAM series on statistics and probability*. Philadelphia/Alexandria, VA: SIAM/ASA.

Champ, C. W., & Woodall, W. H. (1987). Exact results for Shewhart control charts with supplementary runs rules. *Technometrics, 29*, 393–399.
Crowder, S. V. (1987a). Computation of ARL for combined individual measurement and moving range charts. *Journal of Quality Technology, 19*, 98–102.
Crowder, S. V. (1987b). A simple method for studying run length distributions of exponentially weighted moving average charts. *Technometrics, 29*, 401–407.
Crowder, S. V. (1989). Design of exponentially weighted moving average schemes. *Journal of Quality Technology, 21*, 155–162.
Deming, W. E. (1986). *Out of the crisis*. Cambridge, MA: MIT Press.
Evans, J. R., & Lindsay, W. M. (2011). *Managing for quality and performance excellence* (8th ed.). Mason: South-Western Cengage Learning.
Foster, S. T. (2010). *Managing quality: Integrating the supply chain* (4th ed.). Upper Saddle River: Pearson Education.
Goetsch, D. L., & Davis, S. B. (2009). *Quality management for organizational excellence: Introduction to total quality* (6th ed.). Upper Saddle River: Prentice-Hall.
Hawkins, D. M., & Olwell, D. H. (1998). *Cumulative sum charts and charting for quality improvement*. New York: Springer.
Hoerl, R. W., & Snee, R. D. (2003). *Statistical thinking improving business performance*. Pacific Grove, CA: Duxbury Thomson Learning.
Hunter, J. S. (1986). The exponentially weighted moving average. *Journal of Quality Technology, 18*, 203–210.
Johnson, N. L. and Leone, F. C. (1962). Cumulative sum control charts—mathematical principles applied to their construction and use. *Industrial Quality Control*, **18**, June, 15–21; July, 29–36; and August, 22–28.
Kotz, S., & Johnson, N. L. (1993). *Process capability indices*. London: Chapman-Hall.
Lowry, C. A., Champ, C. W., & Woodall, W. H. (1995). Performance of control charts for monitoring process variation. *Communications in Statistics: Simulation and Computation, 24*, 409–437.
Lucas, J. M., & Saccucci, M. S. (1990). Exponentially weighted moving average control schemes, properties and enhancements. *Technometrics, 32*, 1–12.
Meisel, R. M., Babb, S. J., Marsh, S. F., & Schliting, J. P. (2007). *The executive guide to understanding and implementing lean six sigma: The financial impact*. Milwaukee, WI: ASQ Quality Press.
Montgomery, D. C. (2009a). *Design and analysis of experiments* (7th ed.). New York: Wiley.
Montgomery, D. C. (2009b). *Introduction to statistical quality control* (6th ed.). New York: Wiley.
Munro, R. A., Maio, M. J., Nawaz, M. B., Ramu, G., & Zrymiak, D. J. (2008). *The certified six sigma green belt handbook*. Milwaukee, WI: ASQ Quality Press.
Myers, R. H., Montgomery, D. C., & Anderson–Cook, C. M. (2009). *Response surface methodology: Process and product optimization using designed experiments* (3rd ed.). New York: Wiley.
Nelson, L. S. (1984). The Shewhart control chart Ñ Tests for special cases. *Journal of Quality Technology, 16*, 237–239.
Nelson, L. S. (1985). Interpreting Shewhart $\bar{X}$ control charts. *Journal of Quality Technology, 17*, 114–116.
Ohno, T. (1988). *Toyota production system: Beyond large-scale production*. Portland, OR: Productivity Press.
Page, E. S. (1954). Continuous inspection schemes. *Biometrika, 3*, 100–115.
Page, E. S. (1961). Cumulative sum charts. *Technometrics, 1*, 1–9.
Roberts, S. W. (1959). Control chart tests based on geometric moving averages. *Technometrics, 1*, 239–250.
Ross, P. J. (1996). *Taguchi techniques for quality engineering* (2nd ed.). New York: McGraw–Hill.
Ryan, T. P. (2000). *Statistical methods for quality improvement* (2nd ed.). New York: Wiley.
Schilling, E. G., & Neubauer, D. V. (2009). *Acceptance sampling in quality control* (2nd ed.). New York: Chapman and Hall.
Shewhart, W. A. (1931). *Economic control of quality of manufactured product*. New York: Van Nostrand.
Wald, A. (1947). *Sequential analysis*. New York: Wiley.
Walker, E., Philpot, J. W., & Clement, J. (1991). False signal rates for the shewhart control chart with supplementary runs tests. *Journal of Quality Technology, 23*, 247–252.
Western Electric Co. (1956). *Statistical quality control handbook*. Indianapolis: Western Electric Corporation.
Womack, J. P., & Jones, D. T. (2003). *Lean thinking: Banish waste and create wealth in your organization* (2nd ed.). New York: Simon & Schuster.
Womack, J. P., Jones, D. T., and Roos, D. (1991). *The machine that changed the world: The story of lean production*. New York, NY: First Harper Perennial edition. (Originally published New York: Rawson Associates, 1990).

Quasi-concave Function

Given a function $f(\cdot)$ and points $x, y \in X$, with $x \neq y$ and X convex, if $f(y) \geq f(x)$ implies that $f(\lambda x + (1-\lambda)y) \geq f(x)$ for all $0 < \lambda < 1$, then f is a quasi-concave function.

See

- ▶ Concave Function
- ▶ Convex Function

Quasi-convex Function

Given a function $f(\cdot)$ and points $x, y \in X$, with $x \neq y$ and X convex, if $-f(y) \geq -f(x)$ implies that $-f(\lambda x + (1-\lambda)y) \geq -f(x)$ for all $0 < \lambda < 1$, then f is a quasi-convex function.

See

- ▶ Concave Function
- ▶ Convex Function

Quasi-reversibility

A property of a node in a queueing network where the state of the system at t_0, the departure process prior to t_0, and the arrival process subsequent to t_0 are independent.

See

- ▶ Networks of Queues

References

Kelly, F. P. (1979). *Reversibility and stochastic networks*. New York: Wiley.

Queue Inference Engine

Richard C. Larson
Massachusetts Institute of Technology,
Cambridge, MA, USA

Introduction

Imagine receiving your monthly bank statement and with it is your personal probability distribution of the times you spend waiting in bank queues. The queues could include both those involving human tellers and automatic teller machines (ATMs). With the technology of the Queue Inference Engine (QIE) such an innovation is now well within the realm of possibility.

Background, Motivation and Overview

The idea of QIE was born in the late 1980s as a result of M.I.T.-based queueing research for Bay Banks, an eastern Massachusetts bank, under the auspices of a grant from the National Science Foundation. Bay-Banks had provided a large sample of transactional data from three of their ATM sites. Their question was, "Which, if any, of these sites is 'too congested' from a queueing point of view, thereby requiring additional ATM capacity at the site?" The transactional data consisted of the times of each ATM transaction by each customer over a period of up to a month.

The first approach to this problem was traditional: estimate arrival rates and service times from the data and then apply well known (steady state) queueing models, such as Erlang's results or the M/G/1 model. Examining the data set, it was realized that a substantial portion of the sample path of the queue had been preserved in the data set. That is, the data set contained a large subset of the information one would have if one tracked the actual queue with "clipboard and stopwatch." For instance, one could identify which customers had been delayed in queue (rather then enter service immediately) by noting the "signature" of a queued customer: a back-to-back service completion and service initiation at the same ATM, during a time when all N ATM's are busy with customers. The customer entering service in such a back-to-back situation was, with probability near one, delayed in queue. Moreover, by following this signature over time-adjacent customers, one could identify the entire set of customers who were delayed in queue during a single congestion (or busy) period, a continuous period of time during which all N servers are continuously busy (except the small intervals during which a customer whose service is completed departs and the new [queued] customer enters service). Further, the information content of the data set was explored to see if it contained additional queue-related information.

Surprisingly, the partial information in the data set allowed a wide variety of queueing measures for each congestion period to be computed efficiently. Assuming Poisson arrivals, these measures include mean queue delay, mean queue length, probability distribution of the queue length and even the transient mean queue length over the course of the congestion period. Later research extended these first results in a number of important directions.

Here, the focus is four-fold: (1) to illustrate the types of physical situations in which the QIE can be applied; (2) to describe one of three alternative

analytical approaches to obtaining QIE results; (3) to guide the reader through the emerging literature in this new and exciting field; and (4) to discuss briefly several implementation experiences.

Illustrative Queue Inferencing Problems

Retail Sales — With most human server retail service systems, one has to collect the transactional data either from a modern POS (Point Of Sale) computer system that does the time marking or from some type of customer sensing device (e.g., pressure sensitive mats, infrared or ultrasonic sensors). For an ATM, the transactional data are recorded automatically, by time marking the moment that a customer inserts a bank card (corresponding to service initiation) and the moment that the ATM ejects the card (corresponding to service completion). The queue statistics generated by the QIE for ATMs may be used by bank managers to monitor the use of ATM sites, thereby providing an accurate method of identifying those sites requiring additional (or fewer) machines. With human servers in retail sales, at banks, post offices, fast food restaurants, etc., the manager would most likely use the results to (1) monitor service levels throughout the day and week, to assure that queue delays are within prescribed quality limits, and (2) to schedule servers optimally over the course of a day and week.

Invisible Queues in Telecommunication Systems — During periods of congestion, many finite capacity telecommunications systems have invisible queues of customers outside the system continuously trying to gain access to it. One example is a k-channel land mobile radio system. Whenever all k channels are simultaneously in use, potential users having a message to transmit (often in the field, in vehicles) continuously monitor channel use and at-tempt to acquire a channel as soon as any one of the current k communications is completed. If at any given time t there are $n(t)$ such potential users awaiting a channel, they constitute a spatially dispersed invisible queue, a queue in which one of the waiting customers enters service very shortly after another customer completes service. This queue can grow in size due to the Poisson arrivals of new potential users desiring channel access. The user entering service next is the one who successfully "locks in" the channel very shortly after termination of a previous message. Service discipline is most likely not first come first served (FCFS). Within the context of the QIE the customer transaction times are the moments of gaining channel access (service initiation) and message termination (service completion). These times can be routinely monitored and recorded by technology, and thus the QIE can be used to deduce queueing behavior. The same argument, perhaps with minor modifications, can be applied to other telecommunications systems, including phone systems from airplanes, mobile cellular telephone systems, standard telephone systems and various digital communications networks.

Using Order Statistics to Derive QIE Performance Measures

The analysis of the queue inferencing problem is rooted in order statistics. Suppose a homogeneous Poisson process is considered with rate parameter $\lambda > 0$. Over a fixed time interval $[0, T]$, it is known that precisely N Poisson events (e.g., queue system arrivals) occur. The N ordered arrival times are $0 \leq X_{(1)} \leq X_{(2)} \leq \cdots \leq X_{(N)} \leq T$ (by implication $X_{(N+1)} > T$). The N unordered arrival times are $X_1, X_2, \ldots, X_N, 0 \leq X_i \leq T (i = 1, 2, \ldots, N)$. Since the Poisson process is time homogeneous, it is well known that the $\{X_i\}$ are independent and uniformly distributed over $[0, T]$. If the Poisson process is non-homogeneous, that is, having time varying rate parameter $\lambda(t)$, then the N unordered arrival times are independent identically distributed (iid) over $[0, T]$, with a PDF (probability density function) proportional to $\lambda(t)$. For simplicity in this discussion, homogeneous processes are focused on.

A Pedestrian Queueing Example — To illustrate queue inference, consider a signalized pedestrian cross walk having fixed cycle time T. Poisson-arriving pedestrians queue at curbside waiting to cross the street during a time interval of length T, and all such queuers are served in "bulk" fashion when the light changes at time T allowing them safely to cross the street. The number N of queued arrivals in any particular light cycle is Poisson distributed with mean λT. Given N, $X_{(i)}$ is the arrival time during $[0, T]$ of the ith queued pedestrian. Here X_i could be viewed as the arrival time of a random queued pedestrian, selected from, say, a photograph of

all queued pedestrians taken just before the light changed at time T. The Poisson arrival assumption is usually thought of as evolving sequentially over time, with customer interarrival times selected in an iid manner from a negative exponential PDF with mean λ^{-1}.

An equivalent way to conduct the pedestrian cross walk experiment is to first select N from the Poisson distribution, and then, for each of the N queuers, to select the arrival time over $[0, T]$ independently from a uniform PDF. This experiment is probabilistically identical to the sequential Poisson arrival realization of the experiment. Suppose now at some intermediate time T, the total number of queued pedestrians $N(t)$ is focused on, defined as the total number of arrivals (at curbside) during the interval $[0, t)$. The following results, derived from the second model of the process, are well known for $N(t)$:

$$\begin{aligned} &E[N(t)] = (t/T)T \\ &\mathrm{Var}[N(t)] \equiv \sigma^2 N(t) = [N(t)/T]y[(T-t)/T] \\ &\Pr\{N(t) = k\} = \binom{N}{k}\left(\frac{t}{T}\right)^k \left(\frac{T-t}{T}\right)^{N-k} \end{aligned} \tag{1}$$

Here the transactional data are N, the total number of queuers, and T, the time until bulk service. From these data transient values of conditional mean have been found, variance and probability distribution of the queue length. Similar logic can be applied to find other performance measures, such as mean delay in queue, that in this case is trivially equal to $T/2$. This is one of the simplest examples of queue inferencing.

Queue Inference in More General Queues —In most queues, customers usually leave one-at-a-time. Their service completion times within a congestion period, recorded as part of the transactional data set, impose a set of inequality constraints on the arrival times of customers who waited in queue. It is this set of inequality constraints that produces precise conditioning information within the general context of order statistics, conditioning information that can be used to deduce queue behavior.

Suppose for a M/D/1 system, a congestion period having precisely $N = 2$ queued customers is examined. For simplicity the service time is one minute per customer and the server's congestion period starts at time zero. Then since $N = 2$, it is known that precisely 2 customers queued during this congestion period and after their service the server was again idle. The busy period for the server is *3* minutes in length, the time to serve 3 customers, the two who queued and the first arrival who initiated the congestion period. From the transactional data, it is known that zero customers arrived during service of the last customer, the third in the congestion period and the second to queue (assuming FCFS queueing). It is known that at least one of the 2 queued customers must have arrived in [0, 1], else there would be no queued customer to select for service commencement at time $T = 1^+$. Similarly, the second queued customer must have arrived by time $T = 2$.

Without the ordering information, the conditional arrival times for the two queued customers are independent uniformly distributed over [0, 2]. In the joint sample space of random variables X_1 and X_2, this corresponds to X_1 and X_2 uniformly distributed over the square of size *2* in the positive quadrant. The sample space can be split into four equal subsquares, (1) $0 \le X_1 \le 1, 0 \le X_2 \le 1$; (2) $1 \le X_1 \le 2, 0 \le X_2 \le 1$; (3) $0 \le X_1 \le 1, 1 \le X_2 \le 2$; (4) $1 \le X_1 \le 2, 1 \le X_2 \le 2$. Without the additional conditioning information regarding service completion times, the outcome of the experiment is equally likely to be within each of the four subsquares, and conditional on being in a subsquare the r.v.'s X_1 and X_2 are conditionally uniformly distributed over that subsquare. But the additional conditioning information from the transactional data imposes the constraints: $X_{(1)} \le 1$, $X_{(2)} \le 2$, thereby eliminating subsquare (4). The a priori probability of this event, called the master probability, is 3/4. *For any number of queued customers N, the master probability is the a priori probability that the order statistics will obey the ordered inequalities imposed by the transactional data*. Once the master probability can be efficiently calculated, most other quantities of interest are easy to compute.

Continuing with the $N = 2$ example, if it is known that X_1 and X_2 fall in subsquare (1), then these two arrival times are uniform identically distributed over [0, 1]. If one falls in [0, 1] and the other falls in [1, 2], that is, subsquare (2) or (3), then the minimum is uniformly distributed over [0, 1] and the maximum is uniformly independently distributed over [1, 2]. This property generalizes: *once it is known that* n_1 *of*

N arrival times are contained in subinterval $[t_k, t_{k+1})$, *where* t_k and t_{k+1} *are the entry into service times of queued customers* k and k_{+1}, *respectively, during the congestion period, then the* n_1 *arrival times are conditionally uniform and independently distributed over* $[t_k, t_{k+1})$ (Larson 1990). These facts allow one to obtain many useful performance characteristics of the queueing system, conditioned on the transactional data.

A simple application of the above observation yields for the PDF of the arrival time A of a randomly queued customer the step-wise decreasing PDF shown in Fig. 1a. The form of this PDF generalizes to arbitrary N: *the marginal PDF for the arrival time of a random queued customer has a stepwise de creasing PDF over the duration of the congestion period, with each step occurring at an end-of-service time* t_i (Hall 1992; Larson 1990).

As a second illustration, the conditional arrival time $X_{(1)}$ of the first queued customer is either the minimum of two uniform independent r.v.'s over [0, 1] *or* simply uniform over [0, 1], with the former situation applying only if the experimental outcome is within subsquare (1). Likewise the conditional arrival time $X_{(2)}$ of the second queued customer is either the maximum of two uniform independent r.v.'s over [0, 1] or simply uniform over [1, 2], the former applying again only within subsquare (1). Recalling that such minimum and maximum r.v.'s have triangular PDFs and combining results appropriately, the PDFs for the arrival times of the two respective customers are immediately obtained, as shown in Fig. 1b. Finally, assuming a FCFS queueing discipline, the queueing delay for the first queued customer is $1 - X_{(1)}$ and the queueing delay of the second is $2 - X_{(2)}$. The corresponding queue delay PDFs are inverted forms of those in Fig. 1b, as shown in Fig. 1c. If a bank knows that you were the second customer in this congestion period, it then has the required information to begin to build your personal PDF for bank queueing. To obtain the monthly PDF, the bank simply has to add together such conditional PDFs for each banking service session that you had during the month.

A General Result in Order Statistics and Application to Queue Inference — Suppose that the service end/start time transactional data are given by the vector $t = \{t_i; i = 1, \ldots, N\}$. In a queue inferencing setting, t_i has two definitions: (1) it is the observed time of departure of the ith departing customer to leave the system during the congestion period; (2) it is also the observed time for the ith customer from the queue to enter service, not necessarily in a FCFS manner. The two sets of individuals comprising the set of arriving customers and the set of departing customers during a congestion period are never identical and may be disjoint. The number of servers M does not enter into the analysis, nor do any distributional properties of the service times (e.g., there is no requirement for iid service times). It is assumed that service times are independent of arrival times. For any given congestion period, the QIE computations may occur any time after completion of the congestion period.

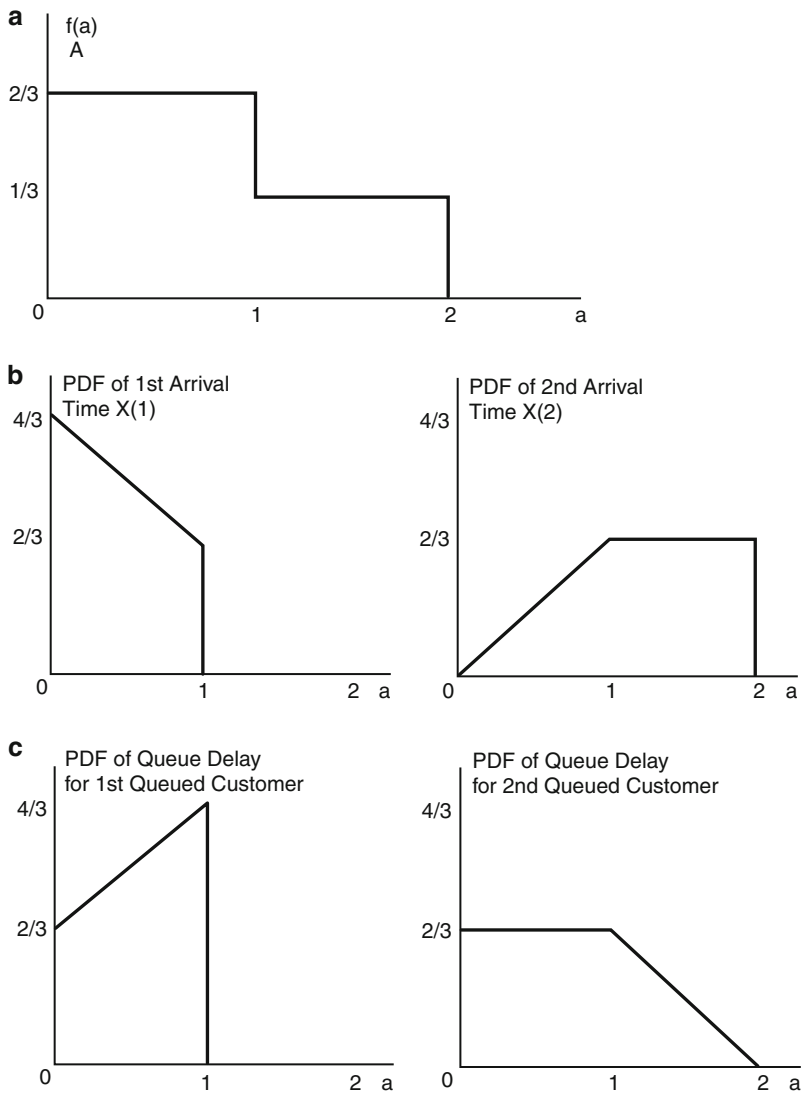
Let $X_1, X_2, = , X_{N(1)}$ be an iid sequence of r.v.'s with values in [0, 1], where the sequence length $N(1)$ is an independent random integer. The goal is to find a computationally efficient algorithm to calculate the probability of an order statistics vector lying in a given N-rectangle,

$$\begin{aligned} \Gamma(\underline{s}, \underline{t}) \equiv \mathrm{P}\{\mathrm{s}_1 < \mathrm{X}_{(1)} \le \mathrm{t}_1, \mathrm{s}_2 \\ < X_{(2)} \le t_2, \ldots, s_N < X_{(N)} \le t_N | N(1) = N\}, \end{aligned} \tag{2}$$

where $\underline{s} \equiv (\mathrm{s}_1, \mathrm{s}_2, \ldots \mathrm{s}_\mathrm{n}), \underline{t}(\mathrm{t}_1, \mathrm{t}_2, \ldots \mathrm{t}_\mathrm{n})$ and without loss of generality the sequences $\{s_i\}$ and $\{t_i\}$ are increasing. Using the fact that the N unordered Poisson arrival times during any fixed time interval $(0, T]$ are iid and now (for convenience) scaling the congestion period to $(0, 1]$, then in our notation, $\Gamma(0, \mathrm{t})$ is the a priori probability that the (unobserved) arrival times $X_{(1)}, X_{(2)}, \ldots, X_{(N)}$ obey the inequalities $X_{(i)} \le t_i$ for all $i = 1, 2, \ldots, N$, that is, it is the "master probability" discussed above. That is, $X_{(i)} \le t_i$ simply says that the ith arriving queued customer must arrive (and enter the queue) before completion of service of the ith departing customer from service.

If the Poisson arrival process is homogeneous, then the unordered arrival times are iid uniform and the rate parameter of the process does not enter the analysis. If the arrival process is nonhomogeneous, then the time-dependent arrival rate parameter $\lambda(t)$ must be known up to a positive multiplicative constant for use in computing the CDF $F(x)$, that is,

$$F(x) = \frac{\int_0^x \lambda(t)dt}{\int_0^1 \lambda(t)dt} \qquad 0(\le x \le 1).$$



Queue Inference Engine, Fig. 1 Probability density functions of arrival times and queueing delays of queued customers

For simplicity, it is assumed that $F(x)$ is strictly monotone nondecreasing continuous. Jones and Larson (1995) have derived an $O(N^3)$ algorithm for finding $\Gamma(\underline{s}, \underline{t})$. Next several of its queue inference applications will be discussed briefly.

The Maximum Experienced Queue Delay — Assume a FCFS queue and consider a congestion period having N customers with observed departure time vector $\underline{t}$, and where the interest is in the maximum time that any of the N customers was delayed in queue, given $\underline{t}$. More precisely, to goal is finding the CDF of the maximum of N nonindependent r.v.'s, the in-queue waiting times of the N queued customers, given $\underline{t}$.

Define $D(\tau/\underline{t})$ as the conditional probability that none of the N customers waited τ or more time units, given the observed departure time data. Set $s_i = \max\{t_i -, 0\}$ for all $i = 1, 2, \ldots, N$. Then $\Gamma(\underline{t} - \tau, \underline{t})$ is the a priori probability that the observed departure time inequalities will be obeyed and that no arrival waits τ or more time units in queue. Clearly,

$$D(\tau|\underline{t}) = \Gamma(\underline{t} - \tau, \underline{t})/\Gamma(0, \underline{t}). \tag{3}$$

Maximum Queue Length — Without any assumption regarding queue discipline, suppose $\underline{s} = \underline{t}^{*k}$ is defined such that $s{*}t_{(i-K)}$ for all

$i = 1, 2, \ldots, N$; $K = 1, 2, \ldots, N$, where a non-positive subscript on t implies a value of zero. These values for $\underline{s}$ imply that each arriving customer i has to arrive after the departure time of departing customer $i = K$ during the congestion period. Now the conditional probability can be computed that the queue length did not exceed K during the congestion period, given $\underline{t}$:

$\Pr\{Q \le k|t\} = P$ {queue length did not exceed K during the congestion period | observed departure time data}, or

$$\Pr\{Q \le K|\underline{t}\} = \Gamma\left(\underline{s}^{*K}, \underline{t}\right)/\Gamma(0, \underline{t}). \tag{4}$$

Probability Distribution of Queue Length — Following the same arguments as in Larson (1990), the $O(N^3)$ computational algorithm can be used to determine for any queue discipline the probability distribution of queue length at departure epochs, and by a balance of flow argument, this distribution is also the queue length distribution experienced by arriving customers.

The Cumulative Distribution of Queue Delay — The algorithm allows computation of points on the conditional in-queue waiting time distribution, given the observed departure data. Again assume a FCFS queue. Define $\beta_i(\tau|\underline{t}) \equiv \mathrm{P}\{j\text{th customer to arrive during the congestion period waited less than } \tau \text{ time units} \mid \text{observed departure time data}\}$. Then setting $\underline{s} = \mathrm{s}^{\mathrm{j}}$, defined so that

$$s_i^j = 0 \; i = 1, 2, \ldots, \; j-1$$
$$s_i^j = \mathrm{Max}\{t_j - \tau, 0\} \quad i = j, \; j+1, \ldots, N$$

it follows that

$$\beta_j(\tau|\underline{t}) = \Gamma\left(\underline{s}^j, \underline{t}\right)/\Gamma(0, \underline{t}). \tag{5}$$

This result allows the determination for any congestion period the probability that a random customer waited less than τ time units, given the observed departure data. Simply compute Eq (5) for each value of j and average the results. Jones and Larson (1995) developed a separate algorithm that allows $O(N^3)$ computation of this average probability of queue delay exceeding some threshold.

Research Literature

Research in queue inferencing is rather extensive. For $O(N^3)$ algorithms for queue performance estimation, see Bertsimas and Servi (1992), Larson (1990), Daley and Servi (1992, 1993); for personnel queue delay PDF, see Hall (1992); for balking, see Larson (1990), Daley and Servi (1993), Jones (1994, 1999); for unknown number of servers, see Kim and Park (2008). Applications of QIE are discussed in Gawlick (1990) and Chandrs and Jones (1994). QIE concepts have been incorporated into a commercial software product Queue Management System (QMS) and has been used by banks, an airline, and the United States Postal Service.

See

- Queueing Theory
- Retailing

References

Bertsimas, D. J., & Servi, L. D. (1992). Deducing queues from transactional data: The queue inference engine revisited. *Operations Research, 40*(S2), 217–228.

Chandrs, K. & Jones, L. K. (1994). "Transactional data inference for telecommunication models," presentation at first annual technical conference on telecommunications R & D in Massachusetts, University of Massachusetts, Lowell, MA.

Daley, D. J., & Servi, L. D. (1992). Exploiting Markov chains to infer queue-length from transactional data. *Journal of Applied Probability, 29*, 713–732.

Daley, D. J., & Servi, L. D. (1993). A two-point Markov chain boundary-value problem. *Advances in Applied Probability, 25*, 607–630.

Gawlick, R. (1990). Estimating disperse network queues: The queue inference engine. *Computer Communication Review, 20*, 111–118.

Hall, S. A. (1992). New directions in queue inference for management implementations. Ph.D. dissertation in Operations Research, Massachusetts Institute of Technology, available as technical report No. 200, Operations Research Center, M.I.T., Cambridge.

Jones, L. K. (1994). "Inferring balking behavior and queue performance from transactional data," technical report, Operations Research Center, M.I.T., Cambridge.

Jones, L. K. (1999). Inferring balking behavior from transactional data. *Operations Research, 47*, 778–784.

Jones, L. K., & Larson, R. C. (1995). Efficient computation of probabilities of events described by order statistics and

applications to queue inference. *INFORMS Journal on Computing, 7*, 89–100.
Kim, Y. B. & Park, J. (2008). New approaches for inference of unobservable queues. In S. J. Mason, R. R. Hill, L. Monch, O. Rose, T. Jefferson, J. W. Fowler (Eds.), *Proceedings of the 2008 Winter Simulation Conference* (pp. 2820–2825).
Larson, R. C. (1990). The queue inference engine: Deducing queue statistics from transactional data. *Management Science* 36, 586–601. (1991) *Addendum* 37, 1062.

Queueing Discipline

Rules used to select the next customer to be served. The first-come, first-served (FCFS) discipline chooses the head of the line customer, the last-come, first-served (LCFS) chooses the tail of the line, and random order chooses the next customer at random, usually equally likely. Other disciplines include putting customers into priority classes.

See

► Queueing Theory

Queueing Networks

► Networks of Queues

Queueing Theory

Daniel P. Heyman
Lincroft, NJ, USA

Introduction

Queueing theory is the study of service systems with substantial statistical fluctuations in either the arrival or service rates. Other names for the subject are stochastic service systems and the theory of mass storage. An example of a stochastic service system from everyday life is a line for bank tellers (human or machine); customers arrive at random, and the transaction lengths will vary depending on the services requested. An example from the world of technology is a computer system; jobs arrive randomly and require different amounts of system resources. An all-too-common source of service-rate variability is a hardware or software crash, which probably occurs randomly even though it might appear that they happen just when you want to use the computer. Looking inside the computer system reveals some more stochastic-service systems. The components (e.g., disk drives, I/O devices, the CPU) have randomly arriving tasks, and the time required to execute a task may be subject to significant statistical fluctuations.

Queueing theory traces its roots back to 1905, starting with the work of A.K. Erlang, who was designing automatic telephone exchanges and needed to know how many calls might be carried simultaneously. Since the calls start at random times and have random durations, the number of calls in progress fluctuates as a stochastic process. Erlang developed several concepts (e.g., birth-and-death processes and statistical equilibrium) about stochastic processes before the formal mathematical theory of stochastic processes was developed. Most of the first 30 years of queueing theory was done in the context of telephony, and telephony continues to be a major consumer of queueing theory. The creation of operations research during World War II led to other applications, such as capacity evaluation of toll booths and port facilities, the order to assign "stacked" airplanes to runways, and scheduling patients in hospital clinics. Areas of extensive current activity include the analysis of production systems, supply chains, call centers, and computer/communication systems.

Basic Notions

The paradigm of a queueing model is that there is a facility consisting of some servers, and customers arrive at the facility to receive some sort of service. Upon arrival, the customers will go to a server if one is available; if all the servers are busy, the customer will either join the queue (also called the waiting room or buffer) or leave. There are two typical reasons that a customer will leave before obtaining service: the queue may be full (in this case the customer is said to be blocked) or the customer may be adverse to waiting in line (in this case the customer is said to have balked).

A departing customer may leave forever or retry after some time. Customers who join the queue may wait until a server is free (the alternative is to bolt from the queue, which is called reneging), and then one of them enters service; the rule that selects the lucky customer is called the queue discipline. Some queue disciplines allow newly arrived customers to displace a customer that is in service, which is called preemption. Once at the server, the customer receives the desired service and then departs. When there is a single facility, the departing customer leaves the system. When there are multiple facilities, the model is called a queueing network, and a routing rule determines where a departing customer goes.

A statistical description of the arrival and service times is almost always given. The objective of the theory is to describe some performance measures, which include the following: the delay of a customer is the time spent in the queue waiting to start service (sometimes it is called the queueing time), the sojourn time of a customer is the total time spent in the facility (it is sometimes called the total waiting time or waiting time), the queue length is the number of customers in the queue, and the number in the system is the queue length plus the number of customers in service. Other performance measures include the number of busy servers, the proportion of blocked customers, and the proportion of non-blocked customers who have a positive delay.

Taxonomy

In the 1950s, D.G. Kendall (1953) introduced a compact notation for describing queueing models. In the current form of the notation, a model is generally described by five parameters, written $A/S/c/K/Q$: A describes the distribution of the times between arrivals, S describes the service time distribution, c is the number of parallel servers, K is the maximum number of customers that can be in queue or in service, and Q is the queue discipline. It is required that $K \geq c$; when $K = \infty$, it is often omitted. Sometimes another parameter is included between K and Q that gives the size of the customer population arrival source, which again is infinite if omitted (see, e.g., Kleinrock 1975).

The following symbols are used for both A and S: M for exponential (Markov), D for deterministic, E_k for Erlang k, H_k for hyperexponential of order k, and PH for phase-type. When the service-times have a general distribution, the letter G is used. The symbol G is used for interarrival times when they are not necessarily independent; independence is emphasized by using the pair of letters GI.

A common queue discipline is FIFO (First-In, First-Out), which is usually taken to be the same as FCFS (First-Come, First-Served). They are identical when there is a single server, but when there are multiple servers, FIFO is stronger than FCFS. Since this is considered the default queue discipline, it is commonly omitted in the notation. Thus, an M/M/1 queue refers to an FCFS single-server queue with a Poisson arrival process and exponentially distributed service times. Other common rules include LIFO (Last-In, First-Out), SIRO (Service In Random Order), and processor sharing. Customers may be partitioned into priority classes, so that more important customers get favored treatment.

There are tacit assumptions that service times are independent and identically distributed (i.i.d.), that service times are independent of interarrival times, that (except for the G case) interarrival times are i.i.d., and that arrivals and services occur one at a time. Other notations are used to represent bulk arrivals or departures, dependencies, and other features.

It is common to denote the mean interarrival time by $1/\lambda$ (λ is the arrival rate) and the mean service time by $1/\mu$. Then $a = \lambda/\mu$ is the rate at which work is brought to the system; it is called the offered load. The offered load is dimensionless, but it is often expressed in Erlangs to honor the contributions of A. K. Erlang. When there are c servers, the load on each server is a/c, which is usually denoted by ρ and is called the traffic intensity.

General Theorems

Most results in queueing theory are formulas for operating characteristics in particular models. There are some theorems that apply to many queueing models, and two of them will be described here. Before doing so, the notion of statistical equilibrium, also called the steady state, is introduced.

Let zero be the time that a queueing system starts operating; for example, for a computer system, it is the time that the installation procedures are completed. Let t be the current time, and let $X(t)$ be the operating characteristic being modeled at time t, for example,

the number in the system t time units after the system started. The initial conditions at time zero usually affect $X(t)$. If there was a backlog of work at time zero, $X(t)$ would be larger than if there was no backlog. The effects of the initial conditions usually decrease as t increases; statistical equilibrium is reached when the effects of the initial conditions have faded away. A queueing system with this property is usually called stable. The mathematical description of this idea starts with defining $p_{ij}(t) = \Pr\{X(t) = j, \text{ given } X(0) = i\}$, and then showing that $p_j = \lim_{t\to\infty} p_{ij}(t)$ exists and is independent of i. Another interpretation of the steady state is that probabilities are not changing with time, that is, the derivatives of $p_{ij}(t)$ with respect to t are zero.

The steady-state solution $\{p_j\}$ is typically much easier to obtain than the transient solution $\{p_{ij}(t)\}$. Interpreting p_j as the long-run proportion of time that X is in state j,

$$p_j = \lim_{t\to\infty}\left[\frac{1}{t}\int_0^t I\{X(s) = j\}\, ds\right] = \lim_{t\to\infty}\left[\frac{T_j(t)}{t}\right] \quad (1)$$

where $I\{\cdot\}$ is the indicator function, which is equal to 1 if the condition is true, 0 otherwise, and $T_j(t)$ is the amount of time $X(s)$ equals j during $(0, t]$. Results of this type are called ergodic theorems, and some conditions on the model are needed for (1) to be valid. The theories of stationary and regenerative processes often are used to prove ergodic theorems.

Among the general theorems, these two are used most frequently.

Little's Theorem (often referred to as Little's Law) – For any stable queueing system, or part of a queueing system, let λ be the arrival rate, L be the steady-state mean number of customers present, and W be the steady-state mean waiting time. If λ and W are well defined, then so is L and $L = \lambda W$.

The use of this theorem is clearly to obviate the need to compute separately both L and W. Three subtler uses of the theorem are the following. It is important to know, and difficult to measure, the average time to get a telephone dial tone (W). It is not so difficult to measure the arrival rate of calls (λ) and the average number of calls waiting to receive dial tone (L). Little's theorem gives an indirect way to estimate the dial tone delay.

In a model with homogeneous servers where all arriving customers are served and the steady-state queue length is finite, suppose the objective is to calculate the mean number of busy servers; this can be very intricate if done in a straightforward way. By considering the servers as "the system," the arrival rate is the arrival rate of customers λ, the "waiting time" of a customer in the system is the service time, with mean $1/\mu$ say, and the "number in the system" is the number of busy servers. Little's theorem shows that our answer is simply λ/μ, which is the offered load. This result shows that at least λ/μ servers are needed. When the queue discipline is such that all servers are equally used, the traffic intensity is the proportion of time a server is busy. The third application concerns comparisons among queue disciplines. Disciplines that produce the same L as FIFO (LIFO and SIRO are examples) must produce the same value of W. Some disciplines use information about service times (e.g., serve the shortest job first, also known as the shortest processing time priority discipline) and reduce L (compared to FIFO); these must also reduce W.

PASTA is an acronym for Poisson Arrivals See Time Averages. Equation (1) shows that p_i can be interpreted as a time average, that is, as the proportion of time i customers are present. A customer is said to see the stochastic process $X(t)$ in state i if $X = i$ just before the customer arrives. Let t_n be the arrival epoch of the nth customer. The state seen upon arrival by the nth customer is $X(t_n)$, and

$$\pi_i = \lim_{N\to\infty}\left[\frac{1}{N}\sum_{n=1}^{N} I\{X(t_n) = i\}\right] \quad (2)$$

is the customer average for state i. A simple example where $\pi_i \neq p_i$ is a $D/D/1$ queue where the times between arrivals are 1 and all the service times are 1/2. Here, $\pi_0 = 1$ yet $p_0 = 1/2$. The PASTA theorem asserts that when the arrivals occur according to a Poisson process, if either π_i or p_i exists, then the other one exists and is equal to it. There is a technical proviso that roughly states that at any time, the future of the arrival process is independent of the past of the X-process. This theorem will be invoked several times in subsequent sections.

Birth-and-Death Queues

The simplest stochastic queueing model has exponentially distributed service and interarrival times. Let $X(t)$ be the number of customers present at time t. When $X(t) = i$, the probability that an arrival will occur in $(t, t + h]$ is $\lambda_i h$, and the probability that a service will be completed in the interval is $\mu_i h$. It is implicit that h is small, and terms of order h^2 can be (and are) ignored. The probability of an arrival and a service occurring, or of more than one arrival or service occurring in $(t, t + h]$, is of order h^2. Implicit in this description is the memoryless property of the exponential distribution. The μ's are called death rates and the λ's are called birth rates because of the interpretation of this model as the size of a population.

The parameter μ_0 will have no role in the analysis; it is convenient to set it equal to zero. Otherwise, it is assumed that $\mu_i > 0$ for $i > 0$ so that the population does not have an a priori lower bound. When $\lambda_n = 0$, a population of size $n + 1$ will not occur after the population drops below $n + 1$, so this is a device to model a finite capacity system. The goal is to obtain $p_i(t) = \Pr\{X(t) = i\}$, where the $\{p_i(0)\}$ are given initial conditions.

Flow of probability argument – Think of probability as a fluid flowing between buckets numbered 0, 1, 2, ..., and $p_i(t)$ as the amount of probability in bucket i at time t. Think of λ_i as the rate at which each molecule of probability in bucket i flows to bucket $i + 1$, and μ_i is the rate at which each molecule of probability flows to bucket $i - 1$. Then the rate at which probability flows from bucket i to bucket $i + 1$ is $\lambda_i p_i(t)$, and the rate of flow in the reverse direction is $\mu_{i+1}\ p_{i+1}(t)$. Since the rate of change of the contents of a bucket is the inward flow rate minus the outward flow rate, for $i = 0, 1, 2, \ldots$,

$$\frac{dp_i(t)}{dt} = \lambda_{i-1}\ p_{i-1}(t) + \mu_{i+1}\ p_{i+l}(t) - (\lambda_i + \mu_i)\, p_i(t) \tag{3}$$

ignoring the $p_{i-1}(t)$ term when $i = 0$. These are called the backward Kolmogorov equations of the birth-and-death process. In the steady-state, the derivatives in (3) equal 0 and the probabilities on the right-hand side equal their steady-state limits, leading to the steady-state balance equations

$$(\lambda_i + \mu_i)p_i = \mu_{i+1}p_{i+1} + \lambda_{i-1}p_{i-1}, \quad i = 0,\ 1,\ 2,\ \ldots \tag{4}$$

These are currently written as second-order difference equations (because they have $i - 1$, i, $i + 1$); the flow-of-probability argument can be used to make them first-order difference equations $\lambda_i p_i = \mu_{i+1} p_{i+1}, i = 0, 1, 2, ...,$ yielding the solution

$$p_i = p_0 \frac{\lambda_0 \lambda_1 \ldots \lambda_{i-1}}{\mu_1 \mu_2 \ldots \mu_i}, \qquad i = 0,\ 1,\ 2,\ \ldots, \tag{5}$$

where p_0 is chosen to make the sum of all the probabilities equal to 1. This can only be done if the sum of the product terms in (5) converges, so some restrictions on the birth and death parameters apply.

M/M/1 queue – In the *M/M/1* queue, the memoryless property of the exponential distribution implies that $\lambda_i = \lambda$ for all i, and $\mu_i = \mu$ for all i, so (5) yields $p_i = p_0\ \rho^i$, where $\rho = \lambda/\mu$ and is called the traffic intensity. The summability condition requires that $\rho < 1$, and then $p_0 = 1 - \rho$ is obtained. (No calculation is required because this was proved via Little's theorem.) The mean of this distribution is $\rho/(1 - \rho)$, and Little's theorem yields $W = 1/[\mu(1 - \rho)]$. The probability that more than k customers are present is ρ^k. The formulas exhibit some of the qualitative features of all stochastic service systems. The operator of the system typically wants to keep the server busy, so the closer ρ is to 1 the better. However, keeping ρ close to 1 will produce very long waiting times, which tend to make customers complain.

To obtain the delay distribution, the memoryless property of the exponential distribution and PASTA are used to argue that at arrival epochs, the remaining service time of the customer in service (if any) is exponential. Thus, a customer who arrives to find i customers in the system has a delay that is distributed as the sum of i independent and identically distributed exponential random variables, which is a gamma (or more specifically, an Erlang) distribution with shape parameter i. PASTA is invoked again to interpret p_i as the probability that an arriving customer sees i other customers. Hence, with probability $1 - \rho$ the delay is zero, and with probability ρ the delay is exponentially distributed with mean $1/(\mu - \lambda)$.

M/M/1/N queue – When at most N customers can be present, set $\lambda_i = 0$ for $i \geq N$. Then (5) yields $p_i = p_0 \rho^i$ and the normalizing condition produces $p_0 = (1-\rho)/(1-\rho^{N+1})$ for $\rho \neq 1$. When $\rho = 1$, $p_i = 1/(N+1)$, for $i = 0,1, \ldots, N$. The condition $\rho < 1$ is not needed here because the normalizing sum has finitely many terms.

M/M/c queue – Here $\lambda_i = \lambda$ for all i, $\mu_i = i\mu$ for $i \leq c$, and $\mu_i = c\mu$ for $i \geq c$.

This is called the Erlang delay or Erlang C model. The quantity $a = \lambda/\mu$ is the offered load; the traffic intensity is $\rho = a/c$. Using (5) leads to

$$p_i = \begin{cases} \frac{p_0 a^i}{i!} & \text{if } 1 \leq i \leq c \\ \frac{p_0 a^i}{c^{i-c} c!} & \text{if } i \geq c \end{cases}$$

where

$$p_0 = \left[\sum_{j=0}^{c-1} \frac{a^j}{j!} + \frac{a^c}{c!(1-\rho)} \right]^{-1}, \qquad a < c.$$

The probability that all servers are busy is given by

$$C(c, a) = \frac{p_0 a^c}{c!(1-\rho)}$$

which is called the Erlang C formula.

The M/M/c/c queue – This is the same as the *M/M/c* queue except that $\lambda_i = 0$ for $i \geq c$. It is called Erlang's loss model or sometimes the Erlang B model. Equation (5) yields

$$p_i = \frac{a^i/i!}{\sum_{k=0}^{c} a^k/k!} \qquad (0 \leq i \leq c)$$

which is a truncation of the Poisson distribution. From PASTA, p_c is the probability that a customer is blocked; it is called Erlang's loss (or sometimes Erlang B) formula and denoted by $B(c, a)$. A remarkable feature of this formula is that it is valid for any service-time distribution with mean $1/\mu$. This is an example of an insensitivity theorem.

M/M/∞ queue – This is the previous model with $c = \infty$. It may be an appropriate model for a self-service system. The steady-state probabilities are given by the Poisson distribution with mean a.

Machine-repair (finite-source) queue – This is a model where there are m machines attended by c mechanics. When the times between machine failures are i.i.d. and exponential, and so are the repair times, then the number of inoperative machines is a birth-and-death process with $\lambda_i = (m-i)\lambda$ and $\mu_i = \min(i, c)\mu$. Equation (5) can be used to calculate the steady-state probabilities, which will be denoted by $p_i(m)$ to emphasize the dependence on m. Since the machine failures do not constitute a Poisson process, it does not necessarily follow that $\sum_{i \geq c} p_i(m)$ is the probability that a failed machine has to wait for repair to begin. A surprising feature of this model is that $p_i(m-1)$ is the probability that i other machines are down at a failure epoch.

When $\lambda_i = 0$ for $i \geq c$, this model is the finite source analog of the *M/M/c/c* queue (sometimes denoted as an *M/M/c/c/m* queue). It is appropriate when the number of sources is not so large that the arrival rate is not diminished when all servers are busy. Let $b = \lambda/\mu$; then Eq. 5 yields

$$p_i(m) = \frac{\binom{m}{i} b^i}{\sum_{k=0}^{c} \binom{m}{k} b^k} \qquad i = 0, 1, \ldots, c$$

An important feature of this model is that this formula is valid for any service-time or interarrival time distributions. This insensitivity result can be extended to sources with different interarrival time distributions. When $m > c$, the probability that all servers are busy at an arrival epoch is

$$\pi_c(m) = p_c(m-1) = \frac{\binom{m-1}{c} b^c}{\sum_{k=0}^{c} \binom{m-1}{k} b^k}$$

which is called the Engset formula.

Balking and reneging – Balking and reneging can be incorporated into the models above by adjusting the birth and death rates. Suppose that customers will balk at a queue of length i with probability b_i. To describe this, replace λ_i with $\lambda_i b_i$. Suppose that when i customers are present, the probability that one of them will renege in a short time interval of length h is $r_i h$. To describe this, replace μ_i with $\mu_i + r_i$.

Output theorem – Let $\Delta(t)$ be the number of departures in an interval of length t in the steady-state. When $\lambda_i \equiv \lambda$, $\Delta(t)$ is a Poisson process. This result is also known as Burke's theorem.

Additional details on these fundamental models are presented in the classic texts of Morse (1958), Cox and Smith (1961), and Prabhu (1965).

Markov Chain Models

The birth-and-death process is the special case of a continuous-time Markov chain in which all transitions are to neighboring states. The added flexibility of the continuous-time Markov chain permits analysis of bulk service and arrival, and some forms of non-exponential service and interarrival times.

A continuous-time Markov chain is described by its rate matrix $\boldsymbol{Q} = (q_{ij})$ where q_{ij} is the rate of making transitions from state i to state j, $i \neq j$, and $q_{ii} = -\sum_{j \neq i} q_{ij}$ by convention, corresponding to the rate of making transitions out of state i. The flow of probability argument is valid for continuous-time Markov chains, and the generalization of (5) is that the row vector of steady-state probabilities, $\boldsymbol{p} = (p_0, p_1, \ldots)$, satisfies the matrix equation $\boldsymbol{pQ} = \boldsymbol{0}$, with the elements of $\boldsymbol{p}$ summing to 1.

Erlang distributions – Erlang devised the following way to use exponential distribution arguments for some non-exponentially distributed random variables. For a random variable with mean $1/\lambda$, imagine that it is constructed by adding k i.i.d. exponential random variables called stages, each having mean $1/(k\lambda)$. The resulting distribution is called an Erlang distribution of order k; it is a gamma distribution with an integer shape parameter, and the density function at t is given by $k\lambda(k\lambda t)^{k-1}e^{-k\lambda t}/(k-1)!$. The standard deviation is $1/(\lambda\sqrt{k})$, which is less than $1/\lambda$ for all $k > 1$, the standard deviation of the exponential distribution with the same mean.

For the $M/E_k/1$ model, the state is taken as the number of customers present and the stage of the customer in service (if any). A customer in stage $j < k$ that completes a service stage moves to the next larger-numbered stage. A customer that completes stage k actually leaves the server. This is called the method of stages. The balance equations for this model are more intricate than for the $M/M/1$ queue, and solving them requires more work. One result is that the steady-state mean delay D is given by

$$D = \frac{k+1}{2k} \frac{\rho}{1-\rho} \frac{1}{\mu}$$

which is the expected delay of the $M/M/1$ model multiplied by $(k + 1)/(2k)$, a number less than 1 for all $k > 1$.

Extended Erlang family of distributions – Extensions of the method of stages are based on the following distributions. A hyperexponential random variable is formed by selecting from among k different exponential distributions according to a probability distribution. Let a_j be the probability of choosing distribution j and $1/\lambda_j$ be the mean of distribution j. Then the density function at t is given by $\sum_{j=1}^{k} a_j \lambda_j e^{-\lambda_j t}$. This produces a larger standard deviation than an exponential distribution with the same mean. This distribution can be used in queueing models in the same way as the Erlang distributions.

Erlang distributions can be pictured as exponential stages in series, and hyperexponential distributions can be pictured as exponential stages in parallel. Replacing these exponential stages by Erlang or hyperexponential distributions yields a broader class of distributions. Repeating this procedure as many times as desired produces the family of general Erlangian distributions. General Erlangian distributions can be pictured as directed graphs. The time required to traverse an edge is an exponential random variable. At each node, an edge is traversed; the choice of which edge to take is determined by a chance event that is independent of how the node was reached. The time to go from the source node to the sink node has a generalized Erlang distribution.

The generalized Erlang distribution is the time to absorption in some continuous-time Markov chain where the initial state is fixed. A phase-type distribution is the time of absorption of a finite continuous-time Markov chain with a single absorbing state, where the initial state can be chosen at random. The representation expands and unifies the extensions of the exponential distribution described above. The family of PH distributions has properties that can be exploited to obtain algorithms for solving

the balance equations when they are used for either the interarrival or service times.

Bulk queues – Heretofore the assumption has been that arrivals and services occur one at a time; this need not be the case. A busload of customers may arrive at a ticket counter, or a bus may serve several customers waiting at a bus stop. The batch sizes may be random variables, for example, partially filled buses. In the bus example, it is natural to assume that the number of customers served will be the smaller than the number of the spaces available and the number of waiting customers. This may not always be valid; when there are large setup costs, a minimum number of customers may be required. Similarly, if there is not enough queueing space for an entire arriving batch, sometimes the entire batch is blocked (some communication systems will not accept part of a message) and sometimes part of the batch is blocked.

In the $M/M/1$ queue with batch arrivals, let c_k be the probability that a batch consists of k customers, and let $\bar{c}$ be the mean batch size. The arrival rate is $\lambda\bar{c}$, so that $\rho = \lambda\bar{c}/\mu < 1$ is required for the steady-state to exist, and Little's theorem yields $p_0 = 1 - \rho$. The number in system goes from i to $i + k$ at rate λc_k and from i to $i - 1$ at rate μ, so the balance equations are

$$\lambda p_0 = \mu p_1,$$

$$(\lambda + \mu)p_i = \mu p_{i+1} + \sum_{k=1}^{i} p_{i-k}c_k, \qquad i = 1,\ 2,\ \ldots$$

These equations can be solved for the probability generating function of the $\{p_n\}$ in terms of the probability generating function of the $\{c_k\}$. For geometrically distributed batch sizes, i.e., $c_k = (1 - \alpha)\ \alpha^{k-1},\ 0 < \alpha < 1$, an explicit solution is

$$p_i = (1 - \rho)[\alpha + (1 - \alpha)\rho]^{i-1}(1 - \alpha)\rho, \qquad i > 0$$

The mean of this probability function is $\rho[(1 - \rho)(1 - \alpha)]^{-1}$, which is the mean of the $M/M/1$ queue with the same traffic intensity multiplied by the mean batch size. A reason why the performance is worse with batches than without is that batches make the arrival process more bursty, i.e., in any interval of time, the batch process will tend to have less epochs where arrivals occur, but those epochs will have several customers appearing at once and the arrivals cluster.

Non-Markovian Queues

The exponential interarrival and service times render the queue length processes Markovian; when these conditions are not valid, there may be other ways to formulate a Markov chain model. The models below have embedded Markov chains instead, which are obtained by restricting attention to selected times.

M/G/1 queue – Here, the service times have a known distribution $G(\cdot)$ with mean $v_g = 1/\mu$, second moment v_{2g}, variance σ_g^2, and Laplace-Stieltjes transform $\tilde{G}(\cdot)$. The mean delay can be obtained without detailed calculations using some general theorems. In the steady-state, let D be the mean delay and Q be the mean queue length. Let R be the mean of the remaining service time of the customer in service (if any) when a customer arrives in the steady-state. Then $D = Qv_g + R$, and Little's theorem asserts that $Q = \lambda D$; solving simultaneously yields $D = R/(1 - \rho)$, where $\rho = \lambda v_g$ is the traffic intensity and the probability that the server is busy. The next two statements are justified by PASTA. When the server is idle, $R = 0$. When the server is busy, renewal theory can be applied to argue that R is the mean of the forward-recurrence time associated with $G(\cdot)$, which is $v_{2g}/2v_g$. Hence, $R = \rho v_{2g}/2v_g$ and $D = \lambda v_{2g}/[2(1 - \rho)]$; this is the Pollaczek-Khintchine formula for the mean delay. It is instructive to write this formula in terms of the squared coefficient of variation $c^2 = \sigma_g^2/v_{2g}$,

$$D = \frac{c^2 + 1}{2}\,\frac{\rho}{1 - \rho}\,\frac{1}{\mu}. \tag{6}$$

From this equation it is easily seen that constant service times produce one-half the mean delay of exponential service times.

The waiting-time distribution is obtained by looking at the number in the system only at service-completion epochs. In any queue where arrivals and services occur one at a time, the number of customers that see state i upon arrival differs from the number of customers that leave state i upon departure by at most 1, so the steady-state distributions at arrival and departure epochs are equal. From PASTA, it follows that looking at only departure epochs will yield time-averaged probabilities.

Let X_n be the number present just after the nth departure, and let A_n be the number of arrivals during the nth service time. To make matters easy, assume that the 0th customer arrives at time 0 and sees an empty system. Then $X_{n+1} = \max(0, X_n - 1) + A_n$, which shows that $\{X_n\}$ forms a Markov chain. When $\rho < 1$, this Markov chain has a limiting distribution independent of the initial state, say $\{\pi_i\}$, with probability generating function $\hat{\pi}(\cdot)$. The transition matrix is

$$\begin{bmatrix} a_0 & a_1 & a_1 & a_3 & a_4 & \cdots \\ a_0 & a_1 & a_2 & a_3 & a_4 & \cdots \\ 0 & a_0 & a_1 & a_2 & a_3 & \cdots \\ 0 & 0 & a_0 & a_1 & a_2 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \tag{7}$$

where $a_i = \int_0^\infty e^{(-\lambda t)}(\lambda t)^i / i!\, dG(t)$ is the probability that i customers arrive during a service time. It is not hard to show that

$$\hat{\pi}(z) = \frac{(1-\rho)(1-z)\tilde{G}(\lambda - \lambda z)}{\tilde{G}(\lambda - \lambda_z) - z}$$

which is another equation associated with Pollaczek and Khintchine. Let $W(\cdot)$ be the waiting-time distribution when the customers are served FIFO, and $\tilde{W}(\cdot)$ be its Laplace-Stieltjes transform. Then π_i is the probability that i customers arrive in an interval of time whose length is distributed as $W(\cdot)$, so $\pi_i = \int_0^\infty e^{-\lambda t}(\lambda t)^i / i!\, dW(t)$; whence $\hat{\pi}(z) = \tilde{W}(\lambda - \lambda_z)$ or

$$\tilde{W}(s) = \frac{(1-\rho)s\tilde{G}(s)}{s - \lambda[1 - \tilde{G}(s)]}.$$

Moments of the queue length and waiting time can be obtained from the transforms $\hat{\pi}$ and $\tilde{W}$.

Another performance measure is the busy period, which commences when the server goes from idle to busy and ends when the server next becomes idle. Let $\tilde{B}(\cdot)$ be the Laplace-Stieltjes transform of its distribution function, which satisfies the implicit equation $\tilde{B}(s) = \tilde{G}[s + \lambda - \lambda\tilde{B}(s)]$. The mean length is finite when $\rho < 1$ and equals $1/(\mu - \lambda)$.

M/G/1/Priority queue – Let the customers be partitioned into K priority classes, where class i has priority over class j if $i < j$. At a service completion epoch, the next customer to enter service is a member of the class with the lowest priority number among those present. Assume priority is non-preemptive, that is, the customer in service is not replaced when a customer with more priority arrives. The notation is the same as above with the script k denoting class k. The service-time distribution for an arbitrary customer is a mixture of the service-time distributions of the classes, and has mean $1/\mu$ and coefficient of variation c^2. The mean delay of a class j customer is

$$D_j = \frac{c^2 + 1}{2} \frac{\rho}{\left(1 - \sum_{i=1}^{j-1} \rho_i\right)\left(1 - \sum_{i=1}^{j} \rho_i\right)\mu},$$
$$j = 1,\ 2,\ \ldots,\ K.$$

Comparison with the FIFO formula (6) shows that class 1 does better with priorities than without, and class K does worse.

Suppose that the cost of keeping members of class j waiting in queue per unit time is C_j and that priorities can be ordered in any arbitrary manner. The way to minimize the waiting costs is to assign priorities in increasing order of $C_j\mu_j$; this is called the $C\mu$ -rule. Taking $C_j = 1$ shows that the overall mean delay is minimized when priorities are assigned in increasing order of mean service time. Letting the number of priority classes become infinite, so that customer i has priority over customer j if its service time is shorter, shows that "serve the shortest job first" is the optimum non-preemptive priority rule. When preemption is allowed, the optimum rule is "serve the job with the shortest remaining-processing-time first."

GI/M/c queue – The *GI/M/c* queue is analyzed similarly to the *M/G/1* queue. There is an embedded Markov chain at arrival epochs. The c-server model is analyzed similarly to the single-server model and is more intricate, so only the single-server model is presented in detail.

Let Y_n be the number of customers present when the nth customer arrives and B_n be the number of customers served during the nth interarrival time. Then $Y_{n+1} = Y_n + 1 - B_n$. Let $A(\cdot)$ and $1/\lambda$ be the distribution function and mean of a generic interarrival time, respectively, $b_k = P(B_n = k)$, and $\hat{b}(\cdot)$ be its probability generating function. Let $1/\mu$ be the mean service time, then $b_n = \int_0^\infty e^{-\mu t}(\mu t)^n / n!\, dA(t)$. The transition matrix is

$$\begin{bmatrix} 1-b_0 & b_0 & 0 & 0 & 0 & \cdots \\ 1-\sum_{k=0}^{1} b_k & b_1 & b_0 & 0 & 0 & \cdots \\ 1-\sum_{k=0}^{2} b_k & b_2 & b_1 & b_0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{8}$$

When $\rho < 1$, this chain has a limiting distribution, say $\{q_i\}$, and $q_i = (1-r)r^i$, where r is the unique root in (0,1) of the equation $\hat{b}(r) = r$. The delay distribution is similar to the M/M/1 model; with probability $1-r$ the delay is zero, and with probability r the delay is exponentially distributed with mean $1/[\mu\ (1-r)]$. The time-averaged probabilities are obtained from $\lambda q_i = \mu p_i$, $i > 0$, which equates upward and downward transition rates.

The analysis with multiple servers is similar. The embedded chain at arrival epochs has the same structure as (8) except care has to be taken to account for the changing number of active servers when the row or column index is less than c.

GI/G/1 queue – For the *GI/G/1/FIFO* queue, for the nth arriving customer, let D_n be the delay, T_n be the arrival time, $U_n = T_{n+1} - T_n$, and S_n be the service time. The departure time is $T_n + D_n + S_n$, so that

$$D_{n+1} = \max(0, D_n + S_n - U_n).$$

When $\rho < 1$, the D_n-process has a proper limit; let $D(\cdot)$ be the limiting distribution. $\{S_i\}$ and $\{U_i\}$ are mutually independent and individually i.i.d., so let $F(t) = P\{S_1 - U_1 \leqslant t\}$. Then

$$D(t) = \int_{-\infty}^{t} D(t-x)dF(x),$$

which is called Lindley's equation. (Note: The recursion for delay D_{n+1} above is also referred to as Lindley's equation; see Lindley 1952). A tractable general solution to this equation is not known; but the equation has proved useful for obtaining qualitative information. For example, it has been shown that the mean delay is no larger than the quotient $\lambda(\sigma_u^2 + \sigma_S^2)/[2(1-\rho)]$, where the σ^2 values are the variances of U_1 and S_1, respectively. When ρ approaches one from below, $D(\cdot)$ approaches an exponential distribution whose mean is the upper bound multiplied by ρ. This is an example of a heavy-traffic approximation.

Numerical Methods

Many results in queueing theory, such as the Pollaczek-Khintchine equations, are given as transforms. These can be numerically inverted using methods that are devised for probability distributions. Abate and Whitt (1992) survey past work and present new algorithms for carrying out numerical inversion.

The models described above have many variants, such as heterogeneous servers, enforced servers idle times, and blocked customers. With suitable probabilistic assumptions, these models are Markovian or possess useful embedded Markov chains. The stationary distribution of the chains is often the desired numerical quantity of interest, or computing it may be an intermediate step. For finite chains, the state-space is often large (hundreds of thousands or more) and sometimes there is a special structure that can be exploited in algorithms for computing the stationary distribution. Stewart (1994) describes many of the methods that can be used.

The *M/M/1*, *M/G/1*, and *GI/M/c* queues are analyzed via Markov chains with infinitely many states. The special structure of these chains is used to obtain analytical results. M. F. Neuts introduced extensions of these models where the special structure is used to obtain algorithmic solutions. The basis for these models is an extension of the Erlang and hyperexponential distributions that can approximate any density with support $[0, \infty]$ and maintain the Markov property. See Neuts (1981) and the entry on matrix-analytic stochastic models.

Control of Queues

The performance models described heretofore can be used to optimize a performance measure subject to resource constraints by trial and error. For example, one might seek the fewest number of servers in an *M/M/c/c* queue setting that make the loss probability no larger than a given target. When a cost structure is introduced to the model, such as server operating and customer waiting costs, optimization models can be formulated directly. When the cost of a lost customer and operating a server are known, instead of choosing the number of servers to achieve a given loss probability, one could choose the number that minimizes the sum of operating and lost customer

costs. Other optimization problems that have been studied include joining rules for heterogeneous customers, server on-off policies, and scheduling rules. Since queueing performance models are frequently analyzed with Markov processes, Markov-decision processes are often used to obtain optimal controls. Stidham (2009) covers these models in depth.

Queueing Networks

In the models above, a customer gets service from one server and then departs. In a queueing network, the departures may join another queue. This may be described as a network, where the nodes represent service centers (a queue and some parallel servers) and a directed arc connects service centers i and j if departures from node i can join the queue at node j. New issues that arise include specifying the routing rule and the disposition of customers that attempt to go to a service center where all the waiting positions are occupied.

Feedback queues – The simplest network is a single node where some departures rejoin the queue. This is called a feedback queue and it can be used to model rework in a manufacturing context. Placing the items to be reworked at the head of the queue is equivalent to using an expanded service time, but placing them at the tail of the queue resembles an increase in the arrival rate.

Tandem networks – A queueing system in which the arrivals first appear at node 1, then go to nodes 2, 3, …, and N in order, and then depart is called a tandem network. This is useful for describing repair or assembly operations. When the arrivals are Poisson, the service times are exponential and independent from node to node, and every arrival to a node will be granted a waiting space, Burke's theorem shows that in the steady state, node i is a birth-and-death queue. Let $p_n(i)$ be the steady-state probability that i customers are present at node n, and let $p(\boldsymbol{i})$, where $\boldsymbol{i} = i_1, i_2, \ldots, i_N$, be the joint probability that i_n customers are at node n, $n = 1, 2, \ldots, N$; then $p(\boldsymbol{i}) = \prod_n p_n(i_n)$. This is a product-form solution, which greatly simplifies computing the joint distribution and shows that the steady-state queue lengths at the various nodes act as independent random variables.

Jackson networks – Let r_{ij} be the probability that a customer will go from node i to node j, and assume that this probability is independent of all other routings of this and all other customers. This is called Markovian routing. The probability that a departure from node i leaves the network is $1 - \sum_j r_{ij}$. A network with birth-and-death assumptions for arrival and service times and Markovian routing is called a Jackson network.

Let α_j be the arrival rate to node j from outside the network and λ_j be the arrival rate including arrivals from other nodes. The arrival rates are related by the traffic equations:

$$\lambda_j = \alpha_j + \sum_i \lambda_i r_{ij}.$$

$j = 1, 2, \ldots, N$, where N is the total number of nodes in the network.

A network is called open if some $\alpha_j > 0$. Open networks have been used to model flexible manufacturing systems and communication networks. When the routing matrix (r_{ij}) is irreducible, the traffic equations have a unique solution for open networks when some row sum of (r_{ij}) is less than 1. Open Jackson networks have a product-form solution based on birth-and-death queues for the steady-state probability $p(\boldsymbol{i})$, with arrival rate λ_j used at node j.

A network is called closed if every $\alpha_j = 0$ and all of the row sums of (r_{ij}) equal 1 and a fixed number of customers, say M, circulate among the nodes of the network. Closed networks have been used to model time-shared computer systems. When the system is almost always heavily loaded, the number of jobs is essentially constant, and they sequentially require work from different components (processors, disks, etc.). Closed networks are also used to model manufacturing systems where the number of pallets in the system is fixed, sometimes called a CONWIP (constant work in process) system. The traffic equations have infinitely many positive solutions for a closed network; if $\boldsymbol{\lambda}$ is a solution, then so is $C\boldsymbol{\lambda}$ for any $C > 0$. There is a product-form solution, $p(\boldsymbol{i}) = C\prod_n p_n(i_n)$, where the $p_n(\cdot)$ are computed from birth-and-death formulas with λ_n taken from some particular solution of the traffic equations. The normalizing constant C must be chosen to make the probabilities sum to 1, which can become computationally intractable. Specifically, there are

${}_{N+M-1}C_M$ ways to place M customers in N service centers, which is roughly 4.25×10^{12} for $M = 100$ and $N = 10$.

There have been many texts written on queueing theory over the years, and some of the classics include Cohen (1969), Cooper (1984), Cox and Smith (1961), Gross et al. (2008), Heyman and Sobel (1982), Kelly (1979), Kleinrock (1975), Morse (1958), Prabhu (1965), Takács (1962), Walrand (1988), and Wolff (1989).

See

- ▶ Birth-Death Process
- ▶ CONWIP
- ▶ Little's Law
- ▶ Markov Chains
- ▶ Markov Decision Processes
- ▶ Markov Processes
- ▶ Matrix-Analytic Stochastic Models
- ▶ Networks of Queues
- ▶ PASTA
- ▶ Pollaczek-Khintchine Formula

References

Abate, J., & Whitt, W. (1992). The Fourier-series method for inverting transforms of probability distributions. *Queueing Systems, 10*, 5–88.

Cohen, J. W. (1969). *The Single Server Queue*. New York: Wiley.

Cooper, R. B. (1984). *Introduction to Queueing Theory* (2nd ed.). New York: North-Holland.

Cox, D. R., & Smith, W. L. (1961). *Queues*. London: Methuen.

Gross, D., Shortle, J. F., Thompson, J. M., & Harris, C. M. (2008). *Fundamentals of Queueing Theory* (4th ed.). New York: Wiley.

Heyman, D. P., & Sobel, M. J. (1982). *Stochastic Models in Operations Research* (Vol. 1). New York: McGraw-Hill.

Kelly, F. P. (1979). *Reversibility and Stochastic Networks*. New York: Wiley.

Kendall, D. G. (1953). Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov chain. *Annals of Mathematical Statistics, 24*(3), 338–354.

Kleinrock, L. (1975). *Queueing Systems* (Vol. 1 and 2). New York: Wiley.

Lindley, D. V. (1952). The theory of queues with a single server. *Mathematical Proceedings of the Cambridge Philosophical Society, 48*(2), 277–289.

Morse, P. M. (1958). *Queues, Inventories and Maintenance*. New York: Wiley.

Neuts, M. F. (1981). *Matrix-Geometric Solutions in Stochastic Models*. Baltimore: Johns Hopkins University Press.

Prabhu, N. U. (1965). *Queues and Inventories*. New York: Wiley.

Stewart, W. J. (1994). *Introduction to the Numerical Solution of Markov Chains*. Princeton: Princeton University Press.

Stidham, S., Jr. (2009). *Optimal Design of Queueing Systems*. New York: Chapman and Hall.

Takács, L. (1962). *Introduction to the Theory of Queues*. New York: Oxford University Press.

Walrand, J. (1988). *An Introduction to Queueing Networks*. Englewood Cliffs: Prentice Hall.

Wolff, R. W. (1989). *Stochastic Modeling and the Theory of Queues*. Englewood Cliffs: Prentice Hall.

R

R Chart

A quality control chart that shows the variations in the ranges of samples.

See

► Quality Control
► $\bar{X}$ Chart

R&D

► Research and Development

RAC

► Operations Research Office and Research Analysis Corporation

Radon-Nikodym Derivative

If measure P is absolutely continuous w.r.t. measure Q, then dP/dQ is called the Radon-Nikodym derivative. For example, if P is a probability measure (distribution) for a continuous random variable and Q is Lebesgue measure, then the Radon-Nikodym derivative corresponds to the probability density function.

Rail Freight Operations

Carl D. Martland
Massachusetts Institute of Technology, Cambridge, MA, USA

Introduction

In North America, the railroad industry is predominantly privately owned and overwhelmingly oriented toward freight rather than passenger operations. An emphasis on profitability and practical problems has been a characteristic of OR/MS applications in the rail industry. This article focuses on freight car utilization, operations planning and control, and line dispatching, the three rail areas that have had the longest history of successful OR/MS applications.

Freight Car Utilization

There are three main issues in freight car utilization: fleet sizing, allocation of equipment to specific services, and distribution of empty equipment. Each of these issues is complicated by the fact that cars are interchanged among the North American railroads. A complex set of rules has been developed for the use of "foreign" cars owned by another railroad or "private" cars owned by a shipper or a car-supply company. The Freight Car Utilization Research/Demonstration Program (1980), jointly established in 1975 by the Association of American Railroads and the Federal Railroad Administration, conducted a series of studies addressing all facets of freight car utilization. Each study was supervised by an

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

industry task force in order to promote consideration of the most important issues, to provide access to data, and to enhance implementation. Task Force I-2 (1977), for example, developed an integrated set of performance measures for car utilization and showed how these measures could be used in both fleet sizing and fleet allocation decisions.

The process of moving cars from an unloading point to a location where they can be reloaded is called empty car distribution. Railroads typically try to hold enough empty cars at each major yard to cover the expected demand in the surrounding region; extra cars can be sent to locations where additional, or more profitable, loads are required. Southern Railway was a leader in the application of linear programming (LP) to car distribution (AAR 1976). They divided the railroad into 37 zones and created monthly supply and demand estimates for each of 13 car types. They then used a linear program to define flow rules that determined where local car distributors should send any extra cars. After Southern merged with Norfolk & Western to form the Norfolk Southern Railroad, this approach to car distribution was expanded to the entire merged system. The program has gone through several revisions and is now run weekly, but the underlying logic is similar to what was originally installed in the late 1960s (Gohring et al. 1993). In 1993, the Norfolk Southern used 70 distribution areas and 15 car types. The revised program addressed shortages and surpluses more realistically and it also provided more flexibility for forecasting supply and demand.

When Philip (1980) surveyed car distribution models, Southern Railway had the most successful intraroad application in the industry. The review by Dejax and Crainic (1987) cited 151 separate studies of the empty vehicle distribution problem, many of which involved railroads, but the Southern Railway's model and two discontinued models were the only ones identified as having been implemented by a railroad.

Two other LP models have been used to overcome problems in the car service rules, which govern car distribution among railroads. The car service rules generally favor the use of system as opposed to foreign cars, which tends to increase fleet sizes and empty mileage. The "Clearinghouse" encouraged member railroads to use each other's cars as if they were system cars, thereby reducing the flow of empty cars (Task Force I-5 1978). A linear program was run on a weekly basis to determine how best to balance the supply of empties among the member railroads. A study conducted in 1977 showed that the percentage of cars reloaded had risen from 55% to 62%, while the percentage of loaded car days increased from 50.7% to 56.2% and the percentage of loaded miles increased from 60% to 64%. The success of the Clearinghouse led to changes in the car service rules that produced similar benefits for the entire industry.

The Multilevel Reload program (I.C.C. Finance Docket 29653, 1981) targeted a particularly expensive and poorly utilized portion of the fleet, namely the two- or three-decked equipment used to transport new automobiles from assembly plants to distribution centers. Historically, separate fleets were assigned to movements between each assembly plant and distribution center, so that the empty mileage equaled the loaded mileage. In 1979, the Multilevel Reload Program combined all the fleets serving General Motors assembly plants (later expanded to the other major manufacturers) and created an industry group that used an LP to minimize empty movements. This led to an immediate, significant reduction in empty mileage of multilevel cars. By 1981, more than 9,000 cars were managed under this program and the ratio of empty to loaded miles had dropped from 0.95 at the outset to 0.55 for the GM fleet and 0.84 for the Ford fleet. The program was still in operation at the end of 1993. It is noteworthy that the analytic application in both of these very successful programs was only a small part of major institutional changes.

Blocking and Scheduling

The consolidation of individual cars into blocks and the movement of blocks on trains is the essence of railroading. A block is a group of cars that move together from one location to another. A block can be carried by one train or by two or more trains. Blocks can be defined based upon the type of traffic, the destination, the priority of the customer, and many other factors. The operating plan specifies how and where cars are classified into blocks, which trains can carry a block, and which blocks are carried by each train. Unfortunately, it has proven impossible so far to define an optimization technique that can solve simultaneously for blocking and scheduling for realistic networks. Successful OR applications have, therefore, focused on specific aspects of operations

planning and paid close attention to the institutional and organizational contexts.

Algorithms have been developed for blocking policy, for assigning blocks to trains, and for scheduling trains. The Automatic Blocking Model and the Train Scheduling System have been widely used in the North American rail industry to determine yard work loads under alternative operating plans (Van Dyke and Davis 1990).

The operating plan implies a trip plan for cars, where a trip plan is the sequence of train movements required to move the car from its origin, through a series of yards, to its destination. For a typical boxcar movement, the shipment will depart on the first available local train after the car is made available by the shipper. The local train is scheduled to bring the car to a nearby yard. The car is next scheduled to move in a particular block that could move on various outbound trains. The car can be scheduled to the first outbound train whose cut-off is later than the car's scheduled arrival time. This process is repeated until the car has a scheduled arrival time at its destination. The first computerized freight car scheduling system was developed by the Missouri Pacific Railroad (1976). In 1991, the rail industry established a plan to implement inter-line car scheduling as a major element of interline service measurement (Ad Hoc Committee 1991).

Predicting the time required for a train connection is the most difficult portion of car scheduling and it is also a critical problem in establishing standards for terminal control systems. Many railroads have terminal control systems that include connection standards, which are usually based upon cutoffs. Problems arise with these systems because it is difficult to maintain a coherent system of cutoffs or connection standards. There is also a conceptual problem. Since operating conditions are variable, better predictions of yard times and connection reliability can be obtained by considering the probability of making a connection, which can be called "PMAKE." It is possible to calibrate PMAKE functions that express the probability of making a connection as a function of the time available, the priority of the car, the priority of the inbound and outbound trains, the time of arrival, and other factors (Martland 1982). PMAKE functions can also be calibrated by convolving yard processing time distributions.

The Service Planning Model uses PMAKE analysis to predict trip times and reliability for a given operating plan and traffic flows (McCarren and Martland 1980). The SPM has been used by the rail industry to set standards for trip time reliability, to evaluate alternative operating plans, and to evaluate merger possibilities.

Network models have also been widely used in railroad rationalization studies. These models are more concerned with traffic flows and line capacity than with the details of operating plans. In many cases, shortest path algorithms are used to route traffic over various proposed networks and the results are displayed graphically (Hornung and Kornhauser 1979).

Line Dispatching

Effective control systems are essential to rail systems. Dispatching is the process of giving trains authority to move along a route, maintaining a safe distance between trains, deciding which sidings to use for meets between trains traveling in opposite directions on a single track railroad, and allowing faster trains to overtake slower trains. In systems where trains routinely run on schedule, train meets and passes are worked out carefully as part of the development of the operating plan. In complex environments, such as is the case in systems with high density passenger operations, it may take a year or more to develop a workable schedule. Various algorithms have been developed to assist in these processes. Here, dispatching involves enforcing the plan and responding to emergencies.

In North American operations, train schedules are seldom planned at this level of detail and departure times vary considerably from one day to the next. As a result, meets and passes are continually different and the dispatching function is very critical to train performance. Several approaches have been take to providing support for dispatching. Sauder and Westerman (1983) formulated the dispatching problem as an integer program, which they solved using a branch and bound solution technique on Southern Railway. Their procedure identifies the optimal set of meets and passes for the upcoming 4 h, that is updated continually and presented as information to the dispatcher. Other systems have used less complex algorithms to plan meets and passes, but implement these plans automatically

unless overridden by the dispatcher. It appears that the savings from making routine decisions in a timely manner (e.g., avoiding delays while the dispatcher is on the phone) are at least as large as the savings from making "optimal" decisions.

Models have been used to study line capacity, line scheduling, and dispatching, many of them building upon the work of Petersen and Taylor (1982). Jovanovich and Harker (1991) developed the SCAN system (1991), which was used by the Burlington Northern Railway to evaluate the potential improvements from advanced line control systems (Smith et al. 1990).

See

- ▶ Linear Programming
- ▶ Network
- ▶ Scheduling and Sequencing

References

Ad Hoc Committee to Develop ETA and Trip Plan Capabilities Among Railroads (1991). *A proposal for systems to support interline service management*. Association of American Railroads Report R-776, Washington, DC.

Association of American Railroads. (1976). *Manual of car utilization practices and procedures*. Association of American Railroads Report R-234, Washington, DC.

Dejax, P. J., & Crainic, T. G. (1987). A review of empty flows and fleet management models in freight transportation. *Transportation Science, 21*, 227–246.

Freight Car Utilization Program. (1980). *Catalog of projects and publications* (2nd ed.). Washington, DC: Association of American Railroads Report R-453.

Gohring, K. W., Spraker, T. W., Lefstead, P. M., & Rarvey, A. E. (1993). *Norfolk Southern's empty freight car distribution system using goal programming*. Presented to ORSA/TIMS Annual Spring Meeting, Chicago, IL.

Hornung, M. A., & Kornhauser, A. L. (1979). *An analytic model for railroad network restructuring*. (Report 70-TR-11, Princeton University, NJ).

I.C.C. Finance Docket 20653. (1981). *Application of certain common carriers by railroad under 49 U.S.C. Paragraph 11342 for approval of an agreement for the pooling of car service*. (see especially verified statements of H.H. Bradley, W.E. Leavers, and J.M. Slivka).

Jovanovic, D., & Harker, P. T. (1991). Tactical scheduling of rail operations: The SCAN I system. *Transportation Science, 25*, 46–64.

Martland, C. D. (1982). PMAKE analysis: Predicting rail yard time distributions using probabilistic train connection standards. *Transportation Science, 16*, 476–506.

McCarren, J. R., & Martland, C. D. (1980). *The MIT service planning model* (Vol. 31). Cambridge, MA: MIT Studies in Railroad Operations and Economics.

Missouri Pacific Railroad. (1976). *Missouri Pacific's Computerized Freight Car Scheduling System*. Federal Railroad Administration Report No. FRA-OPPD-76-5, Washington, D.C.

Petersen, E. R., & Taylor, A. J. (1982). A structural model for rail line simulation and optimization. *Transportation Science, 16*, 192–205.

Philip, C. E. (1980). *Improving freight car distribution organization support systems: A planned change approach* (Vol. 34). Cambridge, MA: MIT Studies in Railroad Operations and Economics.

Sauder, R. L., & Westerman, W. M. (1983). *Computer aided train dispatching: Decision support through optimization*. Atlanta, GA: Norfolk Southern Corporation/Southern Railway Corporation.

Smith, M., Patel, P. K., Resor, R. R., & Kondapalli, S. (1990). Benefits of the meet/pass planning and energy management subsystems of the advanced railroad electronics system (ARES). *Journal of Transportation Research Forum*, 301–309.

Task Force I-2. (1977). *Freight car utilization: Definition, evaluation and control*. Association of American Railroads Report R-298, Washington, DC

Task Force I-5. (1978). *Freight car clearinghouse experiment — Evaluation of the expanded clearinghouse*. Association of American Railroads Report R-293, Washington, DC.

Van Dyke, C., & Davis, L. (1990). *Software tools for railway operations/service planning: The service planning model family of software*. Comprail Conference, Rome, Italy.

RAND Corporation

Gene H. Fisher[1], Warren E. Walker[2] and Michael Rich[1]

[1]RAND Corporation, Santa Monica, CA, USA

[2]Delft University of Technology, Delft, The Netherlands

Introduction

As World War II was ending, a number of individuals, both inside and outside the U.S. government, saw the need for retaining the services of scientists for government and military activities after the war's end. They would assist in military planning, with due attention to research and development. Accordingly, Project RAND was established in December 1945 under contract to the Douglas Aircraft Company. The first RAND report was published in May 1946. It dealt with the potential design, performance, and use of man-made

satellites. In February 1948, the Chief of Staff of the Air Force approved the evolution of RAND into a nonprofit corporation, independent of the Douglas Company.

On May 14, 1948, the RAND Corporation was incorporated as an independent non-profit organization and on November 1, 1948, the Project RAND contract was formally transferred from the Douglas Company to the RAND Corporation. The Articles of Incorporation set forth RAND's purpose:

> "To further and promote scientific, educational, and charitable purposes, all for the public welfare and security of the United States of America."

It accomplishes this purpose by performing both classified and unclassified research in programs treating defense, international, and domestic issues. The current staff numbers nearly 1,000 researchers and about 600 support persons, with about 21% of the researchers being operations researchers, mathematicians, physical scientists, engineers, and statisticians. For much of its history, RAND's research departments were discipline based (e.g., mathematics, economics, physics, etc.). However, research is now carried out by five units that address social and economic policy issues both in the United States and overseas; by three federally funded research and development centers (FFRDCs) that focus on national security; by professors and graduate fellows at the Pardee RAND Graduate School; and by RAND Europe, an independently chartered European affiliate.

This article focuses on RAND's contributions to the theory and practice of operations research. However, RAND has also made major theoretical and practical contributions in other areas, including engineering, physics, political science, and the social and behavioral sciences. A broader and more comprehensive history of RAND's early years is contained in Jardini (1996).

The First Ten Years (1948–1957)

The first decade saw RAND accomplishments ranging from the beginning of the development of systems analysis, which evolved from the earlier more specific and more narrowly focused military operations analysis of World War II, to the creation of new methodological concepts and techniques to deal with problems involving many variables and multiple objectives.

Systems analysis may be defined briefly as the systematic examination and comparison of alternative future courses of action in terms of their expected costs, benefits, and risks. The main purpose of systems analysis is to provide information to decision makers that will sharpen their intuition and judgment and provide the basis for more informed choices. From the beginning it was evident that to be successful, systems analysis would require the conception and development of a wide range of methodological tools and techniques. One of the most important sources of these tools and techniques was the emerging discipline of operations research.

In the early 1950s Edwin Paxson led the project that produced a report entitled *Strategic Bombing Systems Analysis*, which is generally regarded as the first major application of the concept of systems analysis, as well as the source of the name for the new methodology. Among other things, the report advocated the use of decoys to help mask bombers from enemy defenders. This study was a catalyst that stimulated the development and rise of a number of analytical methods and techniques. Some of the more important examples are:

- *Game theory* — Mathematics and game theory were prominent subjects in the early research agendas of Project RAND. Lloyd Shapley, J.C.C. McKinsey, Melvin Dresher, Martin Shubik, Rufus Isaacs, and Richard Bellman were among the numerous early RAND contributors to this area, while John Williams and Herman Kahn played an important role in popularizing some of the simpler aspects of game theory. John von Neumann, who is often cited as the father of game theory, and Oskar Morgenstern, who linked game theory to economic behavior, were active RAND consultants, as were many others with connections to major universities.
- *Enhanced computer capabilities* — The Paxson project required computer capabilities beyond those available at that time. This stimulated developments that led to the building of the JOHNNIAC digital computer, which became fully operational in the first half of 1953. Based on a design by John von Neumann, it was one of the six "Princeton class" stored programming machines, and the first operational computer with core memory in the world. After using the JOHNNIAC to implement the first distributed on-line time-shared computer system (1960), RAND built the JOHNNIAC Open Shop System (JOSS),

one of the first interactive programming languages for individual users.

- *Dynamic Programming* — The Paxson project also demanded the examination, through dynamic programming, of key strategic bomber components (e.g. decoys) in the context of an overall enhanced strategic capability. This, along with the demands of other projects in the early 1950s, provided a significant part of the motivation for the development of the mathematical theory of dynamic programming. Richard Bellman, together with a few collaborators, almost exclusively pioneered the development of this theory. The first RAND report on dynamic programming was published in 1953. Bellman's well-known book (*Dynamic Programming*) followed in 1956, and his book with Stuart Dreyfus (*Applied Dynamic Programming*) was published in 1962.

A second large systems analysis study of this period was a study led by Albert Wohlstetter on the selection and use of strategic air bases. It developed basing and operational options for improving the survivability of SAC forces and helped shift the focus of strategic thinking in the United States toward deterrence based on a secure second-strike force.

Another major effort beginning in the 1950s that led to the development of operations research tools was research on logistics policy issues. RAND's involvement with Air Force logistics stressed the demand for spare parts and the need for logistics policies that could cope with demand uncertainty. Major players in this effort were Stephen Enke, Murray Geisler, James Peterson, Chauncey Bell, Charles Zwick, and Robert Paulson. The key analytical issue here was the examination of alternative policy issues under conditions of strategic uncertainty. Early research used "expected value" analysis. Later, RAND researchers developed and used more sophisticated methods, such as:

- The use of sensitivity analysis to determine what areas of uncertainty really matter in final outcomes;
- Iteration of the analysis across several relevant future scenarios to seek problem solutions that are robust for several of the possible (uncertain) scenarios;
- Given the outcomes of the above, design R&D activities that will (1) reduce key areas of uncertainty, (2) provide hedges against key uncertainties, (3) preserve options for several possible courses of action, any one of which might be used when the future environment becomes less uncertain.

Finally, the first decade witnessed the development of a number of methods and techniques that were useful across a range of RAND projects and elsewhere. Some important examples are:

- *Problem Solving with the Monte Carlo Techniques* — Although not invented at RAND, the powerful mathematical technique known as the Monte Carlo method received much of its early development at RAND in the course of research on a variety of Air Force and atomic weapon problems. RAND researchers pioneered the use of the method as a component of a digital system simulation. RAND's main contributions to Monte Carlo lie in the early development of two tools: generating random numbers, and the systematic development of variance-reduction techniques.
- *Cost Analysis* — David Novick, as head of RAND's Cost Analysis Department, developed the fundamental building blocks of cost analysis during the 1950s and 1960s. Gene Fisher documented this work in his seminal book *Cost Considerations in Systems Analysis*.
- *A Million Random Digits with 100,000 Normal Deviates* — The tables of random numbers in this 1955 report have become a standard reference in engineering and econometrics textbooks and have been widely used in gaming and simulations that employ Monte Carlo trials. It is one of RAND's best selling books.
- *Approximations for Digital Computers* — This book, by Cecil Hastings and J.P. Wong, Jr., contained function approximations for use in digital computations of all sorts.
- *Systems Development Laboratory* — This laboratory was set up under the leadership of John Kennedy to help examine how groups of human beings and machines work under stress. The work ultimately led to the formation of the System Development Corporation.

The Second Ten Years (1958–1967)

This period in RAND's history witnessed the beginning of the evolution of systems analysis into

policy analysis. It also witnessed a branching out from national security research into research on domestic policy issues.

One of the most important dimensions of change as systems analysis evolved into policy analysis was the *context* of the problem being analyzed. Contexts became broader and richer over time. What was taken as given (exogenous to the analysis) before became a variable (endogenous to the analysis) later. For example, in the typical systems analysis of the 1950s and early 1960s, many considerations were not taken into account very well, often not at all: for example, political, sociological, psychological, organizational, and distributional effects. Thus, as systems analysis evolved into policy analysis, the boundaries of the problem space expanded. This had important implications for changes in concepts and methods of analysis. For example, with respect to models, the demands of the expanded boundaries of the problem space could not be met by merely trying to make models used in policy analysis bigger and more complex. Of equal importance, was the development of sophisticated strategies for the development and use of models.

While the evolution of systems analysis into policy analysis did not progress very far during this period, there are several areas of RAND research that were conducted in broader contexts than were typical of the 1950s. These included Ed Barlow's Strategic Offense Forces Study (SOFS), Bernard Brodie's work on the development of a strategy for deterrence in the new age of abundant nuclear weapons and ballistic missiles, Herman Kahn's analysis of civil defense in the event of a nuclear war, and Charles Hitch and Roland McKean's book *Economics of Defense in the Nuclear Age*, which espoused the view that the economic use of scarce resources should be a critical aspect of defense planning. This view was adopted by Secretary of Defense Robert McNamara and led to RAND's involvement in the development of the defense Planning, Programming, and Budgeting System (PPBS).

In addition to policy strides like those discussed above, the second ten years witnessed further development of methodological tools for quantitative analysis — primarily operations research tools. Major advances were made at RAND in the areas of mathematical programming, queueing theory, computer simulation, stochastic processes, and operational gaming.

Linear programming (LP) was probably RAND's most important and most extensive contribution to the theory and practice of operations research as well as to economic decision making. Between 1947 and 1952, George Dantzig and others who worked in the Pentagon on the Air Force's Project SCOOP developed the simplex method and other basic features of LP. Dantzig moved to RAND in 1952. During the following decade, RAND was the world's center of LP developments. In addition to methodological developments by Dantzig and other RAND employees and consultants (e.g., the development of the dual simplex algorithm), there was seminal work on classic problems like production planning and the traveling salesman problem. In addition, most of the pioneering computer programming of LP algorithms (e.g., the first code for the revised simplex method) was carried out by William Orchard-Hays and others at RAND. Much of the work of this period is captured in Dantzig's book, *Linear Programming and Extensions*, published in 1963.

Seminal work in other areas of mathematical programming also took place at RAND during the 1950s and 1960s. Ralph Gomory developed the first integer programming algorithms; Philip Wolfe, George Dantzig, and Harry Markowitz initiated work on quadratic programming; and George Dantzig and Albert Madansky initiated work on stochastic programming.

Five other examples of RAND work in the "tools" area during this period are worthy of note:

- *Simulation* — In the early 1960s, after doing complex simulation modeling "the hard way," Harry Markowitz and Herb Karr developed SIMSCRIPT, a programming language for implementing discrete event simulation models. This work led in l968 to SIMSCRIPT II, which introduced ideas that eventually inspired the modern object-oriented programming paradigm.
- *Artificial intelligence* (*AI*) — The man–machine partnerships explored in the Systems Research Laboratory gained new impetus as Allen Newell, Herb Simon, and Cliff Shaw began to construct a general problem solving language that employed symbolic (non-numerical) processes to simulate human thinking on a computer. One of their initial efforts to carry out a "theory of thinking" involved

programming computers to play chess. On a broader scale, this research resulted in several information processing languages (e.g., IPL V), which were similar to LISP and were used in some of the early AI computer work.

- *Flows in networks* — In 1962, Lester Ford, Jr. and Delbert R. Fulkerson, RAND mathematicians, published the first unified treatment of methods for dealing with a variety of problems that have formulation in terms of single commodity flows in capacity-constrained networks. Their book, entitled *Flows in Networks*, introduced concepts (e.g., "max-flow/mincut") and algorithms (e.g., "out-of-kilter") that have been used to treat network problems ever since.
- *Branching processes* — The notion of a branching process concerns individuals from some population that can reproduce and die (become extinct), subject to some probabilistic laws of chance. The theory of branching processes is the mathematical formulation of the development of that population subject to those laws of chance. RAND mathematician Ted Harris was preeminent in this field. His book entitled *The Theory of Branching Processes* was first published in 1964. The theory has been applied to problems associated with such diverse issues as neutron diffusion, cosmic rays, gene attributes, and biological populations.
- *Multi-echelon Inventory theory* — In 1966 the METRIC model was documented by Craig Sherbrooke. This was a pioneering development in dealing with inventory systems having hierarchies of stockage locations.

The year 1964 saw the publication of the first of several RAND books by mathematician Edward Quade, who played a major role in developing and disseminating the methodology of systems analysis and (later) policy analysis. *Analysis for Military Decisions* documents an intensive five-day course that RAND offered to military officers and civilian decision makers in 1955 and 1959.

The Third Ten Years (1968–1977)

This period in RAND's history saw an acceleration of many of the trends begun in the previous ten years. One of these trends involved the development of improved procedures for the use of expert judgment as an aid to military decision making. The Delphi procedures grew out of this effort. These procedures incorporate anonymous response, iteration and controlled feedback, and statistical group response to elicit and refine group judgments where exact knowledge is unavailable.

Other trends involved the continued evolution of systems analysis into policy analysis and an increasing emphasis on analyzing major domestic research issues. Important in the last two trends was the establishment of the New York City-RAND Institute (NYCRI). Important RAND research efforts during 1968–1977 include the work performed at the NYCRI and policy analysis studies for the government of the Netherlands.

The New York City-RAND Institute — In 1968, RAND began a long-term relationship with the City of New York to tackle problems in welfare, health services, housing, fire protection, law enforcement, and water resources. The NYCRI was formally established in 1969. The research staff evaluated job training programs, suggested solutions to shortages of nurses in municipal hospitals, helped change rent control, altered fire department deployment policies, reallocated police manpower, and helped improve Jamaica Bay's water quality.

The most successful of the NYCRI's projects was the one devoted to improving the operations and deployment of the Fire Department of New York. In 1968, the major problem facing the Department was the rising alarm rate. Its increasing workload was not significantly relieved by adding more men and equipment; nor were traditional methods of fire company allocation, dispatching, and relocation working. The Institute's studies altered the way the Department managed and deployed its men and equipment and operated its dispatching system. An integral part of the research involved creation of a wide variety of computer models to analyze and evaluate deployment, which led to the formulation of new policies. Warren Walker and Peter Kolesar were awarded ORSA's 1974 Lanchester Prize for a paper that described how mathematical programming methods were applied to the problem of relocating available fire companies to firehouses vacated temporarily by companies fighting fires. The entire body of work from this project is documented in Walker et al. (Walker et al. 1979).

Policy Analysis Studies for the Dutch Government — Reflecting an increasing interest in

doing policy analysis studies in international contexts, RAND started working for the Dutch government in the 1970s.

One important study was concerned with protecting an estuary from floods. In April 1975, RAND began a joint research venture with the Dutch government to compare the consequences of three alternative approaches for protecting the Oosterschelde, the largest Dutch estuary, from flooding. Seven categories of consequences were considered for each alternative: financial costs, ecology, fishing, shipping, recreation, national economy, and regional effects. Within each category, several types of consequences were considered. In June 1976, the Dutch Parliament adopted one of the alternatives based in large part on the results of the RAND study: to build a 10 km, multi-billion dollar storm surge barrier with large movable gates across the mouth of the estuary. The study required the development of sophisticated computer models of estuaries and coastal seas.

A second study was focused on improving water management in the Netherlands. Begun in April 1977, the Policy Analysis for the Water Management of the Netherlands (PAWN) project was conducted jointly by RAND, the Dutch Government, and the Delft Hydraulics Laboratory. It analyzed the entire Dutch water management system and provided a basis for a new national water management policy for the country. It developed a methodology for assessing the multiple consequences of possible policies, and applied it to generate alternative policies and to assess and compare their consequences. Considering both research and documentation, it directly involved over 125 person-years of effort (including a considerable amount of support from Dutch organizations). The project won a Franz Edelman Award for Management Science Achievement in 1984.

In 1970, RAND established one of the original eight public policy graduate schools in the United States, the Pardee RAND Graduate School (PRGS). PRGS is the world's largest doctoral program in the field. PRGS doctoral fellows take advanced courses in such fields as economics, statistics, political science, and the social sciences. They also work part-time as members of RAND's interdisciplinary research teams, which is how they earn their fellowships. This combination of advanced course work and on-the-job training is unique. Fellows obtain research training in RAND's classrooms, and get to apply it to real problems with RAND mentors and clients.

The Fourth Ten Years (1978–1988)

This period witnessed a number of major institutional milestones. Some examples:

- In 1982 the joint RAND/UCLA Center for Health Policy Study was funded by the Pew Memorial Trust. A year later RAND and UCLA established a joint Center for the Study of Soviet International Behavior.
- In 1984 a new Federally Funded Research and Development Center — the National Defense Research Institute (NDRI) — was established, funded by the Office of the Secretary of Defense.
- The Arroyo Center, the Army's Federally Funded Research and Development Center for studies and analysis, was established at RAND in 1984.
- The Center for Policy Research in Health Care Financing, sponsored by the U.S. Department of Health and Human Services, was created in 1984.

These institutional developments helped RAND to enhance its work in existing areas of the research program — for example, health policy — and to stimulate work in new areas — for example, analysis of Army policy issues.

During this period, RAND's research program increased substantially in size and diversity. Many of the trends of the past continued — for example, the increase in efforts devoted to domestic policy research and the tendency to conduct research in broader contexts. Several new trends began to emerge — for example, an increase in emphasis on research done in international contexts other than the (then) USSR. The development of analytical concepts, methods, and techniques also continued. Some of the more important of these were:

- *RAND Strategy Assessment System* (*RSAS*) — Because of perceived limitations in methods of strategic analysis, in 1982 RAND began to develop methods for strategic analysis that combined classical gaming, systems analysis methods and techniques, artificial intelligence, and advanced computer technology. The RSAS provided a structure and tools for analyzing strategic decisions at the national command level as well as decisions at the operational level. It also

provided great flexibility in choosing which roles are to be played by people and which by machines.

- *Dyna-METRIC* — The Dyna-METRIC logistics support model provided a major new tool for relating the availability of spare parts to wartime aircraft sortie generation capability. The model, which was developed by Richard Hillestad, Manuel Carrillo, and Gordon Crawford, combines elements of queueing theory, inventory theory, and simulation. It is still an integral part of the Air Force logistics and readiness management system.
- *CLOUT* (*Coupling Logistics to Operations to Meet Uncertainties and the Threat*) is a RAND-developed set of initiatives for improving the ability of the Air Force logistics system to cope with the uncertainties and disruptions of a conventional war overseas. The CLOUT initiatives are intended to offset the substantial variability expected in the demand for spare parts, maintenance, and other support activities, as well as the consequences of damage to theater air bases. CLOUT was the basis for the DRIVE model (Distribution and Repair in Variable Environments), developed by Lou Miller and John Abell, that became the kernel of a management system, called EXPRESS, that is still used today in Air Force depots.
- *The Enlisted Force Management System* (*EFMS*)—The EFMS project is notable for the scope and complexity of the decision support system that it developed, and for demonstrating how the tools of operations research could be married with emerging information technologies to provide real-time decision support throughout an organization. Warren Walker led a large RAND team that worked with Air Force counterparts beginning in the early 1980s. Together, they produced an organizational decision support system (ODSS) to help make decisions about the grade structure of the enlisted force, enlisted promotion policies, and the recruitment, assignment, training, compensation, separation, and retirement of Air Force enlisted personnel. Since 1990, the EFMS has been the primary analytical tool used to support major policy decisions affecting the enlisted force. The success of the system motivated the publication of a 1992 book, *Building Organizational Decision Support Systems*.

Concluding Remarks

After the first 40 years, most of the main trends outlined above continue to play themselves out — for example, domestic research represents about half of RAND's $250 million annual research budget, and methodological enhancements driven by the practical needs of the problem-oriented research continue to have high priority. Methodologically, some of the most important advances have involved new approaches for dealing with uncertainty in making decisions. Chief among these approaches are Assumption-Based Planning (Dewar et al. 1993), exploratory modeling (Bankes 1993), and adaptive policies (*RAND* (Europe 1997)). All three of these approaches are combined in a methodology for long-term policy analysis called Robust Decisionmaking (Lempert et al. 2003).

Over the past two decades, RAND has been becoming a more global institution. The RAND research staff now includes citizens of more than 50 nations and RAND now performs research for many countries besides the United States. In 1992, RAND established an affiliate in the Netherlands called RAND Europe. It conducts policy studies to inform public- and private-sector decision making throughout Europe. Major research efforts so far have included studies of the safety of Schiphol Airport, ways of improving river dikes in the Netherlands while preserving the environment, a systematic examination of alternative strategies for reducing the negative effects of road freight transport in the Netherlands, and a cost-effectiveness analysis of strategies for improving shipping safety in the North Sea. Other efforts include pioneering work on widening the application of discrete choice analysis, guiding appraisals of transport infrastructures, and providing cost analysis for the UK Ministry of Defence. RAND Europe now operates from its headquarters in Cambridge, England and a representation office in Brussels. With a diverse range of research areas ranging from defense to healthcare, it has become a key provider of evidence-based policy research across the European Union.

In 2003, RAND established the RAND-Qatar Policy Institute (RQPI). The RQPI is located in Doha, Qatar, and is part of Education City. RQPI serves clients throughout the Middle East, North Africa, and

South Asia, performing studies on such issues as education reform, health services delivery, governmental reorganization, and environmental health. RQPI also helps train analysts and build capacity in Qatar and other countries of the region.

RAND analysts continue to be recognized for their contributions to the profession of operations research and the management sciences. In 1996, Moore et al. were awarded the Richard H. Barchi Prize for the best paper given at the Military Operations Research Society (MORS) Symposium. Brooks et al. won that prize in 1998. Warren Walker received the INFORMS President's Award in 1997 for his "contributions to the welfare of society through quantitative analysis of governmental policy problems."

See

- Artificial Intelligence
- Cost Analysis
- Deep Uncertainty
- Delphi Method
- Dynamic Programming
- Emergency Services
- Exploratory Modeling and Analysis
- Fire Safety Modeling and Applications
- Game Theory
- Gaming
- Inventory Modeling
- Linear Programming
- Logistics and Supply Chain Management
- Military Operations Research
- Network Planning
- Network Optimization
- Nonlinear Programming
- Optimization
- Public Policy Analysis
- Queueing Theory
- Simulation of Stochastic Discrete-Event Systems
- Systems Analysis
- Traveling Salesman Problem

References

Bankes, S. C. (1993). Exploratory modeling for policy analysis. *Operations Research, 4*(3), 435–449.

Bellman, E. R. (1957). *Dynamic programming*. Princeton, NJ: Princeton University Press.

Bellman, E. R., & Dreyfus, S. E. (1962). *Applied dynamic programming*. Princeton, NJ: Princeton University Press.

Brooks, A., Bennett, B., & Bankes, S. (1998). An application of exploratory analysis: The weapon mix problem. *MORS Journal, 4*(1), 67–80.

Carter, G. M., Murray, M. P., Walker, R. G., & Walker, W. E. (1992). *Building organizational decision support systems*. San Diego, CA: Academic Press.

Dalkey, N. C., Rourke, D. L., Lewis, R. J., & Snyder, D. (1972). *Studies in the quality of life: Delphi and decision-making*. Lexington, MA: D.C. Heath.

Dantzig, G. (1963). *Linear programming and extensions*. Princeton, NJ: Princeton University Press.

Dewar, J. A., Builder, C. H., Hix, W. M., & Levin, M. H. (1993). *Assumption-based planning: A planning tool for very uncertain times, MR-114-A*. Santa Monica, CA: RAND.

Europe, R. A. N. D. (1997). *Adaptive policies, policy analysis, and civil aviation policymaking, DRU-1514-VW/VROM/EZ*. Santa Monica, CA: RAND.

Fisher, G. H. (1971). *Cost considerations in systems analysis*. New York: American Elsevier Publishing Co.

Ford, L. R., Jr., & Fulkerson, D. R. (1962). *Flows in networks*. Princeton, NJ: Princeton University Press.

Goeller, B. F., et al. (1983). *Policy analysis of water management for the Netherlands* (Vol. 1), Summary Report, R-2500/1–NETH, RAND, Santa Monica, CA.

Harris, T. E. (1964). *The theory of branching processes*. Englewood Cliffs, NJ: Prentice-Hall.

Hitch, C. J., & McKean, R. (1960). *The economics of defense in the nuclear age*. Cambridge, MA: Harvard University Press.

Jardini, D. R. (1996). *Out of the blue yonder: The RAND corporation's diversification into social welfare research, 1946–1968*. Doctoral dissertation, College of Humanities and Social Sciences, Carnegie Mellon University, Pittsburgh, PA.

Kahn, H. (1960). *On thermonuclear war*. Princeton, NJ: Princeton University Press.

Lempert, R. J., Popper, S. W., & Bankes, S. C. (2003). *Shaping the next one hundred years: New methods for quantitative, long-term policy analysis, MR-1626-RPC*. Santa Monica, CA: RAND.

Markowitz, H. M., Hausner, B., & Karr, H. W. (1963). *SIMSCRIPT: A simulation programming language*. Englewood Cliffs, NJ: Prentice-Hall.

Moore, S. C., Kakalik, J. S., Benjamin, D. R., & Stanton, R. E. (1996). *Choosing force structures: Modeling interactions among wartime requirements, peacetime basing options, and manpower and personnel policies, MR-550-AF*. Santa Monica, CA: RAND.

Newell, A. (Ed.). (1961). *Information processing language-V manual*. Englewood Cliffs, NJ: Prentice-Hall.

Novick, D. (Ed.). (1965). *Program budgeting*. Cambridge, MA: Harvard University Press.

Paxson, E. (1950). *Strategic bombing systems analysis*. Santa Monica, CA: RAND.

Quade, E. S. (Ed.). (1964). *Analysis for military decisions*. Chicago: Rand McNally.

RAND. (1955). *A million random digits with 100,000 normal deviates, MR-1418-RC*. Santa Monica, CA: RAND (Reissued in 2001 with a new Foreword by Michael Rich.).

Shaw, J. C. (1964). *JOSS: A designer's view of an experimental on-line computing system, P-2922*. Santa Monica, CA: RAND.
Sherbrooke, C. C. (1966). *METRIC: A multi-echelon technique for recoverable item control, RM-5078-PR*. Santa Monica, CA: RAND.
Walker, W. E., Chaiken, J. M., & Ignall, E. J. (Eds.). (1979). *Fire department deployment analysis: A public policy analysis case study*. New York: Elsevier North Holland.
Williams, J. D. (1954). *The compleat strategyst: Being a primer on the theory of games of strategy*. New York: McGraw-Hill.
Wohlstetter, A. J., Hoffman, F. S., Lutz, R. J., & Rowen, H. S. (1954). *Selection and use of strategic air bases, R-266*. Santa Monica, CA: RAND.

Random Field

A stochastic process with a multi-dimensional index set; for example, $\{R(x, y), -\infty < x, y < \infty\}$, where $R(x, y)$ equals the amount of rain falling during a given day at location (x, y).

Random Number Generators

Pierre L'Ecuyer
Université de Montréal, Montréal, Québec, Canada

Introduction

Many algorithms and heuristics in operations research and management science require a source of random numbers. This is needed in particular to simulate stochastic models, to estimate multivariate integrals and the solutions of differential equations numerically by Monte Carlo, and in probabilistic algorithms in general. These so-called random numbers are typically produced by a deterministic computer program, and are therefore not random at all. The program that produces them is nevertheless called a random number generator (RNG). Its aim is to imitate the realization of a sequence of i.i.d. (independent and identically distributed) random variables, say from the $\mathcal{U}(0,1)$ distribution (the uniform distribution over the interval between 0 and 1). The generator and the numbers it returns are sometimes called pseudorandom to emphasize their deterministic nature, but the common usage of just random will be adopted here for simplicity.

True random numbers can be produced by physical devices such as noise amplifiers in electric resistances, photon trajectory detectors, and the like. Typically, these devices construct a binary sequence by sampling a signal periodically and returning 1 if the signal is above a given threshold, and 0 otherwise. Simple transformations are applied to this sequence, for example by combining bits, to remove (or reduce) the bias and the dependence between the bits. Although one cannot prove that the returned bits are 1 or 0 with probabilities exactly 1/2 and that they are all independent, the output binary sequences of some of these devices are so close to having these properties that the difference is practically undetectable. Such physical RNGs are appropriate for applications such as cryptography and gaming machines, for example, where unpredictability is essential. But for stochastic simulation, well-crafted algorithmic RNGs, based on a purely deterministic mathematical recurrence, have a good enough statistical behavior and are sufficiently reliable. They are also much more convenient because they require no specialized hardware and their output sequences can be repeated as many times as desired without the need for storage. The ability to repeat the same sequence exactly is a key requirement for the implementation of efficient stochastic simulation techniques (Asmussen and Glynn 2007; L'Ecuyer 2008; Law and Kelton 2000).

There is a well-developed body of theory on the construction and analysis of algorithmic RNGs (Knuth 1998; L'Ecuyer 1994, 2004, 2006; L'Ecuyer and Panneton 2009; Niederreiter 1992; Tezuka 1995). Obvious requirements such as a fast implementation and a very long period are not sufficient. Good and reliable RNGs cannot be constructed at random by just trying arbitrary recurrences and testing them empirically. They must be constructed on the basis of a good understanding of the uniformity of their vectors of output values, by proper analysis of their mathematical structure. Empirical statistical testing is certainly relevant, but it should comes only after a solid theoretical analysis.

Unfortunately, poor and unreliable RNGs still abound in the scientific literature and in popular software (L'Ecuyer and Simard 2007). Typically, their weakness comes from too much structure, due to overly simplified recurrences constructed to

optimize speed. For most applications, it turns out that these weaknesses have little or no noticeable impact on the simulation results, because the random numbers are sufficiently transformed by the application to distort the simple structure and this has the accidental effect of improving the RNG for that particular case. But in some situations, when the RNG structure is not distorted by the application, things can quickly become disastrous in the sense that simulation results are totally wrong, sometimes by huge factors. Examples of specific well-tested and recommended generators, with computer codes, can be found in L'Ecuyer (1999a, b), L'Ecuyer and Touzin (2000), L'Ecuyer et al. (2002), Panneton et al. (2006), for example.

Random variates from non-uniform distributions, such as the normal, exponential, Poisson, and so on, as well as random vectors from multivariate distributions, stochastic processes, and other types of random objects, are generated by transforming uniform random numbers in an appropriate way (Devroye 1986, 2006; Gentle 2003; Hörmann et al. 2004; L'Ecuyer 2004).

Algorithmic Generators

An algorithmic RNG is defined in L'Ecuyer (1994) as a structure $(\mathcal{S}, \mu, f, \mathcal{U}, g)$, where $\mathcal{S}$ is a finite set of states, $s_0 \in \mathcal{S}$ is the seed or initial state selected with the probability distribution μ on $\mathcal{S}$, $f : \mathcal{S} \to \mathcal{S}$ is the transition function, $\mathcal{U}$ is a set of output values taken here as the interval $\mathcal{U} = (0,1)$, and $g : \mathcal{S} \to \mathcal{U}$ is the output function. The generator starts in state s_0, evolves according to $s_i = f(s_{i-1})$, and outputs $u_i = g(s_i)$ at step i. The output sequence u_0, u_1, u_2, ... should behave as the realization of a sequence of i.i.d. $\mathcal{U}(0,1)$ random variables, Since $\mathcal{S}$ is finite, the sequence of states is ultimately periodic. The period is the smallest positive integer ρ such that for some integer $\tau \geq 0$ and for all $n \geq \tau, s_{\rho+i} = s_i$. Usually, $\tau = 0$, in which case the sequence is said to be purely periodic. The role of μ is to allow a random selection of the seed s_0, using an external source of randomness. Generating a truly random seed is much less work and is more reasonable than generating a long sequence of truly random numbers. An RNG with a random seed can be viewed as an extensor of randomness, whose purpose is to save "coin tosses". It stretches a short truly random seed into a long sequence of values that is supposed to behave like a true random sequence.

Basic requirements for a good RNG include a very long period (say, at least 2^{100} and preferably 2^{200} or more), the availability of a fast platform-independent implementation, the possibility of splitting the sequence into long disjoints subsequences (more on this later), and efficient ways of jumping between those sequences and to replay the same sequences over and over again. To show that a long period is not sufficient, consider an RNG defined by $s_0 = 1$, $s_i = (s_{i-1} + 1) \bmod 2^{1000}$, and $u_i = s_i/2^{1000}$. It has a huge period and a fast implementation is trivial to obtain, but it is definitely not an acceptable RNG; the successive output values are very strongly correlated.

Multivariate Uniformity

Uniformity and independence of successive output values can be assessed by studying the uniformity of the sets $\Psi_s = \{(u_0, \ldots, u_{s-1}) = (g(s_0), \ldots, g(f^{s-1}(s_0)) : s_0 \in \mathcal{S}\}$ of all vectors of s successive output values produced the the RNG, from all possible initial seeds s_0, for arbitrary positive integers s. The theoretical ideal that the u_i are i.i.d. $\mathcal{U}(0,1)$ is equivalent to having $(u_i, \ldots, u_{i+s-1})$ uniformly distributed over $(0,1)^s$ for each $s \geq 1$. One can argue that the RNG provides a good approximation of this if Ψ_s covers the unit hypercube $(0,1)^s$ very evenly (or uniformly). The rationale is that the (large) finite set Ψ_s can be viewed as a sample space from which a tiny fraction of the points are taken at random by the generator, without replacement, and this provides a good approximation of the uniform distribution over $(0,1)^s$ only if Ψ_s covers the space very uniformly. This argument also suggests that RNGs should have huge periods, many orders of magnitude larger than whatever will be used in practice, so that only a tiny fraction of the points from Ψ_s are used.

A key issue is how to measure this uniformity. This must be done from a mathematical analysis of the recurrence, without enumerating the points explicitly (which would be infeasible). Measures of uniformity are usually defined in a way that they can be computed efficiently by exploiting the linear structure of the recurrence. This is the main reason why RNGs based on linear recurrences are still the most widely used.

Good RNGs are typically constructed by first selecting a form of recurrence that can be implemented efficiently, choosing a period length (or the cardinality of $\mathcal{S}$), and then searching for parameters of the recurrence for which the target period length is reached and for which Ψ_s has the best possible uniformity, for all s up to a given value. This type of search can take months of computing time (L'Ecuyer 1999a, b; Matsumoto and Nishimura 1998; Panneton et al. 2006).

Multiple Streams and Substreams

Modern stochastic simulation software tools contain uniform RNGs with multiple streams and substreams of random numbers (L'Ecuyer 2008; L'Ecuyer et al. 2002). Each stream provides a very long (practically inexhaustible) sequence of numbers, divided into a large number of long disjoint substreams, and methods (or procedure calls) are readily available to jump ahead to the beginning of the next substream, or jump back to the beginning of the current substream or the beginning of the first stream. These streams and substreams can be taken as independent sequences. Streams can be created at will, in practically unlimited numbers, just like instances of other types of objects in a computer program. For their implementation, one needs efficient jump-ahead algorithms, which permit one to quickly compute the state s_{i+v} for an arbitrarily large v, given the current state s_i (L'Ecuyer 1990; L'Ecuyer et al. 2002).

These streams are certainly convenient when simulating a system on multiple processors; each processor can be given its own stream(s) and run it without having to communicate with other processors or with a central monitor to get its random numbers.

An important situation where multiple streams and substreams are useful even on a single processor is when one wishes to simulate two (or more) similar systems with common random numbers, to reduce the variance when comparing their performance (Asmussen and Glynn 2007; Law and Kelton 2000). For example, suppose one wishes to estimate the sensitivity (or derivative) of some performance measure of a complicated queueing system with respect to a small change in some parameter of the system. The model would be simulated with and without the change, with the same random numbers used for the same purpose (as much as possible) in the two configurations, and this would be repeated several times, independently. To this end, one would reserve a separate random stream for each type of random numbers required in the model (e.g., each type of arrival, each type of service time, each type of routing decision, etc.), to make sure that the same random numbers play the same role even though they are generated in a different order and their required quantity differs across the two simulation runs of any given pair. To ensure that the same random numbers from each stream are reused within any pair of runs, one would advance all streams to a new substream before each pair of runs, simulate the first configuration, rewind the streams back to the beginning of the current substream, simulate the second configuration, then move them ahead to the beginning of the next substream, for the next runs. This example also illustrates an important advantage of algorithmic RNGs compared with those based on physical devices: replaying the same sequence several times, which is frequently needed, is much easier.

Empirical Statistical Tests

Theoretical analysis of RNGs by computing their period and measuring the uniformity of Ψ_s must be complemented by empirical statistical tests. These empirical tests can never prove that an RNG has no defect, no matter how many tests are applied, but at least they can be reassuring. There is no limit on the number of tests that can be defined. The *TestU01* library (L'Ecuyer and Simard 2007) describes and implements a large collection of statistical tests for RNGs, as well as predefined batteries of tests of different strengths. It includes most of the tests previously proposed in Knuth (1998) and Marsaglia (1996), for example, and many more. Ideally, the tests should be selected in relation with the target application. So, when a general-purpose generator provided in a software library is used for a sensitive application, it may be wise to submit it to additional specialized empirical testing. For any RNG whose output sequence is periodic, one can always construct (in principle) statistical tests that this RNG will fail unequivocally (L'Ecuyer 2006). In some sense,

the good RNGs are those for which those tests are more complicated, hard to construct, or take too much time to run.

Linear Recurring Sequences and MRGs

The most popular currently-used RNGs are based on linear recurrences of the form

$$x_i = (a_1 x_{i-1} + \cdots + a_k x_{i-k}) \mod m, \qquad (1)$$

where m is a positive integer called the modulus, $a_1, \ldots, a_k$ are integers between $-m$ and m called the multipliers, with $a_k \neq 0$, and k is the order of the recurrence. Define the state of the recurrence at step n as $s_i = \mathbf{x}_i = (x_i, \ldots, x_{i+k-1}) \in \mathbb{Z}_m^k$. The zero state (0,... ,0) is absorbing and must be avoided. The longest possible period is $m^k - 1$, and it is achieved if and only if m is a prime number and the characteristic polynomial of the recurrence,

$$P(z) = z^k - a_1 z^{k-1} - \cdots - a_k,$$

is a primitive polynomial modulo m. Such primitive polynomials are easy to find once the factorizations of $m - 1$ and $(m^k - 1)/(m - 1)$ and available (Knuth 1998; L'Ecuyer 1990, 1999a). For a full-period recurrence, in any subsequence of ρ consecutive values of the state $_i$, each element of $\{0, \ldots, m-1\}^k$ appears exactly once, except for $\mathbf{x}_i = (0,\ldots,0)$. A multiple recursive generator (MRG), in its classical definition, uses the recurrence (1) with a large m, and returns $u_i = x_i/m$ as its output at step i (L'Ecuyer 1994; Niederreiter 1992). In implementations, the output transformation is modified slightly so that the MRG never returns 0 or 1, for example by taking $u_i = (x_i + 1)/(m + 1)$. When $k = 1$, it is a linear congruential generator (LCG). LCGs with modulus $m \leq 2^{64}$ have been popular in the past, but they should no longer be used except for very short simulations, because their state space (and period) is too small.

The MRG recurrence can be written in matrix form as $\mathbf{x}_i = \mathbf{A}\mathbf{x}_{i-1} \mod m$ for some matrix $\mathbf{A}$, where $\mathbf{x}_i = (x_{i-k+1}, \ldots, x_i)^t$. This form can be exploited to jump ahead by an arbitrary number of steps in a single leap, via the matrix-vector multiplication $\mathbf{x}_{i+v} = (\mathbf{A}^v \mod m)\mathbf{x}_i \mod m$, after having precomputed $\mathbf{A}^v \bmod m$ (L'Ecuyer 1990, 2006).

An important characteristic of MRGs is that the corresponding sets Ψ_s have a lattice structure: Ψ_s is the intersection of a lattice with the unit hypercube $[0, 1)^s$. This implies that all s-dimensional vectors $\mathbf{u}_i = (u_i, \ldots, u_{i+s-1})$, for $i \geq 0$, lie in a relatively small number of equidistant parallel hyperplanes (Knuth 1998). The shorter the distance d_s between those hyperplanes, the better, because this means thinner empty slices of space, and so a more evenly distributed set Ψ_s. This d_s can be computed in reasonable time in up to about 50 dimensions. A standardized figure of merit between 0 and 1 can then be computed by dividing a lower bound on the best possible value of d_s (which depends on s and m^k) by the actual value of d_s, and taking the minimum over a given range of values of s (L'Ecuyer 1996a, 1999a). For good MRGs, the resulting figure should be above 0.6, say.

It is tempting to take m as a power of two in (1), because then the mod m operation can be performed trivially by just chopping-off the high-order bits, without caring for overflow. However, there is a high price to pay in terms of period length and statistical robustness. When $m = 2^e$, the period of (1) cannot exceed $(2^k - 1)2^{e-1}$ if $k > 1$, and 2^{e-2} if $k = 1$ and $e \geq 4$. In the latter case, the period of the j-th least significant bit is at most 2^{j-2} (so the low-order bits have very short periods).

To obtain recurrences with a large period and a fast implementation, it also appears attractive to take a large k, many coefficients a_j equal to 0, and the other ones small. One extreme case is to have only two nonzero coefficients a_j, say a_r and a_k, both equal to ± 1; this is an additive/subtractive lagged-Fibonacci RNG. But in this case, all the vectors (u_i, u_{i-r}, u_{i-k}) produced by the RNG lie in only two parallel planes in $[0, 1)^3$ (L'Ecuyer 1997). This is quite bad. In general, the distance d_{k+1} between the successive hyperplanes in Ψ_{k+1} always satisfies $1/d_{k+1}^2 \leq 1 + a_1^2 + \cdots + a_k^2$ (L'Ecuyer 1997). Thus, there is no chance of having a good lattice structure if the latter sum of squares is small. This means that taking only small coefficients a_j, and many of them equal to 0, is a bad idea. L'Ecuyer and Touzin (2004) point out a similar problem when all nonzero a_j's are equal to the same constant a, as suggested by Deng and Lin (2000), Deng and Xu (2003), for example. A class of generators named

add-with-carry and subtract-with-borrow are slight modifications of MRGs having essentially the same bad properties as the lagged-Fibonacci.

One way to construct reliable and efficient MRGs is combine two or more MRGs for which a fast implementation is available (their coefficients can be small). If they are combined by adding their outputs modulo 1, then the combination turns out to be another MRG, and one should select the components so that the resulting combined MRG has a long period and excellent multivariate uniformity. Good parameters for such constructions have been found by extensive computerized searches and concrete implementations are available in L'Ecuyer (1996a, 1999a) and L'Ecuyer and Touzin (2000), for example.

Linear Recurrences Modulo 2

A second popular and efficient class of RNGs are based on linear recurrences of the form (1) with modulus $m = 2$ and a large k. They are often called $\mathbb{F}_2$ -linear, because this is a linear recurrence in the finite field $\mathbb{F}_2$ with elements $\{0, 1\}$. Following L'Ecuyer (2006) and L'Ecuyer and Panneton (2009), their general form can be given in matrix notation by

$$\begin{aligned} \mathbf{x}_i &= \mathbf{A}\mathbf{x}_{i-1} \bmod 2, \\ \mathbf{y}_i &= \mathbf{B}\mathbf{x}_{i-1} \bmod 2, \\ u_i &= \sum_{\ell=1}^{w} y_{i,\ell-1}\, 2^{-\ell} \end{aligned}$$

where $\mathbf{x}_i = \left(x_{i,0}, \ldots, x_{i,k-1}\right)^{\mathrm{t}}$ is the k-bit state vector at step i, $\mathbf{y}_i = \left(y_{i,0}, \ldots, y_{i,w-1}\right)^{\mathrm{t}}$ is a w-bit output vector, k and w are positive integers, $\mathbf{A}$ is a $k \times k$ binary matrix, $\mathbf{B}$ is a $w \times k$ binary matrix, and $u_i \in [0, 1)$ is the output at step i. This output is usually modified slightly so that u_i is never exactly 0 or 1.

This class includes the Tausworthe or linear feedback shift register (LFSR) generators, the generalized feedback shift register (GFSR), the twisted GFSR, the Mersenne twister, the WELL generator, and combinations of these, among others (L'Ecuyer 1999b, 2004; L'Ecuyer and Panneton 2009; L'Ecuyer and Simard 2007; Matsumoto and Nishimura 1998; Panneton et al. 2006; Tezuka 1995). The maximal period is $2^k - 1$, reached when the characteristic polynomial $P(z)$ of $\mathbf{A}$ is a primitive polynomial modulo 2. Each coordinate of $\mathbf{x}_i$ obeys the recurrence (1) based on this characteristic polynomial. The matrices $\mathbf{A}$ and $\mathbf{B}$ are selected to allow a fast implementation by using just a few simple binary operations such as or, exclusive-or, shift, and rotation, on blocks of bits, while still providing good uniformity of Ψ_s. This uniformity is assessed differently than for MRGs, by measures of equidistribution of the points in rectangular boxes obtained by partitioning the interval [0, 1) for each axis j into subintervals of lengths 2^{-q_j} for some integers $q_j \geq 0$ (L'Ecuyer 1996b; L'Ecuyer and Panneton 2009). Combined generators of this type, obtained by a bitwise exclusive-or of the output vectors Y_i of two or more simple $\mathbb{F}_2$-linear generators, are equivalent to yet another $\mathbb{F}_2$-linear generator (L'Ecuyer 1996b, 1999b; L'Ecuyer and Panneton 2009; Tezuka 1995). They provide fast and good RNGs.

Nonlinear Generators

Many believe that the structure of linear sequences is too regular and that the right way to go is nonlinear (Eichenauer-Herrmann 1995; Niederreiter 1992). One can introduce nonlinearity by either (a) using a linear-type generator but transforming the state nonlinearly to produce the output, or (b) constructing a generator with a nonlinear transition function f. One example of (a) is the inversive generator, which uses (1) but then takes the inverse of x_i modulo m (discarding the zeros) before dividing by m to produce the output (Eichenauer-Herrmann 1995).One example of (b) is the BBS generator, proposed by Blum et al. (1986) for cryptographic applications, which evolves according to a recurrence of the form $x_i = x_{i-1}^2 \bmod m$, and just a few of the least significant bits of x_i are retained. These RNGs have sets Ψ_s with less regularity than the linear ones, but they are slower and less convenient for simulation applications. A nonlinear RNG can also be obtained by combining two linear generators of different types. L'Ecuyer and Granger-Piché (2003) combine an MRG with an $\mathbb{F}_2$-linear generator, and obtain bounds on certain measures of uniformity of Ψ_s for the combination.

Non-uniform Random Variates

The standard approach for generating random variables from non-uniform distributions is to apply further transformations to the output values u_i of a uniform RNG. This is easily done for several distributions, but not all. Devroye (1986), Gentle (2003), and (Hörmann et al. 2004) provide extensive coverages of the most useful methods. In some cases, compromises must be made between simplicity of the algorithm, quality of the approximation, robustness (with respect to parameter changes), and efficiency (speed and memory requirements). As a general rule, simplicity should not be sacrificed for small speed gains.

Inversion. Conceptually, the simplest way to generate a one-dimensional random variable X with cumulative distribution function (cdf) F is by inversion: put $X = F^{-1}(U) \stackrel{\text{def}}{=} \min\{x \mid F(x) \geq U\}$, where U is $\mathcal{U}(0,1)$. With that definition of X, one has $\mathbb{P}[X \leq x] = \mathbb{P}[F^{-1}(U) \leq x] = \mathbb{P}[U \leq F(x)] = F(x)$, and so X has cdf F. This method requires that F^{-1} (or a good approximation of it) be available. In most simulation applications, inversion should be the method of choice, because it is the only monotone non-decreasing transformation of U into X, which makes it most compatible with major variance reductions techniques such as antithetic variates, common random numbers, external control variates, and randomized quasi-Monte Carlo (Asmussen and Glynn 2007; Law and Kelton 2000; L'Ecuyer 2009). However, in situations where speed is the real issue and where monotonicity is not critical, non-inversion methods might be appropriate.

For a specific example of inversion, if X has the Weibull distribution with parameters α and β, then $F(x) = 1 - \exp[-(x/\beta)^{\alpha}]$ for $x>0$, and $F^{-1}(U) = \beta[-\ln(1-U)]^{1/\alpha}$, so X is easy to generate. As another example, if X is geometric with parameter p, then $F(x) = 1 - (1-p)^{x+1}$ for $x = 0, 1, 2, \ldots$ and $F^{-1}(U) = \lfloor \ln(1-U)/\ln(1-p) \rfloor$.

For some distributions, F^{-1} cannot be written in closed form but it can be approximated numerically. For distributions having only scale and location parameters, it suffices to approximate the cdf of a standardized version (say with scale 1 and location 0) and then rescale and shift the result appropriately. For example, several good approximations of the inverse standard normal cdf Φ^{-1} are available, and some provide essentially machine-precision accuracy while being fast to compute (L'Ecuyer 2008).

For distributions with shape parameters, a different approximation of F^{-1} must be constructed for each value of those parameters. If a large number of variates are to be generated from the same F (with the same shape parameters), it can be worthwhile to construct on-demand an approximation of F^{-1} based on interpolation methods, by a setup algorithm that precomputes tables from which $F^{-1}(u)$ can be evaluated quickly for any u (Derflinger et al. 2010; Hörmann et al. 2004).

Tables can be precomputed in a similar way for discrete distributions such as the Poisson and binomial, for example, with fixed parameters. When the support is large or infinite (such as for the Poisson distribution), one would only store a truncated table, over a finite range with probability close to 1, and compute the remaining values only when needed (which would typically be rare). The fastest methods are obtained by using an index, as follows (Devroye 1986; Hörmann et al. 2004). One partitions the interval $(0, 1)$ into c subintervals of equal sizes, $[j/c, (j+1)/c)$ for $j = 0, \ldots, c-1$, and store the smallest and largest possible values of X for each subinterval, namely $L_j = F^{-1}(j/c)$ and $R_j = F^{-1}((j+1)/c)$. Once U is generated, the corresponding interval number $J = \lfloor cU \rfloor$ is readily computed, and the index I of the returned value is searched only in the interval $[L_J, R_J]$, with linear or binary search. By taking a large enough c, this gives a method that works in constant time for practical purposes.

Rejection methods. To generate X from a complicated density f, one can find a simpler dominating function t (called a "hat" function) for which $f(x) \leq t(x)$ for all x and such that generating variates from the density r defined by $r(x) = t(x)/a$, where $a = \int_{-\infty}^{\infty} t(s)ds$, is easy. This r is just the rescaling of t into a density. The random variate X can be generated by repeating: generate Y from the density r and an independent $\mathcal{U}(0,1)$ variate U, until $U \leq f(Y)/t(Y)$; then return $X = Y$. This is the rejection method (Hörmann et al. 2004; Devroye 1986). The number of returns into the repeat loop is a geometric random variable with mean $a > 1$, which can be reduced by having a closer to 1 (or t closer to f). A compromise must be made between reducing a and

keeping t simple. The rejection method is often combined with a change of variable to transform the density f into a new density for which a more effective hat function and a fast implementation can be constructed.

In general, rejection can also be used to generate a point uniformly in a complicated region R, in an arbitrary space (not necessarily the real space). The idea is to define another region B that contains R, and in which it is easy to generate points uniformly. Random points are generated uniformly in B and the first one that falls in R is retained.

A variant of the rejection method, called thinning, is sometimes adopted to generate events from a non-homogeneous Poisson process. If the process has rate $\lambda(t) \leq \bar{\lambda}$ for all t, where $\bar{\lambda}$ is a finite constant, one can generate Poisson pseudo-arrivals at constant rate $\bar{\lambda}$ by generating interarrival times as i.i.d. exponentials of mean $1/\bar{\lambda}$. Then, any pseudo-arrival at time t is accepted (becomes an arrival) with probability $\lambda(t)/\bar{\lambda}$ (i.e., if $U \leq \lambda(t)/\bar{\lambda}$, where U is an independent $\mathcal{U}(0,1)$, and rejected with probability $1 - \lambda(t)/\bar{\lambda}$.

Multivariate distributions. The cdf of a random vector $\mathbf{X} = (X_1, \ldots, X_d)$ does not have a well-defined inverse, so inversion does not apply directly to generate $\mathbf{X}$. In some situations, one can generate X_1 first, then X_2 conditional on X_1, then X_3 conditional on (X_1, X_2), and so on. In other situations, one can generate a vector of d independent random variates by inversion and transform it to obtain $\mathbf{X}$. For example, if the target distribution for $\mathbf{X}$ is multinormal with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, then one can first decompose $\boldsymbol{\Sigma} = \mathbf{A}\mathbf{A}^T$ (e.g., via the Cholesky decomposition or via an eigendecomposition), where the superscript "T" denotes matrix transpose, and return $\mathbf{X} = \boldsymbol{\mu} + \mathbf{A}\mathbf{Z}$, where $\mathbf{Z}$ is a vector of d i.i.d. standard normals, which can in turn be generated by inversion. A very general (and rich) way of specifying a multivariate distribution is via a copula, which is basically a multivariate distribution whose one-dimensional marginals are all $\mathcal{U}(0,1)$. If $\mathbf{U} = (U_1, \ldots, U_d)$ is generated from the copula, and $X_j = F_j^{-1}(U_j)$ for each j, then $\mathbf{X}$ is a random vector whose jth marginal cdf is F_j and whose dependence structure is determined by the copula. See Hörmann et al. (2004) and Nelsen (1999) for more details.

See

- Hit-and-Run Methods
- Markov Chain Monte Carlo
- Monte Carlo Methods
- Monte Carlo Simulation
- Randomized Algorithm
- Simulation of Stochastic Discrete-Event Systems
- Variance Reduction Techniques in Monte Carlo Methods

References

Asmussen, S., & Glynn, P. W. (2007). *Stochastic simulation*. New York: Springer.

Blum, L., Blum, M., & Schub, M. (1986). A simple unpredictable pseudo-random number generator. *SIAM Journal on Computing, 15*(2), 364–383.

Deng, L.-Y., & Lin, D. K. J. (2000). Random number generation for the new century. *The American Statistician, 54*(2), 145–150.

Deng, L.-Y., & Xu, H. (2003). A system of highdimensional, efficient, long-cycle and portable uniform random number generators. *ACM Transactions on Modeling and Computer Simulation, 13*(4), 299–309.

Derflinger, G., Hörmann, W., & Leydold, J. (2010). Random variate generation by numerical inversion when only the density is known. *ACM Transactions on Modeling and Computer Simulation*, *20*(4). Article 18.

Devroye, L. (1986). *Non-uniform random variate generation*. New York: Springer.

Devroye, L. (2006). Nonuniform random variate generation. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science* (pp. 83–121). Amsterdam, The Netherlands: Elsevier. Chapter 4.

Eichenauer-Herrmann, J. (1995). Pseudorandom number generation by nonlinear methods. *International Statistical Reviews, 63*, 247–255.

Gentle, J. E. (2003). *Random number generation and Monte Carlo methods* (2nd ed.). New York: Springer.

Hörmann, W., Leydold, J., & Derflinger, G. (2004). *Automatic nonuniform random variate generation*. Berlin: Springer.

Knuth, D. E. (1998). *The art of computer programming* (Seminumerical algorithms 3rd ed., Vol. 2). Reading, MA: Addison-Wesley.

L'Ecuyer, P. (1990). Random numbers for simulation. *Communications of the ACM, 33*(10), 85–97.

L'Ecuyer, P. (1994). Uniform random number generation. *Annals of Operations Research, 53*, 77–120.

L'Ecuyer, P. (1996a). Combined multiple recursive random number generators. *Operations Research, 44*(5), 816–822.

L'Ecuyer, P. (1996b). Maximally equidistributed combined Tausworthe generators. *Mathematics of Computation, 65*(213), 203–213.

L'Ecuyer, P. (1997). Bad lattice structures for vectors of non-successive values produced by some linear recurrences. *INFORMS Journal on Computing, 9*(1), 57–60.
L'Ecuyer, P. (1999a). Good parameters and implementations for combined multiple recursive random number generators. *Operations Research, 47*(1), 159–164.
L'Ecuyer, P. (1999b). Tables of maximally equidistributed combined LFSR generators. *Mathematics of Computation, 68*(225), 261–269.
L'Ecuyer, P. (2004). Random number generation. In J. E. Gentle, W. Haerdle, & Y. Mori (Eds.), *Handbook of computational statistics* (pp. 35–70). Berlin: Springer. Chapter II.2.
L'Ecuyer, P. (2006). Uniform random number generation. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science* (pp. 55–81). Amsterdam, The Netherlands: Elsevier. Chapter 3.
L'Ecuyer, P. (2008). *SSJ: A Java library for stochastic simulation*. Software user's guide; available at author's Universit´e de Montr´eal Web site.
L'Ecuyer, P. (2009). Quasi-Monte Carlo methods with applications in finance. *Finance and Stochastics, 13*, 307–349.
L'Ecuyer, P., & Granger-Piché, J. (2003). Combined generators with components from different families. *Mathematics and Computers in Simulation, 62*, 395–404.
L'Ecuyer, P., & Panneton, F. (2009). F_2-linear random number generators. In C. Alexopoulos, D. Goldsman, & J. R. Wilson (Eds.), *Advancing the frontiers of simulation: A festschrift in honor of George Samuel Fishman* (pp. 169–193). New York: Springer.
L'Ecuyer, P., & Simard, R. (2007). August. TestU01: A C library for empirical testing of random number generators. ACM *Transactions on Mathematical Software*, *33*(4). Article 22.
L'Ecuyer, P., Simard, R., Chen, E. J., & Kelton, W. D. (2002). An object-oriented random-number package with many long streams and substreams. *Operations Research, 50*(6), 1073–1075.
L'Ecuyer, P., & Touzin, R. (2000). Fast combined multiple recursive generators with multipliers of the form $a = \pm 2^q \pm 2^r$. In J. A. Joines, R. R. Barton, K. Kang, & P. A. Fishwick (Eds.), *Proceedings of the 2000 Winter Simulation Conference* (pp. 683–689). Pistacaway, NJ: IEEE Press.
L'Ecuyer, P., & Touzin, R. (2004). On the Deng-Lin random number generators and related methods. *Statistics and Computing, 14*, 5–9.
Law, A. M., & Kelton, W. D. (2000). *Simulation modeling and analysis* (3rd ed.). New York: McGraw-Hill.
Marsaglia, G. (1996). DIEHARD: A battery of tests of randomness. Available at author's Florida State University web site.
Matsumoto, M., & Nishimura, T. (1998). Mersenne twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Transactions on Modeling and Computer Simulation, 8*(1), 3–30.
Nelsen, R. B. (1999). *An introduction to copulas* (Lecture notes in statistics, Vol. 139). New York: Springer.
Niederreiter, H. (1992). *Random number generation and quasi-Monte Carlo methods* (SIAM CBMS-NSF regional conference series in applied mathematics, Vol. 63). Philadelphia: SIAM.
Panneton, F., L'Ecuyer, P., & Matsumoto, M. (2006). Improved long-period generators based on linear recurrences modulo 2. *ACM Transactions on Mathematical Software, 32*(1), 1–16.
Tezuka, S. (1995). *Uniform random numbers: Theory and practice*. Norwell, MA: Kluwer Academic.

Random Search

A search algorithm that uses probabilistic sampling to select search points from a neighborhood of the current solution(s).

See

▶ Metaheuristics

Random Variates

Random values generated according to a specified probability distribution, corresponding to the outcomes of a random variable. Generally, this is realized on a computer through a transformation from IID pseudorandom numbers. The most commonly used procedures are the inverse transform method (inversion) and acceptance-rejection (rejection methods).

See

▶ Random Number Generators
▶ Simulation of Stochastic Discrete-Event Systems

References

Devroye, L. (1986). *Non-uniform random variate generation*. New York: Springer-Verlag.

Random Walk

If $S_n = X_1 + X_2 + \ldots + X_n$, then S_n is a special discrete-time Markov process called a random walk if $S_0 = 0$ and the random variables $\{X_i\}$ are independent and identically distributed. The most

common form of the random walk is the discrete one in which $X_i = -1$ or + 1.

See

- ▶ Markov Chains
- ▶ Markov Processes

Randomized Algorithm

An algorithm that employs a probabilistic element in its procedure, as in Monte Carlo sampling implemented using a random number generator. Thus the performance of the algorithm, in terms of results returned and computation time, will be random variables. Examples include random search, evolutionary algorithms, model-based algorithms, and algorithms based on swarm intelligence.

See

- ▶ Evolutionary Algorithms
- ▶ Metaheuristics
- ▶ Random Search
- ▶ Swarm Intelligence

References

Hromkovic, J. (2005). *Design and analysis of randomized algorithms*. New York: Springer.

Ranging

A term equivalent to a full sensitivity analysis of an optimal solution to a linear programming problem. Ranging refers to how much the cost coefficients and the right-hand-side elements of the linear program can vary before the optimal feasible basis is no longer optimal or feasible. A ranging analysis could also include variations of a technological coefficient, but it is not standard practice. A full cost and right-hand-side ranging analysis is part of computer-based simplex method solutions.

See

- ▶ Linear Programming
- ▶ Sensitivity Analysis
- ▶ Simplex Method (Algorithm)

Rank

The rank of an $m \times n$ matrix $\boldsymbol{A}$ is the maximum number of linearly independent columns in $\boldsymbol{A}$. The rank of $\boldsymbol{A}$ equals the rank of its transpose $\boldsymbol{A}^T$, with the rank not greater than m or n.

See

- ▶ Matrices and Matrix Algebra

Ranking and Selection

Statistical methods to choose the best (or a rank ordering) among a finite set of alternatives according to some probabilistic criterion, where the performance of each alternative must be estimated through statistical sampling, e.g., through simulation runs (replications).

See

- ▶ Statistical Ranking and Selection

Rare Event Simulation

Søren Asmussen[1], Paul Dupuis[2],
Reuven Y. Rubinstein[3] and Hui Wang[2]
[1]Aarhus University, Aarhus, Denmark
[2]Brown University, Providence, RI, USA
[3]Technion – Israel Institute of Technology, Haifa, Israel

Introduction

The estimation of rare event probabilities is probably one of the most challenging topics in Monte Carlo

simulation. Interest in rare events arises from many branches of science. Examples include performance analysis in communication theory and computer science where extremely small buffer overflow probabilities are of concern, chemical physics where the transition probabilities from one metastable state to another plays a key role, and risk management where measuring rare but catastrophic losses is a prerequisite. Under these circumstances, one is often interested in both qualitative and quantitative information directly related to the rare event, such as how likely is the rare event and, given that it does occur, how does it happen.

To illustrate the inefficiency of standard Monte Carlo in simulating rare events, consider a simple example. Let X be a random variable defined on some probability space $(\Omega, \mathcal{F}, \mathcal{P})$. Suppose that one is interested in estimating the probability that X is in some given set A:

$$p = P(X \in A).$$

Standard Monte Carlo would generate k independent identically distributed samples $\{X_i : i = 1, \ldots, k\}$ from the distribution of X and form an unbiased estimate

$$\hat{p}_k = \frac{1}{n} \sum_{i=1}^{k} 1_{\{X_i \in A\}},$$

where $1_{\{x \in A\}}$ is the indicator function of the set A:

$$1_{\{x \in A\}} = \begin{cases} 1 & \text{if } x \in A, \\ 0 & \text{if } x \notin A. \end{cases}$$

As the sample size k tends to infinity, the estimate $\hat{p}_k$ converges to p with probability one by the strong law of large numbers. The rate of convergence is determined by the variance of $1_{\{X \in A\}}$. More precisely, by the central limit theorem, the distribution of $\hat{p}_k$ is approximately normal with mean p and variance

$$\mathrm{Var}[\hat{p}_k] = \frac{1}{k} \mathrm{Var}\left[1_{\{X \in A\}}\right] = \frac{1}{k} p(1-p).$$

Even though this variance is very small when p is very small, the relative error associated with the estimate $\hat{p}_k$

$$\text{Relative error} = \frac{\text{standard deviation of } \hat{p}_k}{\text{mean of } \hat{p}_k} = \frac{\sqrt{p - p^2}}{\sqrt{k} p}$$

can be very large. Indeed, the relative error is unbounded as the event A becomes rarer. Therefore, a large number of samples are required in order to achieve a fixed relative error bound.

Two major classes of techniques to improve the efficiency of estimating small probabilities are importance sampling and particle splitting. When properly designed, both algorithms can dramatically reduce the number of samples needed to achieve the desired precision. Here, the focus is on importance sampling, although particle splitting is related to importance sampling in an unexpected way via subsolutions; see Glasserman et al. (1999), Dean and Dupuis (2009), Rubinstein (2010), and the references therein.

The basic idea of importance sampling is to simulate the system based on an alternative probability distribution (i.e., change of measure), and an unbiased estimate is formed by multiplying the original estimate by an appropriate likelihood ratio. Thistechnique was first applied to nuclear-physics calculation around the 1940s. Heidelberger (1995) and Asmussen and Rubinstein (1995) survey much of the research up to the early 1990s, whereas research since then is reviewed here, emphasizing basic concepts and innovative ideas. The precise statements of the theorems and their rigorous proofs can be found in the relevant references.

The next section describes two efficiency criteria for Monte Carlo simulation algorithms. Following that, two examples frequently used throughout are presented. Several different techniques for the design of the change of measure in importance sampling are then discussed, including the cross-entropy method, the game/subsolution approach in dynamic importance sampling, and the Lyapunov function method for heavy-tailed distributions.

Efficiency Criteria

There are two commonly used criteria for the performance of a Monte Carlo algorithm in rare event

simulation. Consider a family of rare event probabilities $\{p_n\}$ such that $p_n \to 0$ as $n \to \infty$. One can think of n as an index for rarity. For example, p_n may denote the probability that a one dimensional simple random walk with negative drift ever crosses a large threshold n, starting at the origin.

Consider a Monte Carlo algorithm for estimating p_n, where the estimate is the sample mean of independent copies of some random variable Y_n that satisfies $E[Y_n] = p_n$. Then the estimate is unbiased. The estimate is said to have bounded relative error if

$$\limsup_{n\to\infty} \frac{\mathrm{Var}[Y_n]}{p_n^2} < \infty.$$

It is not difficult to show that the number of samples required to achieved a fixed relative error remains bounded as n increases.

Since Y_n is unbiased, minimizing its variance is equivalent to minimizing its second moment. By Jensen's inequality, $E\left[Y_n^2\right] \geq (EY_n)^2 = p_n^2$. This motivates a weaker notion of efficiency, namely, the logarithmic asymptotic efficiency or asymptotic efficiency, which holds if

$$\lim_{n\to\infty} \frac{\log E\left[Y_n^2\right]}{\log p_n} = 2. \tag{1}$$

This criterion is particularly convenient when the rare event probabilities $\{p_n\}$ satisfy the large deviation asymptotics

$$\lim_{n\to\infty} \frac{1}{n} \log p_n = -\gamma,$$

where $\gamma > 0$ is some constant. In this situation, logarithmic asymptotic efficiency amounts to

$$\lim_{n\to\infty} \frac{1}{n} \log E\left[Y_n^2\right] = -2\gamma,$$

and implies that the number of samples required to achieve a fixed relative error grow sub-exponentially as n increases. In the literature, logarithmic asymptotic efficiency is sometimes referred to as asymptotic optimality.

Two Illustrative Examples

Even though the methodologies presented here can be applied to general settings, to convey the main ideas, two examples will be used to illustrate various Monte Carlo schemes.

Simple Random Walk [SRW]. Let $\{Z_i\}$ be a sequence of $\mathbb{R}^d$-valued, independent identically distributed random variables with distribution μ, and assume that the log-moment generating function

$$H(\alpha) = \log E\left[e^{\langle \alpha, Z_1\rangle}\right] = \log \int_{\mathbb{R}^d} e^{\langle \alpha, x\rangle} \mu(dx)$$

is finite for every $\alpha \in \mathbb{R}^d$. Define for $n \geq 1$, $S_n = Z_1 + \cdots + Z_n$. For some Borel set $A \subset \mathbb{R}^d$, it is of interest to estimate the probability

$$p_n = P\left(\frac{S_n}{n} \in A\right).$$

Under some mild conditions, the large deviations asymptotics hold, namely,

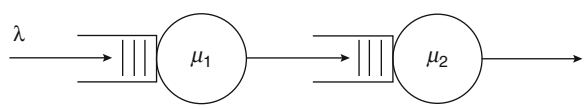
$$\lim_{n\to\infty} \frac{1}{n} \log p_n = -\inf_{\beta\in A} L(\beta), \tag{2}$$

where L is the Legendre transform of H:

$$L(\beta) = \sup_{\alpha\in\mathbb{R}^d} [\langle \alpha, \beta\rangle - H(\alpha)]. \tag{3}$$

Tandem Queueing Network [TQN]. Consider a two-node tandem Jackson queueing network, where the arrival process is Poisson with rate λ and the service times are exponentially distributed with rate μ_1 and μ_2, respectively. The system is assumed to be stable, that is, $\lambda < \min\{\mu_1, \mu_2\}$. See Fig. 1.

Assume that the two queues share a single buffer with total capacity n. The quantity of interest is the buffer overflow probability



Rare Event Simulation, Fig. 1 Two-node tandem queue

$$p_n = P\{\text{network total population reaches } n \text{ before returning to 0, starting from 0}\}.$$

Glasserman and Kou (1995) established the large deviation limit

$$\lim_n \frac{1}{n} \log p_n = -\log \frac{\min\{\mu_1, \mu_2\}}{\lambda}.$$

Importance Sampling

The basic setup of importance sampling is as follows. Suppose that one is interested in estimating

$$p = P(X \in A),$$

where X is a random variable with distribution μ. Importance sampling generates samples from a different probability distribution ν and uses the sample mean of independent copies of

$$Y = 1_{\{X \in A\}} \frac{d\mu}{d\nu}(X)$$

as the estimate. One usually requires that μ be absolutely continuous with respect to ν so that the likelihood ratio $d\mu/d\nu$ is well defined. This requirement can be relaxed as long as the absolute continuity holds for the restrictions of μ and ν to the set A. Note that the estimate Y is unbiased since

$$E_\nu[Y] = \int_A \frac{d\mu}{d\nu}(x) d\nu(x) = \int_A d\mu(x) = P(X \in A).$$

Here $E_\nu[\cdot]$ denotes the expectation taken under the probability distribution ν.

The key question in importance sampling is the choice the sampling distribution ν. Ideally, one would like to find the one that minimizes the variance of Y. To this end, define a measure ν^* such that

$$\frac{d\nu^*}{d\mu}(x) = \frac{1}{p} \cdot 1_{\{x \in A\}}.$$

It is not difficult to verify that ν^* is a probability distribution and the corresponding importance sampling estimator Y has variance zero. However, such a probability measure is of little practical use since it requires the knowledge of p, the unknown quantity one wishes to estimate. Therefore, instead of this unconstrained optimization, it is typical to search within a parameterized family of alternative probability measures. When the problem can be cast into the framework of efficiency criteria, it is desirable for the estimator to achieve logarithmic asymptotic efficiency or bounded relative error.

Remark. For future analysis, observe that the second moment of the importance sampling estimate Y admits a very simple form

$$E_\nu[Y^2] = \int_A \left(\frac{d\nu}{d\mu}\right)^2 (x) d\nu(x) = \int_A \frac{d\nu}{d\mu}(x) d\mu(x) = E_\mu[Y].$$

Classical Results in Importance Sampling

Siegmund (1976) was the first to argue that, using an exponential change of measure, asymptotically efficient importance sampling schemes can be built for estimating gambler's ruin probabilities. The analysis was related to the theory of large deviations, which has since become an indispensable tool for the design of efficient Monte Carlo algorithms.

To illustrate the idea, consider the example of SRW where A is assumed to be a closed convex set. Suppose that instead of generating the increments $\{Z_i\}$ according to μ, samples of $\{Z_i\}$ from an exponential change of measure ν_α are generated where

$$\nu_\alpha(dx) = e^{\langle \alpha, x \rangle - H(\alpha)} \mu(dx)$$

for some $\alpha \in \mathbb{R}^d$. The importance sampling estimate is

$$Y_n = 1_{\{S_n/n \in A\}} \prod_{i=1}^n e^{-\langle \alpha, Z_i \rangle + H(\alpha)} = 1_{\{S_n/n \in A\}} e^{-\langle \alpha, S_n \rangle + nH(\alpha)}.$$

Taking into account Remark 0.1 and application of Varadhan's Lemma, the second moment of Y_n satisfies

$$\lim_{n\to\infty}\frac{1}{n}\log E_{v_\alpha}\left[Y_n^2\right] = \lim_{n\to\infty}\frac{1}{n}\log E_\mu[Y_n] = -\inf_{\beta\in A}\left[\langle\alpha,\beta\rangle - H(\alpha)+L(\beta)\right].$$

The α^* that minimizes the right-hand-side of the above display yields the asymptotically most efficient exponential change of measure (say $v^* = v_{\alpha^*}$), and is the solution to the min/max problem

$$\sup_{\alpha\in\mathbb{R}^d}\inf_{\beta\in A}\left[\langle\alpha,\beta\rangle - H(\alpha)+L(\beta)\right]. \tag{4}$$

Since A is closed and convex, it is valid to exchange of the order of the sup and inf in the above expression. Then it follows from (3) that

$$\lim_{n\to\infty}\frac{1}{n}\log E_{v^*}\left[Y_n^2\right] = -\inf_{\beta\in A}\sup_{\alpha\in\mathbb{R}^d}\left[\langle\alpha,\beta\rangle - H(\alpha)+L(\beta)\right] = -2\inf_{\beta\in A}L(\beta).$$

In other words, $v^* = v_{\alpha^*}$ is logarithmic asymptotically efficient. Furthermore, if β^* minimizes $L(\beta)$ over $\beta\in A$, then α^* can be identified as the conjugate point of β^* or the point that maximizes $\langle\alpha,\beta^*\rangle - L(\beta^*)$ over $\alpha\in\mathbb{R}^d$.

It turns out that v^* coincides with the change of measure used in the classical proof of the large deviations lower bound for the rare event probabilities $P(S_n/n\in A)$. This formal connection between importance sampling and the theory of large deviations has been subsequently explored by many and made rigorous under certain circumstances, e.g., Asmussen (1985), Heidelberger (1995), Asmussen and Glynn (2007) and the references therein. These investigations gave rise to an entirely new community using exponential change of measure as the driving force for importance sampling.

Glasserman and Kou (1995) was the first to challenge the standard heuristic that the change of measure used in the proof of the large deviation lower bound should perform well. The paper considered a change of measure proposed by Parekh and Walrand (1989) for the example of TQN, which amounts to interchanging the arrival rate and the smallest service rate, and showed that it failed to be asymptotically efficient in general. In Glasserman and Wang (1997), counterexamples were constructed, such as SRW with a non-convex target set A, to show that the importance sampling estimator based on the standard heuristic can be less efficient than the standard Monte Carlo. In retrospect, the failure of the standard heuristic even in very simplistic settings is not surprising. In the previous analysis of the SRW model, a key assumption is that A is convex so that the sup and inf in (1) can be interchanged. This is clearly not true when A is a general non-convex set. The work of Glasserman and Kou (1995) and Glasserman and Wang (1997) made it clear that the standard heuristic had to be applied with great caution and motivated the development of general methodologies such as dynamic importance sampling. Some of these development will be reviewed later.

Cross-Entropy Method

The cross-entropy method is a Monte Carlo technique that originated from a sequence of papers Rubinstein (1997, 1999), which can be used not only for estimating rare event probabilities, but also for solving difficult combinatorial optimization problems; seede Boer et al. (2005) for a tutorial and Rubinstein and Kroese (2004) for a comprehensive treatment.

Consider the generic importance sampling problem for estimating $p = P(X\in A)$, where X is a random variable with distribution μ. Assume that the alternative sampling distribution is restricted to a prescribed, parameterized family of distributions, say $\{\mu_\theta : \theta\in\Theta\}$, that contains the original distribution μ. The reference parameter θ is sometimes termed the tilting parameter. As discussed previously, the zero-variance change of measure v^* is defined by

$$\frac{dv^*}{d\mu} = \frac{1}{p}\cdot 1_{\{x\in A\}}. \tag{5}$$

Under the natural assumption that a sampling distribution close to v^* should be a good choice for importance sampling, the cross-entropy method aims to solve for the distribution μ_θ that is closest to v^* under the Kullback-Leibler distance. This leads to the minimization problem

$$\min_{\theta\in\Theta} R(v^* \parallel \mu_\theta), \tag{6}$$

where $R(\cdot\|\cdot)$ is the Kullback-Leibler cross-entropy, or relative entropy, defined by

$$R(\nu \parallel \mu) = \int \log \frac{d\nu}{d\mu}(x) d\nu(x)$$

if ν is absolutely continuous with respect to μ and ∞ otherwise. Note that $R(\nu\|\mu)$ is always non-negative and equals zero if and only if $\nu = \mu$.

The cross-entropy method provides a simple iterative procedure to obtain a solution to the optimization problem (6). Every iteration involves two phases: (1) samples are generated from the distribution μ_{θ_t} where θ_t is the current candidate of the tilting parameter; (2) based on these samples, the tilting parameter θ_t is updated to θ_{t+1} in order to produce better samples in the next iteration. The iteration is terminated when the convergence of $\{\theta_t\}$ is reached. Suppose that θ^* is the final tilting parameter. Then μ_{θ^*} is used as the importance sampling change of measure to estimate p, the probability of interest.

A big advantage of the cross-entropy method is that θ_{t+1} can often be solved analytically. In particular, this happens when the distributions $\{\mu_\theta\}$ belong to the family of exponential changes of measure, as will be shown in more detail later.

The initialization θ_0 of the cross-entropy algorithms is quite flexible in general. For example, in many situations one can simply choose θ_0 that corresponds to the original distribution μ. However, in the context of rare event simulation, the choice of θ_0 becomes less straightforward. These issues will be discussed shortly.

The Adaptive Updating of θ

Consider the minimization problem (6). Denote by W_θ the likelihood ratio function

$$W(x;\theta) = \frac{d\mu}{d\mu_\theta}(x).$$

Plugging in the formula (5), it follows that

$$\begin{aligned} R(\nu^* \parallel \mu_\theta) &= \int \log \frac{d\nu^*}{d\mu_\theta}(x) d\nu^*(x) \\ &= -\log p + \frac{1}{p} \int 1_{\{x\in A\}} \log W(x;\theta) d\mu(x). \end{aligned}$$

Therefore the minimization problem (6) amounts to minimizing over $\theta \in \Theta$ the integral

$$\int 1_{\{x\in A\}} \log W(x;\theta) d\mu(x).$$

Now let $\gamma \in \Theta$ be an arbitrary reference parameter. Then the above integral equals

$$\begin{aligned} &\int 1_{\{x\in A\}} W(x;\gamma) \log W(x;\theta) d\mu_\gamma(x) \\ &= E_\gamma \left[1_{\{X\in A\}} W(X;\gamma) \log W(X;\theta) \right], \end{aligned}$$

where $E_\gamma[\cdot]$ means that the expectation is taken with X distributed according to μ_γ. Therefore the minimization problem (4.5) is equivalent to the minimization problem

$$\min_{\theta\in\Theta} E_\gamma \left[1_{\{X\in A\}} W(X;\gamma) \log W(X;\theta) \right], \tag{7}$$

for any arbitrarily fixed $\gamma \in \Theta$. In the cross-entropy method, the minimizing θ is estimated by solving the corresponding stochastic program

$$\min_{\theta\in\Theta} \frac{1}{N} \sum_{i=1}^{N} 1_{\{X_i\in A\}} W(X_i;\gamma) \log W(X_i;\theta), \tag{8}$$

where $\{X_1,\ldots,X_N\}$ are independent samples from the distribution μ_γ. The function in (8) is convex and differentiable with respect to θ in typical applications. Thus the minimizing θ is the solution to the equation

$$\frac{1}{N} \sum_{i=1}^{N} 1_{\{X_i\in A\}} W(X_i;\gamma) \nabla \log W(X_i;\theta) = 0,$$

where the gradient ∇ is with respect to θ.

Stated below is the basic adaptive updating rule for the tilting parameter θ in the cross-entropy method. The stochastic program (8) will be used in lieu of the deterministic program (7).

The basic updating rule of θ.

Suppose $\hat{\theta}_t$ is the value of the tilting parameter at the end of the last iteration. Generate independent samples $\{X_1,\ldots,X_N\}$ from the distribution $\mu_{\hat{\theta}_t}$. Define

$$\hat{\theta}_{t+1} = \arg\min_\theta \frac{1}{N} \sum_{i=1}^{N} 1_{\{X_i\in A\}} W\left(X_i;\hat{\theta}_t\right) \log W(X_i;\theta). \tag{9}$$

The iteration continues until a prescribed convergence criterion of $\hat{\theta}_t$ is satisfied.

As mentioned previously, the minimization problem (9) can often be solved analytically (the connection between this minimization problem and the maximum likelihood estimate can be found in Asmussen and Glynn 2007). For illustration, consider the case where $\{\mu_\theta\}$ is the family of exponential changes of measures of the original distribution μ. That is,

$$\frac{d\mu_\theta}{d\mu}(x) = e^{\langle\theta,x\rangle - H(\theta)}$$

Then $\log W(X_i;\theta) = H(\theta) - \langle\theta, X_i\rangle$ and $\nabla W(X_i;\theta) = \nabla H(\theta) - X_i$. It follows easily that the minimizing $\hat{\theta}_{t+1}$ satisfies

$$\nabla H\left(\hat{\theta}_{t+1}\right) = \frac{\sum_{i=1}^N 1_{\{X_i\in A\}} W\left(X_i;\hat{\theta}_t\right) X_i}{\sum_{i=1}^N 1_{\{X_i\in A\}} W\left(X_i;\hat{\theta}_t\right)}.$$

Note that $\nabla H(\theta)$ equals the expected value of a random variable with distribution μ_θ. Therefore if the family of distributions $\{\mu_\theta\}$ is reparametrized by the mean v, then the classical cross-entropy updating formula follows:

$$\hat{v}_{t+1} = \frac{\sum_{i=1}^N 1_{\{X_i\in A\}} W(X_i;\hat{v}_t) X_i}{\sum_{i=1}^N 1_{\{X_i\in A\}} W(X_i;\hat{v}_t)}. \tag{10}$$

This formula actually holds when the distributions $\{\mu_\theta\}$ belongs to a more general natural exponential family that is reparametrized by the mean; see Appendix A.3 of Rubinstein and Kroese (2008).

Example 1. Consider the SRW model. Let $X = (Z_1,\ldots,Z_n)$ where Z_j's are independent with common distribution μ. Denote by $\{\mu_\theta : \theta \in \Theta\}$ the family of exponential change of measure of μ, that is,

$$\frac{d\mu\theta}{d\mu}(z) = e^{\langle\theta,z\rangle - H(\theta)}$$

Suppose that the family of candidate sampling distributions of X is $\{v_\theta : \theta \in \Theta\}$ such that under v_θ, Z_j's are independent with common distribution μ_θ. Then the likelihood ratio $W(x;\theta)$ is given by

$$W(x;\theta) = \prod_{j=1}^n e^{\langle\theta,zj\rangle - H(\theta)} = e^{n\langle\theta,\bar{S}(x)\rangle - nH(\theta)},$$

where $x = (z_1,\ldots,z_n)$ and $\bar{S}(x) = (z_1 + \cdots + z_n)/n$. It is not difficult to solve the updating formula (9) to obtain

$$\nabla H\left(\hat{\theta}_{t+1}\right) = \frac{\sum_{i=1}^N 1_{\{X_i\in A\}} W\left(X_i;\hat{\theta}_t\right)\bar{S}(X_i)}{\sum_{i=j}^N 1_{\{X_i\in A\}} W\left(X_i;\hat{\theta}_t\right)},$$

where $X_i = (Z_1^{(i)},\ldots,Z_n^{(i)})$, $i = 1,\ldots,N$, and $Z_j^{(i)}$ are independent samples from the common distribution $\mu_{\hat{\theta}_t}$. As before, reparametrizing the distribution v_θ by the mean v, the formula becomes

$$\hat{v}_{t+1} = \frac{\sum_{i=1}^N 1_{\{X_i\in A\}} W(X_i;\hat{v}_t)\bar{S}(X_i)}{\sum_{i=1}^N 1_{\{X_i\in A\}} W(X_i;\hat{v}_t)}.$$

In other words, the mean of the updated sampling distribution is the weighted average of the sample path means.

Example 2. Consider the TQN model. Suppose that the family of candidate sampling distributions is $\{P_v : v = (v_1, v_2, v_3), v_i > 0\}$ such that under P_v the system is a Jackson network with exponential interarrival times of mean v_1, and exponential service times of mean v_2 and v_3, respectively. The original distribution corresponds to $v_0 = (1/\lambda, 1/\mu_1, 1/\mu_2)$.

Let $X_1,\ldots,X_N$ be independent sample paths generated from the distribution $P_{\hat{v}_t}$, each of which starts from the origin and stops at the first time either the total population hits the level n or the system becomes empty again. For a sample path X, denote by $\tau_1(X)$, $\tau_2(X)$, and $\tau_3(X)$ the total number of interarrivals, service completion at node 1, and service completion at node 2, respectively. Let $\{Y_{1j}(X) : j = 1,\ldots,\tau_1(X)\}$ be the interarrival times. Similarly, let $\{Y_{2j}(X) : j = 1,\ldots,\tau_2(X)\}$ and $\{Y_{3j}(X) : j = 1,\ldots,\tau_3(X)\}$ be the service times at node 1 and node 2, respectively. Then the density of a sample path X under the distribution P_v equals

$$f(X;v) = \prod_{k=1}^3 \prod_{j=1}^{\tau_k(X)} \frac{1}{v_k} e^{-Y_{kj}(X)/v_k}$$

and the likelihood ratio W is given by

$$W(X;v) = \frac{f(X;v_0)}{f(X;v)}.$$

Denote by A the buffer overflow event. It is not difficult to solve the stochastic program (9) to obtain the analytic formula

$$\hat{v}_{t+1,k} = \frac{\sum_{i=1}^{N} 1_{\{X_i \in A\}} W(X_i; \hat{v}_t) \sum_{j=1}^{\tau_k(X_1)} Y_{kj}(X_i)}{\sum_{i=1}^{N} 1_{\{X_i \in A\}} W(X_i; \hat{v}_t) \tau_k(X_i)}$$

for $k = 1,2,3$. This updating formula is actually valid for much more complicated queueing networks; see de Boer et al. (2004) for more details.

The Initialization in Rare Event Simulation

The initialization of the cross-entropy algorithm, or the choice of $\hat{\theta}_0$, can be quite flexible in general. It usually suffices to set $\hat{\theta}_0 = \theta_0$ where θ_0 corresponds to the original distribution. However, this recipe is problematic in the context of rare event simulation, since most likely the indicator $1_{\{X_i \in A\}}$ will be zero for all i if A is a rare event, rendering the minimization problem (9) meaningless.

A possible approach is as follows. Choose a set $B \supseteq A$ so that it is much less rare than A but shares the same qualitative flavor, e.g., in the TQN example one may choose B to be the event of total population overflow with buffer size $m \ll n$. Setting $\hat{\theta}_0 = \theta_0$, a pilot cross-entropy algorithm delivers the nearly optimal tilting parameter (say) θ^* for estimating $P(X \in B)$. Next with $\hat{\theta}_0 = \theta^*$, the main cross-entropy algorithm is performed to yield the optimal tilting parameter for estimating the actual probability of interest $P(X \in A)$. De Boer et al. (2004) used this approach to estimate buffer overflow probabilities in queueing networks, where B is chosen as the buffer overflow event with a small buffer level.

Generalizing this idea, a two-stage iterative scheme where both the set B and the tilting parameter θ are updated seems to be more convenient for most problems. To describe the idea, assume that the probability of interest is

$$p = P(S(X) \geq \gamma) = P\big(X \in A_\gamma\big)$$

where γ is a fixed level, S is some performance measure, and $A_\gamma = \{x : S(x) \geq \gamma\}$. It is assumed that γ is large and A_γ is a rare event. As before, the distribution of X is denoted by μ and $\{\mu_\theta : \theta \in \Theta\}$ is a parametrized family of candidate sampling distribution. In this two-stage approach for estimating p, one generates a sequence of tilting parameters $\{\hat{\theta}_t\}$, as well as a sequence of levels $\{\hat{\gamma}_t\}$ that are determined by the samples and generally increase to the actually fixed large level γ. In essence, these artificial intermediate levels divide the original difficult rare event A into a sequence of easier, less rare events $A_{\hat{\gamma}_t}$.

The algorithm is as follows. Fix a priori a fraction ρ that is not too small, usually between 1 and 10%. Set $\hat{\theta}_0 = \theta_0$ and generate N samples $X_1,\ldots,X_N$ from the distribution $\mu_{\hat{\theta}_0}$. Estimate the $(1-\rho)$-quantile of $S(X)$ by the sample quantile. That is, order the performances $S(X_i)$ from the smallest to the largest: $S_{(1)} \leq \cdots \leq S_{(N)}$ and define

$$\hat{\gamma}_1 = S_{(N_e)},\ N_e = \lceil (1-\rho)N \rceil,$$

where $[x]$ is the ceiling of x or the smallest integer that is greater than or equal to x. Then update the tilting parameter as in (9) with the set A replaced by $A_{\hat{\gamma}_1}$. In other words, $\hat{\theta}_1$ is estimated on the basis of those samples X_i that satisfies $S(X_i) \geq \hat{\gamma}_1$ and there are about ρN of them (elite samples). Iterated these steps until $\hat{\gamma}_t \geq \gamma$. The algorithm is summarized as follows:

Main Cross-Entropy Algorithm for Rare Event Simulation

1. Let $\hat{\theta}_0 = \theta_0$ and $t = 0$ (iteration counter).
2. Generate samples $X_1,\ldots,X_N$ from the distributions $\mu_{\hat{\theta}_t}$. Calculate the performances $S(X_i)$ and order them from the smallest to the largest: $S_{(1)} \leq \cdots \leq S_{(N)}$ and define $\hat{\gamma}_{t+1} = \min\{S_{(N_e)}, \gamma\}$.
3. Use these samples $X_1,\ldots,X_N$ to solve the stochastic program (9) with the set A replaced by $A_{\hat{\gamma}_{t+1}}$.
4. If $\hat{\gamma}_{t+1} < \gamma$, set $t = t + 1$ and reiterate from Step 2. Otherwise, proceed with Step 5.
5. Let T be the final iteration counter. Estimate the rare event probability p by importance sampling, with the final tilting parameter $\hat{\theta}_T$.

Sometimes between Step 4 and Step 5, one can refine the final tilting parameter by running a few extra iterations of the standard cross-entropy updating program (9) with $\hat{\theta}_T$ as the initial tilting parameter and the set A fixed as A_γ. The analysis of the convergence properties of this algorithm can be found in R.Y. Rubinstein and D.P. Kroese (2004) and Costa et al. (2007).

Dynamic Importance Sampling

The notion of dynamic, or state-dependent importance sampling was introduced in Dupuis and Wang (2004). The development of this methodology was partly motivated by the counterexamples in Glasserman and Kou (1995) and Glasserman and Wang (1997) that had challenged the validity of the standard heuristic. It was shown in Dupuis and Wang (2004) that the second moment of an importance sampling estimator can be interpreted as the value of a small noise stochastic game. In this context it was obvious that the heuristic approach, which amounted to allowing only those state independent changes of measure)) or open loop controls in the language of stochastic games), could not possibly be asymptotically efficient in general; see also Bassamboo et al. (2006). This connection also linked importance sampling to the Isaacs equation of a limiting differential game, which turned out to be *equivalent* to the Hamilton-Jacobi-Bellman (HJB) equation associated with the corresponding large deviation rate function. As a consequence, the solution to this HJB equation can be used to construct asymptotically efficient importance sampling schemes.

Dupuis and Wang (2007) explored this connection in further depth and showed that the design and analysis of dynamic importance sampling algorithms could be based on the classical subsolutions to the HJB equation. One can often construct subsolutions that are structurally much simpler than the actual solution, but which correspond to asymptotically efficient importance sampling schemes that reflect this simplicity. Subsolutions provide a unifying and flexible tool and can be used to study to a broad range of process models; see, e.g., Dupuis et al. (2007) and Dupuis and Wang (2007, 2009).

Limit Differential Game and its Isaacs Equation

In order to formally illustrate the connection between importance sampling and small noise stochastic games, consider the SRW model. Recall that $\{Z_1,\ldots,Z_n\}$ is a sequence of independent random variables with common distribution μ. Define the scaled random walk process

$$X_j = \frac{1}{n}\sum_{i=1}^{j} Z_i, \quad j = 1,\ldots,n, \tag{11}$$

with $X_0 = 0$. The probability of interest is $p_n = P(X_n \in A)$. As before, one can define an exponential change of measure ν_α by

$$\nu_\alpha(dx) = e^{\langle \alpha, x\rangle - H(\alpha)}\mu(dx)$$

for every $\alpha \in \mathbb{R}^d$.

Consider a state-dependent change of measure in the following sense. For each $j = 0, 1, \ldots, n-1$, conditional on the simulation history $\{Z_i : i = 1,\ldots,j\}$, Z_{j+1} is sampled from a distribution μ_{α_j}, where α_j is a function of both the scaled time j/n and the scaled state X_j as defined in (11). The corresponding importance sampling estimator is given by

$$Y_n = 1_{\{X_n \in A\}} \prod_{j=0}^{n-1} e^{-\langle \alpha_j, Z_{j+1}\rangle + H(\alpha_j)}.$$

The estimate Y_n is unbiased. The goal is to minimize the variance, or equivalently, the second moment of Y_n.

This minimization problem connects naturally to a partial differential equation when it is recast as a stochastic control problem with $\{\alpha_j\}$ being the control. To this end extend the problem slightly to allow a general initial time and state. For $i \geq 0$ and $x \in \mathbb{R}^d$, define X_j for $j = i,\ldots,n$ as above except that $X_i = x$, and then define

$$V_n(x,i) = \inf_{\{\alpha_j\}} \bar{E}\left[1_{\{X_n \in A\}} \prod_{j=i}^{n-1} e^{-\langle \alpha_j, Z_{j+1}\rangle + H(\alpha_j)}\right]^2,$$

where $\bar{E}$ denotes the expectation taken under the change of measure determined by the control $\{\alpha_j\}$. In other words, $V_n(x,i)$ is the minimal second moment of the importance sampling estimators given that the state process $\{X_j\}$ startsat time i with initial state x. It will be more convenient to express this in terms of the original distributions:

$$V_n(x,i) = \inf_{\{\alpha_j\}} E\left[1_{\{X_n \in A\}} \prod_{j=i}^{n-1} e^{-\langle \alpha_j, Z_{j+1}\rangle + H\{\alpha_j\}}\right],$$

where the expected value is taken such that $\{Z_j,\ldots,Z_n\}$ are independent with common distribution μ.

As the value function of a discrete time stochastic control problem, V_n satisfies the dynamic programming equation

$$V_n(x,i) = \inf_{\alpha\in\mathbb{R}^d} \int e^{H(\alpha)-\langle\alpha,y\rangle} V_n\left(x+\frac{y}{n}, i+1\right)\mu(dy). \tag{12}$$

Owing to the exponential scaling in n, it is natural to consider the logarithmic transform of V_n and assume that

$$-\frac{1}{n}\log V_n(x,i) \approx W(x,i/n) \tag{13}$$

for some smooth function $W : \mathbb{R}^d \times [0,1] \to \mathbb{R}$. This leads to the approximation

$$V_n\left(x+\frac{y}{n}, i+1\right)\cdot V_n^{-1}(x,i) \approx exp\left\{-\langle\nabla W(x,t), y\rangle - \frac{\partial W}{\partial t}(x,t)\right\}$$

where ∇ is the gradient with respect to x, and $t = i/n$. Plugging the above approximation into equation (12), taking log on both sides, and recalling the definition of H, it follows that

$$0 = -\frac{\partial W}{\partial t} + \inf_{\alpha\in\mathbb{R}^d}[H(\alpha) + H(-\nabla W - \alpha)]. \tag{14}$$

Since $V_n(x,n) = 1_{\{x\in A\}}$, W satisfies the boundary condition $W(x,1) = 0$ if $x\in A$ and ∞ otherwise.

Even though a very special model has been used here, equation (14) leads to a few observations that actually hold in much greater generality.

1. Equation (14) is the Isaacs equation associated with a two-person zero-sum game. Indeed, since H and L are convex duals, for every $\alpha\in\mathbb{R}^d$

$$H(\alpha) = \sup_{\beta\in\mathbb{R}^d}[<\alpha,\beta> - L(\beta)].$$

Thus equation (14) can be written as

$$0 = \frac{\partial W}{\partial t} + \sup_{\alpha}\inf_{\beta}[\langle\nabla W,\beta\rangle + L(\beta) + \langle\alpha,\beta\rangle - H(\alpha)].$$

This is the Isaacs equation corresponds to the following zero-sum differential game. The dynamics $\dot{\phi}(t) = \beta(t)$ only involves the β-player. The running cost $L(\beta) + \langle\alpha,\beta\rangle - H(\alpha)$ is affected by both players, and the terminal cost is $\infty\cdot 1_{A^c}$. Because of the intervening minus sign, the maximizing α-player indeed tries to minimize the variance.

2. Thanks to the convexity of H, the maximizing α in equation (14) is

$$\alpha^*(x,t) = -\frac{1}{2}\nabla W(x,t). \tag{15}$$

This is the basic formula for computing the state-dependent change of measure.

3. Plugging the formula of α^* into equation (14), it follows that

$$0 = \frac{\partial W}{\partial t} + 2H(-\nabla W/2). \tag{16}$$

This is equivalent to the HJB equation associated with the corresponding large deviation rate function. To see this, abuse the notation and extend the definition of p_n to

$$p_n(x,t) = P\left(X_n \in A | X_{\lceil nt\rceil} = x\right)$$

for $x \in \mathbb{R}$ and $t\in[0,1]$. Clearly the probability of interest $P(X_n\in A)$ equals $p_n(0,0)$. Then under suitable conditions

$$-\lim_n \frac{1}{n}\log p_n(x,t) = \inf_{\phi}\int_t^1 L\left(\dot{\phi}(s)\right)ds,$$

where the infimum is taken over all absolutely continuous functions ϕ such that $\phi(t) = x$ and $\phi(1)\in A$. Denote by $U(x,t)$ the value function of this minimization problem. Then U satisfies the HJB equation

$$0 = \inf_{\beta}\left[\frac{\partial U}{\partial t} + \langle\nabla U,\beta\rangle + L(\beta)\right] = \frac{\partial U}{\partial t} + H(-\nabla U)$$

with terminal condition $U(x,1) = 0$ if $x\in A$ and ∞ otherwise. Clearly it is equivalent to (16) by a change of variable $W = 2U$. This equivalence also

indicates that the state-dependent change of measure based on the solution to the Isaacs equation (16) is asymptotically efficient since by equation (13),

$$-\lim_n \frac{1}{n}\log V_n(0,0) = W(0,0) = 2U(0,0) = -2\lim_n \frac{1}{n}\log p_n(0,0),$$

and $V_n(0, 0)$ is the second moment of the corresponding importance sampling estimator. A rigorous proof can be found in Dupuis and Wang (2004).

The Idea of Subsolutions

From the previous discussion, it follows that the solution to a related Isaacs equation can be used to build asymptotically efficient importance sampling schemes. A difficulty with this approach is that a solution to a nonlinear partial differential equation such as Isaacs equation is hard to compute. To circumvent this, Dupuis and Wang (2007) proposed importance sampling schemes based on the subsolutions to the Isaacs equation. Subsolutions are functions that satisfy the partial differential equation with inequality instead of equality, and allow much greater flexibility in the design of importance sampling schemes.

In order to understand the sufficiency of subsolution, examine the criterion of logarithmic asymptotic efficiency more closely. Recall the definition of logarithmic asymptotic efficiency (1). Jensen's inequality implies that

$$\log E\left[Y_n^2\right] \geq 2\log E[Y_n] = 2\log p_n.$$

Therefore, if for some $\gamma > 0$ the large deviation asymptotics

$$\lim_{n\to\infty} \frac{1}{n}\log p_n = -\gamma$$

hold, then (1) is equivalent to the inequality

$$\limsup_{n\to\infty} \frac{1}{n}\log E\left[Y_n^2\right] \leq -2\gamma. \tag{17}$$

In other words, in order to show that Y_n is asymptotically efficient, it suffices to establish the upperbound (17) only. The inequalities in the definition of a subsolution [see below] are consistent with this upper-bound, when the subsolution is combined with a verification argument to bound the second moment of Y_n.

To give the definition of a subsolution, consider a family of Isaacs equations of a given form. The definition easily extends to other types of Isaacs equations. For a broad collection of problems, the probability of interest is of form $p_n = P(S_n/n \in A)$, where S_n is the partial sum of independent identically distributed random variables or functionals of Markov chains. Then under suitable conditions, the large deviations asymptotics

$$\lim_{n\to\infty} \frac{1}{n}\log p_n = -\inf_{\beta\in A} L(\beta) := -\gamma$$

hold for some convex rate function L. Denoting by H the Legendre transform of L, then the Isaacs equation takes the familiar form

$$\frac{\partial W}{\partial t} + \sup_\alpha \inf_\beta \left[\langle \nabla W, \beta\rangle + L(\beta) + \langle \alpha, \beta\rangle - H(\alpha)\right] = 0, \tag{18}$$

with boundary condition $W(x,1) = 0$ if $x \in A$ and ∞ otherwise.

Definition. A classical subsolution (18) to the Isaacs equation is a smooth function W that satisfies

$$\frac{\partial W}{\partial t} + \sup_\alpha \inf_\beta \left[\langle \nabla W, \beta\rangle + L(\beta) + \langle \alpha, \beta\rangle - H(\alpha)\right] \geq 0$$

with boundary inequality $W(x,1) \leq 0$ for $x \in A$.

Given a classical subsolution W, the corresponding change of measure is determined by the maximizing α^* for the min/max term, which has exactly the same form as (15). The following theorem is the key result in the performance analysis of those importance sampling schemes based on subsolutions; see Dupuis and Wang (2007) for more details.

Theorem 1. Let W be a classical subsolution to the Isaacs equation and Y_n the corresponding importance sampling estimate of p_n. Then under suitable conditions

$$\limsup_{n\to\infty} \frac{1}{n} \log E\left[Y_n^2\right] \le -W(0,0). \tag{19}$$

In particular, if $W(0, 0) = 2\gamma$, then Y_n is asymptotically efficient.

Construction of Subsolutions

Theorem 1 reduces the problem of building an asymptotically efficient or nearly asymptotically efficient importance sampling scheme to that of a classical subsolution W with $W(0, 0)$ equal or close to 2γ, respectively. For systems with piecewise homogeneous dynamics, a particularly useful technique is to build a piecewise affine subsolution at first and then obtain a classical subsolution by mollification. The construction of a piecewise affine subsolution, which is usually identified as the minimum of a collection of affine functions, is the key step. Once such a piecewise affine subsolution is given, say,

$$\bar{W} = W_1 \wedge \cdots \wedge W_m,$$

a mollification technique called exponential weighting is used to produce a smooth function:

$$W_\varepsilon = -\varepsilon \log \sum_{i=1}^{m} e^{-W_i/\varepsilon},$$

where ε is a small positive number. It is not difficult to show that W_ε approximates $\bar{W}$ as ε approaches zero. Furthermore, analytic formulas for quantities such as ∇W_ε are readily available. There are two important remarks regarding the function W_ε and the corresponding change of measure: (1) In general, W_ε is not exactly a subsolution, but an approximate one in the sense that the inequality (19) is satisfied with the right hand side replaced by a vanishing negative number. This is usually sufficient for asymptotic efficiency; (2) It is sometimes more convenient to use a change of measure slightly different from the one determined by $\alpha^* = -\nabla W_\varepsilon/2$. It is essentially a state dependent mixture of the changes of measure determined by $\{W_i\}$. Theorem 1 still holds in this case; see Dupuis and Wang (2007) for details.

In general, the construction of piecewise affine subsolutions is accomplished by carefully analyzing the properties of the system dynamics and the relevant large deviation properties. For illustration, consider two concrete examples.

Example 3. Consider the SRW model. Without loss of generality assume that $E[X_1] = 0$. First consider the simple case where $A = [\beta,\infty)$ for some $\beta > 0$. Denote by α the conjugate point of β. Then the affine function

$$W(x) = -2\langle \alpha, x - \beta\rangle - 2(1-t)H(\alpha)$$

is a subsolution to the Isaacs equation. Since $-\nabla W/2 = \alpha$, the corresponding change of measure is exactly the classical one.

A more interesting case is when $A = \left(-\infty, \bar{\beta}\right] \cup [\beta, \infty)$ when $\bar{\beta}<0<\beta$. Let $\bar{\alpha}$ be the conjugate point of $\bar{\beta}$. Define

$$\bar{W}(x) = -2\left\langle \bar{\alpha}, x - \bar{\beta}\right\rangle - 2(1-t)H(\bar{\alpha}).$$

Then it is not difficult to check that $W^* = W \wedge \bar{W}$ is a two-piece affine subsolution. Note that

$$-\nabla W^*/2 = \begin{cases} \alpha & \text{if } W < \bar{W} \\ \bar{\alpha} & \text{if } W > \bar{W} \end{cases}$$

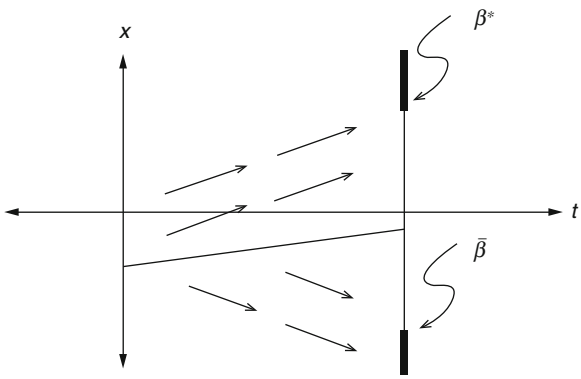
is piecewise constant. Figure 2 illustrates how this would partition the space-time domain.

Example 4. Consider the TQN model. Without loss of generality assume $\lambda + \mu_1 + \mu_2 = 1$. Define by $\left\{Z_k = \left(Z_{k,1}, Z_{k,2}\right) : k = 0, 1, \ldots\right\}$ the embedded discrete time Markov chain, where $Z_{k,i}$ represents the length of the i-th queue at the k-th transition epoch of the network, $i = 1,2$. The space of the possible jumps is

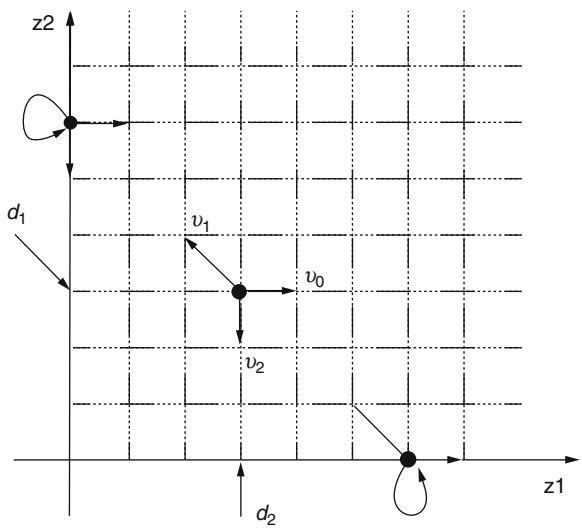
$$\mathbb{V} = \{v_0 = e_1, v_1 = -e_1 + e_2, v_2 = -e_2\}.$$

The system dynamics can be described as $Z_{k+1} = Z_k + \pi[Z_k, Y_{k+1}]$, where $\{Y_k\}$ are random variables taking values in $\mathbb{V}$ and π is the mapping due to the non-negativity constraint on the queue lengths: for $x = (x_1, x_2) \in \mathbb{R}_+^2$ and $y \in \mathbb{V}$

$$x[x; y] = \begin{cases} \text{O if } \mathrm{x_i} = 0 \text{ and } \mathrm{y} = \upsilon_\mathrm{i} \text{ for some i} = 1, 2 \\ y \text{ otherwise} \end{cases}$$



Rare Event Simulation, Fig. 2 Domain decomposition and corresponding drifts



Rare Event Simulation, Fig. 3 The system dynamics

See Fig. 3. Let $\mathcal{P}$ be the space of strictly positive probability measures on $\mathbb{V}$, i.e.,

$$\mathcal{P} = \{\theta = (\theta_0, \theta_1, \theta_2) : \theta_0 + \theta_1 + \theta_2 = 1, \theta_i > 0\}.$$

Under the original distribution, $\{Y_k\}$ are independent identically distributed with distribution $\Theta = (\lambda, \mu_1, \mu_2)$. Recall that $R(\cdot\|\cdot)$ denotes the relative entropy. The relevant Isaacs equation is such that for $x \in \{(x_1, x_2) \in \mathbb{R}_+^2 : x_1 + x_2 < 1\}$

$$\sup_{\bar{\Theta}\in\mathcal{P}} \inf_{\theta\in\mathcal{P}} \left[\left\langle \nabla W(x), \sum_{i=0}^{2} \theta_i \cdot \pi[x, v_i] \right\rangle + \sum_{i=0}^{2} \theta_i \log \frac{\bar{\Theta}_i}{\Theta_i} + R(\theta \,\|\, \Theta) \right] = 0,$$

with the boundary condition $W(x) = 0$ when $x_1 + x_2 = 1$. Here $\bar{\Theta}$ corresponds to the change of measure. Given W, the optimal (maximizing) $\bar{\Theta}$ admits an analytic formula.

The definition of a subsolution is just to replace the "=" by "$\geq$" in the Isaacs equation and replace the boundary condition "$W(x) = 0$" by "$W(x) \leq 0$". Simple piecewise affine subsolutions can be constructed. For example, when $\mu_2 \leq \mu_1$, define vectors

$$r_1 = 2\gamma(-1, -1), r_2 = 2\gamma(-1, 0), r_3 = (0, 0).$$

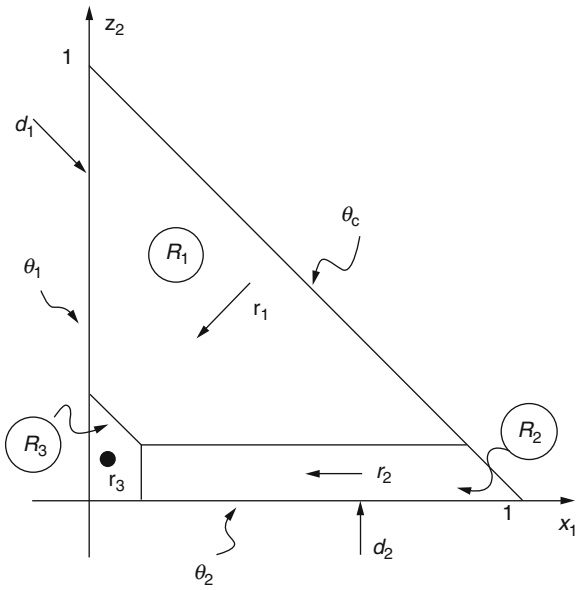
Let δ be a small positive number. Then $\bar{W} = W_1 \wedge W_2 \wedge W_3$ defines a subsolution, where

$$W_k(x) = \langle r_k, x\rangle + 2\gamma - k\delta, \quad k = 1, 2, 3.$$

This subsolution divides the region into three pieces: R_1, R_2, and R_3, such that $\bar{W}(x) = W_k(x)$ for $x \in R_k$. See Fig. 4. The regions R_2 and R_2 are sometimes called "boundary layers". They are closely related to the discontinuity of the dynamics on the boundary $\{x_2 = 0\}$ and the origin, and the large deviations properties of the rare event. Details of the algorithms can be found in Dupuis et al. (2007).

Lyapunov Function Method for Heavy-Tailed Distribution

Much of the previous discussion has assumed that the distributions involved are light-tailed in the sense that their moment generating functions are finite in a small neighborhood of the origin, and thus the exponential changes of measure are meaningful. On the contrary, for a large class of distributions emerging from practice, the tail probabilities decay much more slowly. These heavy-tailed distributions have very different large deviation properties and the exponential scaling is in general not valid. As



Rare Event Simulation, Fig. 4 Piecewise affine subsolution

a consequence, fast rare event simulation algorithms can look very different from those for light-tailed distributions; see Asmussen et al. (2000) and references therein.

Some of the recent works on rare event simulation involving heavy-tailed distributions have been concerned with state-dependent algorithms, e.g., Dupuis et al. (2007). This section reviews a general technique proposed by Blanchet and Glynn (2008) that is based on Lyapunov functions. These Lyapunov functions are closely related to the subsolutions discuss previously (they are in some sense the exponential of subsolutions). Even though the method is applicable to light-tailed distributions as well, the discussion will be made in the context of heavy-tailed distributions, via the example of estimating level crossing probabilities for heavy-tailed random walks.

Let $\{X_i\}$ be a sequence of independent identically distributed heavy-tailed random variables with common distribution μ and strictly negative mean. Define the simple random walk

$$S_n = y + \sum_{i=1}^{n} X_i,$$

with initial condition $S_0 = y$. Here it is assumed that y is a very negative number and the quantity of interest is the level crossing probability

$$p^*(y) = P(S_n \geq 0 \text{ for some } n).$$

Under suitable conditions, as $y \to -\infty$, $p^*(y)$ has the asymptotics

$$p^*(y) \sim \frac{1}{-E[X_1]} \int_{|y|}^{\infty} P(X_1 > s) ds. \tag{20}$$

A useful observation is that if Q^* is a probability measure (Doob's h-transform) that satisfies

$$Q^*(X_{n+1} \in dz | S_n = x) = \frac{p^*(s+x)}{p^*(x)} \mu(dz) \tag{21}$$

for $x < 0$, then Q^* is the zero variance importance sampling change of measure. As before, Q^* is impractical since $p^*(\cdot)$ is unknown.

However, (21) does motivate the use of a change of measure Q such that for $x < 0$

$$Q(X_{n+1} \in dz | S_n = x) = \frac{v(z+x)}{w(x)} \mu(dz), \tag{22}$$

where v is a function that is close to p^* and $w(x)$ is the normalization constant such that

$$w(x) = \int_{\mathbb{R}} v(z+x) \mu(dz). \tag{23}$$

The corresponding importance sampling estimator is just

$$Y = 1_{\{T<\infty\}} \prod_{i=0}^{T-1} \frac{w(S_i)}{v(S_{i+1})},$$

where $T = \inf\{n \geq 1 : S_n \geq 0\}$.

To aid the design of Q, one also need some means to analyze its performance. This is where the Lyapunov function comes into play. Even though the definition here is slightly different from that of Blanchet and Glynn (2008) in the form, they are indeed equivalent.

Definition. A function $H : \mathbb{R} \to [0, \infty)$ is said to be a Lyapunov function associate with the probability measure Q if for every $x < 0$,

$$H(x) \geq \int_{\mathbb{R}} \frac{w(x)}{v(z+x)} H(z+x)\mu(dz).$$

and $H(x) \geq 1$ for $x \geq 0$.

Theorem. Let Y be the importance sampling estimator and H a Lyapunov function associated with Q, then

$$E_Q\left[Y^2 | S_0 = y\right] \leq H(y).$$

Proof. Define the process

$$R_n \doteq H(S_n) \cdot \prod_{i=0}^{n-1} \frac{w^2(S_i)}{v^2(S_{i+1})}.$$

Then it is straightforward to check that the definition of the Lyapunov function H is equivalent to the claim that $R_{T\wedge n}$ is a supermartingale under Q. Therefore by the Optional Sampling Theorem, for $y < 0$

$$E_Q[R_T | S_0 = y] \leq E_Q[R_0 | S_0 = y] = H(y).$$

Observing that $R_T \geq Y^2$ since $H(S_T) \geq 1$, the proof is complete. □

The idea of the Lyapunov function method is to find a pair (v, H) such that (i) v is close to p^* in the sense that they are asymptotically equivalent; (ii) H is a Lyapunov function of the form $H(x) = h(x)v^2(x)$. Then the preceding theorem asserts that the performance of the importance sampling algorithm associated with the change of measure Q is characterized by h. For example, if h is bounded then it is of bounded relative error.

Now consider the objectives (i) and (ii). An immediate problem is that how one can tell if v is close to p^* when p^* is unknown in the first place. The idea is that, comparing (21) and (22), if $v = p^*$ then w defined in (23) should equal p^* as well and thus $w - v = 0$. Therefore, $w - v$ can be used as a criterion to measure how close v and p^* are. With this in mind, it is natural to start with the function on the right-hand-side of (20). Define a non-negative random variable Z that is independent of $\{X_i\}$ and such that for $t > 0$,

$$P(Z > t) = 1 \wedge \int_t^\infty \frac{1}{-E[X_1]} P(X_1 > s)ds.$$

Define $\bar{v}(x) = P(Z > -x)$ for all $x \in \mathbb{R}$. Note that $\bar{v}(x) = 1 = p^*(x)$ if $x > 0$. Furthermore, with $\bar{w}$ defined as in (23) with v replaced by $\bar{v}$, it can be shown that $\bar{w}(x)$ and $\bar{v}(x)$ are asymptotically very close as $x \to -\infty$. Therefore, there exists an $a^* < 0$ such that $v(x) = \bar{v}(x + a^*)$ and $w(x) = w(x + a^*)$ are very close for all $x < 0$. Given the function v, one can find a bounded piecewise constant function h such that $H(x) = h(x)v^2(x)$ defines a Lyapunov function. This leads to an importance sampling scheme with bounded relative error.

It should be mentioned in the end that the sampling distribution is determined by (22), and it is not difficult to check that

$$Q(X_{n+1} \in dz | S_n = x) = P(X_1 \in dz | X_1 + Z > -x - a^*).$$

Samples from this conditional distribution are typically generated by suitable acceptance/rejection schemes, where the acceptance probability remains uniformly bounded away from 0. The design of such schemes is based on the tail behavior of the distribution μ. See Blanchet and Glynn (2008) for the case when μ is regularly varying.

See

- Cross-Entropy Method
- Importance Sampling
- Simulation of Stochastic Discrete-Event Systems
- Variance Reduction Techniques in Monte Carlo Methods

References

Asmussen, S. (1985). Conjugate processes and the simulation of ruin probability. *Stochastic Process Application, 20*, 213–229.

Asmussen, S., Binswanger, K., & Hojgaard, B. (2000). Rare event simulation for heavy-tailed distributions. *Bernoulli, 6*, 303–322.

Asmussen, S., & Glynn, P. (2007). *Stochastic simulation: Algorithms and analysis*. New York: Springer.

Asmussen, S., & Rubinstein, R. Y. (1995). Steady state rare event simulation in queueing model and its complexity properties. In J. Dshalalow (Ed.), *Advances in queueing: Theory, methods and open problems volumn 1* (pp. 429–462). Boca Raton, FL: CRC Press.
Bassamboo, A., Juneja, S., & Zeevi, A. (2006). On the efficiency loss of state-independent importance sampling in the presence of heavy-tails. *Operations Research Letter, 34*, 521–531.
Blanchet, J. H., & Glynn, P. (2008). Efficient rare-event simulation for the maximum of heavy-tailed random walks. *The Annals of Applied Probability, 18*, 1351–1378.
Costa, A., Jones, O. D., & Kroese, D. P. (2007). Convergence properties of the cross-entropy method for discrete optimization. *Operations Research Letters, 35*, 573–580.
De Boer, P. T., Kroese, D. P., Mannor, S., & Rubinstein, R. Y. (2005). A tutorial on the cross-entropy method. *Annals of Operations Research, 134*, 19–967.
De Boer, P. T., Kroese, D. P., & Rubinstein, R. Y. (2004). A fast cross-entropy method for estimating buffer overflows in queueing networks. *Management Science, 50*, 883–895.
Dean, T., & Dupuis, P. (2009). Splitting for rare event simulation: A large deviation approach to design and analysis. *Stochastic Process Application, 119*, 562–587.
Dupuis, P., Leder, K., & Wang, H. (2007). Importance sampling for sums of random variables with regularly varying tails. *TOMACS*, Vol. 17, Article 14.
Dupuis, P., Sezer, A., & Wang, H. (2007). Dynamic importance sampling for queueing networks. *Annals Applied Probability, 17*, 1306–1346.
Dupuis, P., & Wang, H. (2004). Importance sampling, large deviations, and differential games. *Stochastics and Stochastic Reports, 76*, 481–508.
Dupuis, P., & Wang, H. (2007). Subsolutions of an Isaacs equation and efficient schemes for importance sampling. *Mathematics of Operations Research, 32*, 1–35.
Glasserman, P., Heidelberger, P., Shahabuddin, P., & Zajic, T. (1999). Multilevel splitting for estimating rare event probabilities. *Operations Research, 47*, 585–600.
Glasserman, P., & Kou, S. (1995). Analysis of an importance sampling estimator for tandem queues. *ACM Transactions Modeling Computing Simulation, 4*, 22–42.
Glasserman, P., & Wang, Y. (1997). Counter examples in importance sampling for large deviations probabilities. *Annals Applied Probability, 7*, 731–746.
Heidelberger, P. (1995). Fast simulation of rare events in queueing and reliability models. *ACM Transactions Modeling Computing Simulation, 4*, 43–85.
Parekh, S., & Walrand, J. (1989). Quick simulation of rare events in networks. *IEEE Transactions Automation Control TAC, 34*, 54–66.
Rubinstein, R. Y. (1997). Optimization of computer simulation models with rare events. *European Journal Operations Reserach, 99*, 89–112.
Rubinstein, R. Y. (1999). The simulated entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability, 2*, 127–190.
Rubinstein, R. Y. (2010). Randomized algorithm with splitting: Why the classic randomized algorithms do not work and how to make them work. *Methodology and Computing in Applied Probability, 12*, 1–41.
Rubinstein, R. Y., & Kroese, D. P. (2004). *The cross-entropy method: A unified approach to combinatorial optimization, Monte Carlo simulation and machine learning*. New York: Springer.
Rubinstein, R. Y., & Kroese, D. P. (2007). *Simulation and the Monte Carlo method*. Hoboken, NJ: Wiley.
Siegmund, D. (1976). Importance sampling in the Monte Carlo study of sequential tests. *Annals of Statistics, 4*, 673–684.

Rate Matrix

The matrix of transition rates (or intensities) in a continuous-time Markov process with discrete state space (sometimes called a continuous-time Markov chain), such as a birth-and-death stochastic process. Each (non-diagonal) entry gives the probability "rate" of making a transition from one state to another, where the time spent in any state is exponentially distributed. By convention, the diagonal entry is selected so that the rows sum to zero; thus it is equal to the negative of the total instantaneous rate out of the state, corresponding to the inverse of the mean (exponentially distributed) holding time in a state. Also called the infinitesimal generator matrix.

See

- ▶ Birth-Death Process
- ▶ Markov Chains
- ▶ Markov Processes

Ray

A ray is a collection of points $(\boldsymbol{x}_0 + \lambda\, \boldsymbol{d})$, where $\boldsymbol{d}$ is a nonzero vector and $\lambda \geq 0$. The vector $\boldsymbol{d}$ is called the direction of the ray and $\boldsymbol{x}_0$ the origin of the ray.

Readiness

- ▶ Availability

Reasoning

A problem-solving process. Two paradigms are logical and analogical reasoning. Logical reasoning includes deductive and inductive. Deductive reasoning is arriving at a conclusion from premises and rules of inference. Inductive reasoning is forming a general conclusion that explains multiple observations. Analogical reasoning uses analogy of a current situation to familiar ones from previous experiences. One paradigm for analogical reasoning is a neural network.

See

- ▶ Artificial Intelligence
- ▶ Expert Systems
- ▶ Neural Networks

Reasoning Knowledge

Knowledge about what circumstances allow particular conclusions to be considered to be valid.

See

- ▶ Artificial Intelligence
- ▶ Expert Systems

Recognition Problem

A computational problem whose answer is "yes" or "no." For example, "given a graph G, is there an Euler tour?"

See

- ▶ Computational Complexity

Recourse Linear Program

- ▶ Stochastic Programming

Reduced Costs

- ▶ Prices

Reduced Gradient Methods

- ▶ Nonlinear Programming
- ▶ Quadratic Programming

Redundancy

Igor Ushakov
Qualcomm Inc., San Diego, CA, USA

Redundancy is an engineering method of improving system and equipment reliability. Mainly, redundancy consists in using extra units (subsystems, modules and/or additional elements) within the system to increase reliability. This kind of redundancy is usually called structural. Redundancy might be called functional when a system may perform the same operation by several different ways. For instance, a communication network may be able to bypass its failed links or switches. Another possibility is time redundancy, where the system has extra time for possible repetition of the same operation after a failure. As an example, one can consider a computer system with restarting in the case of error. This article only considers structural redundancy.

Redundancy may be implemented on the system or unit levels. System-level redundancy means that an entire system would be replaced upon its failure by an identical structure; unit-level redundancy means that an individual unit would be replaced upon failure by a devoted or shared backup element. Let p_k equal the reliability of unit k, n be the number of system units connected in series, and m the degree of redundancy

(i.e., there are $m-1$ replacements standing by). Further assume that the system is operating under hot standby redundancy, wherein items age as they wait to be used. Then it follows for the system-level case that the system reliability is given by $P_1 = 1 - \left(1 - \prod_{k=1^n} p_k\right)^m$ since total failure occurs when the original and all backup series systems fail. For hot standby at the unit level in the case that each unit is backed up by $m-1$ identical others, the reliability of such a series system would be given by $P_2 = \prod_{k=1^n} [1 - (1-p_k)^m]$ since all m of at least one type must fail for the system to fail. For any fixed set of parameters, one always has $P_2 > P_1$. (An analogous result can be formulated for cold standby wherein each backup unit only begins to age upon connection).

Redundancy on the unit level can also be done using a shared backup pool. A good example might be a group of n redundant units assigned to the simultaneous support of a main group of k units, as would be typical in spares provisioning. Illustrations of system-level redundancy, as in the calculation of P_1 above, could include enclosed power packs and printed circuit boards.

When considering redundant systems, one must take into account the monitoring of main and redundant units, the time needed to switch from failed unit to redundant one, and the reliability of switching devices. One special type of redundancy is represented by voting systems. A common use of a voting system is in mission critical software decisions. In these systems, n independent outputs (signals) are compared against each other, such that if k signals coincide, the system is assumed to be operating successfully. If several outputs are possible, then the output that appears most often is taken as correct one (unless there is a tie).

The effect of redundancy can be dramatically increased by the use of renewal (repair or replacement) of failed units. Thus far, the implicit assumption has been that failed units are never repaired but replaced with new ones when needed. However, most failed units are not thrown out but are repaired where appropriate and retained for future use. In this case, system failure can occur only if all units of the redundant group fail during the renewal procedure. So, reliability of a repairable system depends on the duration of repair. For comparison of redundancy with and without repair, consider a duplicated system (one main with a single redundant unit). Let the distributions of time-to-failure and repair time be exponential with parameters λ and μ, respectively. For hot standby, a system without repair has a mean time-to-failure equal to $T_1 = 1.5/\lambda$, whereas for a system with repair, the mean time-to-failure is $T_2 = (1/\lambda)(\mu + 3\lambda)/2\lambda$ (see Kozlov and Ushakov 1970; Ushakov 1994). For instance, if $\lambda = 0.001$ (i.e., the unit's mean time-to-failure = 1,000 hr) and $\mu = 1$ (i.e., mean repair time = 1 hr), then $T_1 = 1,\ 500$ hr and $T_2 \approx 500,000$ hr. For cold standby, $T_1{}^* = 2/\lambda$ and $T_2{}^* = (1\ /\lambda)(\mu + 2\lambda)/\lambda$, respectively, which for the above numerical data gives $T_1 = 2,000$ hr and $T_2 \approx 1,000,000$ hr. These numerical examples show the effectiveness of repair for redundant systems.

Though redundancy improves system reliability, it requires extra resources and money. Cost effectiveness analysis of redundancy is considered as the problem of optimal redundancy.

Consider a system consisting of n independent units. For simplicity, assume that the considered system is series, i.e., failure of any main unit of the system leads to total system failure. To increase the reliability of the system, one uses redundant units in the following way. Let unit i of the system have x_i redundant units and write the probability of successful operation (PSO) of this group as $R_i(x_i)$.

For independent groups of units, the system PSO can be written as

$$R(X) = \prod_{i=1}^{n} R_i(x_i)$$

where $X = (x_1, \ldots, x_n)$. At the same time, introducing x_i redundant units leads to the expenditure of $C(x_i)$ cost units. Usually, one assumes that $C(x_i) = c_i x_i$. In this case, the system total cost equals

$$C(X) = \sum_{i=1}^{n} c_i x_i$$

Then the optimal redundancy problem consists of solving one of the following problems: find the vector solution that delivers either

$$\max_X \left\{ \prod_{i=1}^{n} R_i(x_i) \middle| \sum_{i=1}^{n} c_i x_i \le C^0 \right\} \qquad (1)$$

or

$$\min_{X} \left\{ \sum_{i=1}^{n} c_i x_i \middle| \prod_{i=1}^{n} R_i(x_i) \geq R^0 \right\}. \qquad (2)$$

These problems are conditional discrete optimization problems that can be solved by means of standard tools such as steepest descent, branch and bound, dynamic programming, and integer programming. Notice that both goal functions, $R(X)$ and $C(X)$, are concave. One of the best ways to solve is by use of Kettelle's Algorithm, which represents a convenient computational modification of dynamic programming (Gnedenko and Ushakov 1995).

Cases of multi-constraint versions of problem (1) and its solution are considered in Barlow and Proschan (1981) and Ushakov (1994). The optimal redundancy problem for multi-functional systems, an important extension of problem (2), is solved in Ushakov (1994) as well.

See

- ▶ Cost Analysis
- ▶ Cost-Effectiveness Analysis
- ▶ Reliability of Stochastic Systems

References

Barlow, R. E., & Proschan, F. (1981). *Statistical theory of reliability and life testing: Probability models* (2nd ed.). Silver Spring, MD: To Begin With.

Gnedenko, B., & Ushakov, I. A. (1995). *Probabilistic reliability engineering*. New York: John Wiley.

Kozlov, B. A., & Ushakov, I. A. (1970). *Reliability handbook*. New York: Holt, Rinehart and Winston.

Ushakov, I. A. (Ed.). (1994). *Handbook of reliability engineering*. New York: John Wiley.

Redundant Constraint

An inequality or equation of a mathematical programming problem that does not define part of the solution space. An equivalent problem can be formed by removing redundant constraints.

Regeneration Points

If there exists a time epoch T_1 in a stochastic process such that the continuation of the process beyond T_1 is a probabilistic replica of the process starting at time 0, then the process is said to be regenerative. For such a regenerative process, the existence of subsequent times T_2, T_3, ..., having the same properties follows by repeating the argument, and the set $\{T_1, T_2, T_3, \ldots\}$ are said to be regeneration points of the process.

See

- ▶ Regenerative Simulation
- ▶ Renewal Process

Regenerative Process

- ▶ Regeneration Points
- ▶ Regenerative Simulation

Regenerative Simulation

Peter J. Haas
IBM Almaden Research Center, San Jose, CA, USA

Introduction

Regenerative simulation refers to a collection of statistical techniques for analyzing the output of a discrete-event stochastic simulation whose underlying stochastic process $\{X(t) : t \geq 0\}$ is a regenerative process – here $X(t)$ denotes the (random) state of the simulated system at time t. A regenerative stochastic process has the characteristic property that there exists an infinite sequence of random times, called regeneration points, at which the process probabilistically restarts. The essence of regeneration is that the evolution of the process between any two successive regeneration points is an independent probabilistic replica of the process in any other such cycle. The basic ideas were

initially hinted at by Cox and Smith (1961) and the method itself was pioneered by Crane and Iglehart (1974, 1975) and Fishman (1974). Books and surveys on the regenerative method and regenerative process theory include Asmussen (2003), Asmussen and Glynn (2007), Haas (2002), Henderson and Glynn (2001), Shedler (1993), and Thorisson (2000).

When applicable, regenerative simulation techniques provide a theoretically rigorous and elegant means of studying the limiting or steady-state behavior of $\{X(t) : t \geq 0\}$. Initial applications of regenerative simulation centered on the problem of obtaining point estimates and confidence intervals for system performance measures comprising time-average limits of the form $r(f) = \lim_{t\to\infty}(1/t)\int_0^t f(X(u))\,du$, where f is a real-valued performance function. Under mild conditions, the value of a time-average limit is determined by the expected behavior of the process in a single regenerative cycle – a fact that has important implications for simulation analysis. Under some additional regularity conditions, the time-average limit can also be interpreted as a steady-state or limiting expected value. In general, when estimating steady-state performance measures, one has to worry about the fact that the initial distribution – i.e., the distribution of $X(0)$ – differs from the steady-state distribution, and that the $X(t)$ process is autocorrelated. Thus the effects of initialization bias persist over time and, moreover, estimation methods based on independent and identically distributed (i.i.d.) observations cannot be directly applied. These problems vanish in the regenerative setting. Later work on regenerative simulation has focused on improving the basic estimators, applying the methodology to other system performance measures, extending the theory and methodology to more general classes of stochastic processes, and identifying conditions on the building blocks of a stochastic model under which the regenerative method is applicable.

Regenerative Processes

Regenerative stochastic processes were originally defined by Smith (1955, 1958). The following discussion focuses on processes that evolve in continuous time, but the results carry over to discrete-time process in a straightforward manner – indeed, one can obtain results for a discrete-time process $\{X_n : n \geq 0\}$ by simply applying continuous-time theory to the process $\{X_{\lfloor t \rfloor} : t \geq 0\}$, where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x.

A stochastic process $\{X(t) : t \geq 0\}$ with state space S is a regenerative process in continuous time if there exists an increasing sequence $0 \leq T_0 < T_1 < T_2 < \cdots$ of almost surely (a.s.) finite random times such that the post-T_k process $\{X(T_k + t) : t \geq 0; \tau_{k+l} : l \geq 1\}$

1. Is distributed as the post-T_0 process $\{X(T_0 + t) : t \geq 0; \tau_l : l \geq 1\}$, and
2. Is independent of the pre-T_k process $\{X(t) : 0 \leq t < T_k; \tau_1, \ldots, \tau_k\}$

for $k \geq 1$, where $\tau_k = T_k - T_{k-1}$ for $k \geq 1$. The sequence $\{T_k : k \geq 0\}$ of regeneration points is a (possibly delayed) renewal process that decomposes sample paths of $\{X(t) : t \geq 0\}$ into i.i.d. cycles; the kth cycle is $\{X(t) : T_{k-1} \leq t < T_k\}$. The random variable τ_k defined above is the length of the kth cycle. When $T_0 = 0$ the process $\{X(t) : t \geq 0\}$ is called a nondelayed regenerative process; otherwise, it is called a delayed regenerative process. For a delayed regenerative process $\{X(t) : t \geq 0\}$, the 0th cycle $\{X(t) : 0 \leq t < T_0\}$ need not have the same distribution as the other cycles. Similarly, the length of this cycle – denoted by τ_0 – need not have the same distribution as τ_1, τ_2, and so forth.

Typically, each random point T_k is a stopping time with respect to the $X(t)$ process, in that, for $t \geq 0$, the occurrence or nonoccurrence of the event $\{T_k \leq t\}$ is completely determined by $\{X(u) : 0 \leq u \leq t\}$. In this case, the cycle lengths are determined by the $X(t)$ process, and verifying the regenerative property amounts to showing that for each $k \geq 1$, the distribution of $\{X(t) : t \geq T_k\}$, is distributed as $\{X(t) : t \geq T_0\}$ and is independent of $\{X(t) : 0 \leq t < T_k\}$.

As a simple example of a regenerative process, let $X(t)$ be the number of jobs waiting or in service at time t in a $GI/G/1$ queue, and let T_k denote the kth time at which a job arrives to an empty system, so that $X(T_k) = 1$ for all k. Assume that $T_0 = 0$, so that the simulation starts with such an arrival. Then the random times $\{T_k : k \geq 0\}$ form a sequence of regeneration points for $\{X(t) : t \geq 0\}$. Similarly, if W_n is the waiting time in the queue (exclusive of service time) experienced by the nth job to arrive at the queue, and if

$N(k)$ is the random index of the kth job that arrives and finds the system empty, so that $W_{N(k)} = 0$, then the random indices $\{ N(k) : k \geq 0 \}$ form a sequence of regeneration points for the discrete-time process $\{ W_n : n \geq 0 \}$. An irreducible positive recurrent continuous time Markov chain (CTMC) having a discrete (i.e., finite or countably infinite) state space S is also a regenerative process, with one set of regeneration points comprising the successive times that the chain enters a fixed state $s \in S$, and another set comprising the successive times just after the chain jumps out of a fixed state s. Indeed, such a chain has many sequences of regeneration points that correspond to different choices of s. Discrete-time Markov chains and semi-Markov processes also have regenerative structure, and hence are amenable to regenerative-process theory and methods.

Because a regenerative process has an embedded i.i.d. cycle structure, classical results for i.i.d. random variables can be readily adapted to study the limiting behavior of a regenerative process $\{ X(t) : t \geq 0 \}$ with state space S and regeneration points $\{ T_k : k \geq 0 \}$. E.g., suppose that the process is non-delayed, and for a real-valued function f defined on S, set $Y_k(f) = \int_{T_{k-1}}^{T_k} f(X(u))\, du$ for $k \geq 1$. Also define the function $|f|$ by setting $|f|(s) = |f(s)|$ for $s \in S$. It follows from the definition of a regenerative process that the sequence $\{ (Y_k(f), \tau_k) : k \geq 1 \}$ consists of i.i.d. random pairs. Defining $r(f)$ as before to be a time-average limit, a relatively straightforward application of the strong law of large numbers (SLLN) for i.i.d. random variables establishes the regenerative ratio formula

$$r(f) = \frac{E[Y_1(f)]}{E[\tau_1]}, \tag{1}$$

provided that $E[\tau_1] < \infty$ and $E[Y_1(|f|)] < \infty$. This SLLN for regenerative process illustrates one sense in which the steady-state behavior is determined by the behavior of the process within a cycle.

This result can be extended to steady-state and limiting expected values. Specifically, call a real-valued random variable X aperiodic if there exists no real number d such that $\sum_{n=0}^{\infty} P\{ X = nd \} = 1$. In the literature such random variables are also called non-arithmetic or non-lattice. If, in addition to the foregoing assumptions for the SLLN, it also holds that $\{ X(t) : t \geq 0 \}$ has right-continuous sample paths and that τ_1 is aperiodic, then there exists a random variable X having state space S such that $\lim_{t\to\infty} P\{ X(t) \leq x \} = P\{ X \leq x \}$ for all x at which the function $F(x) = P\{ X \leq x \}$ is continuous; i.e., $X(t)$ converges in distribution to X as $t \to \infty$, denoted by $X(t) \Rightarrow X$. For any function f having a set $D(f)$ of discontinuity points, the continuous mapping theorem ensures that $f(X(t)) \Rightarrow f(X)$ as $t \to \infty$, provided that $P\{ X \in D(f) \} = 0$; moreover, it follows from some basic results in renewal theory – see, e.g., Asmussen (2003) – that $r(f) = E[f(X)]$. That is, $r(f)$ can be interpreted not only as a time-average limit, but also as a steady-state expected value. Finally, if the process $\{ f(X(t)) : t \geq 0 \}$ is uniformly integrable, then a classical result from probability theory shows that $\lim_{t\to\infty} E[f(X(t))] = E[f(X)] = r(f)$, so that $r(f)$ can also be viewed as a limiting expected value. (A stochastic process $\{ Y(t) : t \geq 0 \}$ is uniformly integrable if $\lim_{c\to\infty} \sup_t E[|Y(t)| I(|Y(t)| > c)] = 0$, where $I(A)$ denotes the indicator of event A.)

The Standard Method

The power of the ratio formula (1) lies in the fact that it reduces the problem of estimating steady-state quantities – such as time-average limits or steady-state expected values – to a classical ratio-estimation problem in statistics. The goal of ratio-estimation methods is to obtain point estimates and confidence intervals for quantities of the form $r = E[U]/E[V]$, based on i.i.d. samples $(U_1, V_1), (U_2, V_2), \ldots, (U_n, V_n)$ from the joint distribution of (U, V). In the current setting, $U_i = Y_i(f)$ and $V_i = \tau_i$.

The standard version of the regenerative method simply applies the delta method from statistics to estimate the ratio of interest. Specifically, suppose that a fixed number n of cycles of $\{ X(t) : t \geq 0 \}$ have been observed, so that observations $Y_1(f), Y_2(f), \ldots, Y_n(f)$ and $\tau_1, \tau_2, \ldots, \tau_n$ are available. Set

$$\hat{r}(n) = \frac{\bar{Y}(n)}{\bar{\tau}(n)}, \tag{2}$$

where $\bar{Y}(n) = (1/n) \sum_{k=1}^{n} Y_k(f)$ and

$$\bar{\tau}(n) = \frac{1}{n}\sum_{k=1}^{n} \tau_k. \quad (3)$$

A simple application of the SLLN for i.i.d. random variables shows that $\lim_{n\to\infty}\hat{r}(n) = r(f)$ a.s.; i.e., $\hat{r}(n)$ is strongly consistent for $r(f)$. Similarly, application of the classical central limit theorem (CLT) to the i.i.d. random variables $Z_1, Z_2, \ldots, Z_n$ – where $Z_i = Y_i(f) - r(f)\tau_i$ for $1 \le i \le n$ – leads to the following CLT. Set

$$s^2(n) = (n-1)^{-1}\sum_{i=1}^{n} (Y_i(f) - \hat{r}(n)\tau_i)^2. \quad (4)$$

Here $s^2(n)$ is a consistent estimator of the common variance of the Z_i's. Then, if $E[\tau_1^2] < \infty$, $E\left[Y_1^2(|f|)\right] < \infty$, and $\mathrm{Var}[Z_i] > 0$,

$$\frac{\sqrt{n}(\hat{r}(n) - r(f))}{s(n)/\bar{\tau}(n)} \Rightarrow N(0,1)$$

as $n \to \infty$, where $N(0,1)$ is a standard (mean 0, variance 1) normal random variable. That is, for large n, the distribution of the estimator $\hat{r}(n)$ is approximately normally distributed with mean $r(f)$ and variance $s^2(n)/(n\bar{\tau}^2(n))$. These results immediately lead to the standard regenerative method, which proceeds as follows. Fix $p \in (0,1)$ and let z_p be the unique nonnegative real number such that $P\{-z_p \le N(0,1) \le z_p\} = p$. Then

1. Select a sequence $\{T_k : k \ge 0\}$ of regeneration points for the process $\{X(t) : t \ge 0\}$.
2. Simulate the process $\{X(t) : t \ge 0\}$ and observe a fixed number n of cycles defined by the random times $\{T_k : k \ge 0\}$.
3. Compute the length τ_k of the kth cycle and the quantity $Y_k(f) = \int_{T_{k-1}}^{T_k} f(X(u))\,du$ for $1 \le k \le n$.
4. Form the strongly consistent point estimate $\hat{r}(n) = \bar{Y}(n)/\bar{\tau}(n)$ for $r(f)$.
5. Form the asymptotic $100p\%$ confidence interval

$$\left[\hat{r}(n) - \frac{z_p\, s(n)}{\bar{\tau}(n)\sqrt{n}}, \hat{r}(n) + \frac{z_p\, s(n)}{\bar{\tau}(n)\sqrt{n}}\right]$$

for $r(f)$.

For simplicity, the key limit theorems that underlie the regenerative method have not been presented here in their strongest possible forms; the necessary conditions for these theorems, and hence for applicability of the method, can be weakened in a variety of ways; see, e.g., Asmussen (2003) and Glynn and Iglehart (1993).

The regenerative method also applies directly to a wide range of performance measures other than simple time-average limits and steady state means of the $X(t)$ process. A key observation is that the foregoing ratio estimation methods work for any random pair (U, V) such that U and V are each completely determined by the behavior of the $X(t)$ process within a cycle; the pair $(Y(f), \tau)$ is only one possibility. For example, suppose that $\{X(t) : t \ge 0\}$ is a regenerative process having a discrete state space and piecewise-constant sample paths (as in the case of a continuous-time Markov chain). Let $U_i =$ the number of transitions from a fixed state s_1 to another fixed state s_2 during the ith cycle and $V_i =$ the total number of state transitions during a cycle. Then by applying the regenerative method with U_i and V_i replacing $Y_i(f)$ and τ_i as the input data, one obtains a point estimate and confidence interval for the long-run fraction of state transitions that go from s_1 to s_2. Similarly, performance measures such as discounted rewards and mean time to failure fall within the ratio-estimation framework (Haas 2002, p. 249), as do many performance measures involving delays in discrete-event systems. More generally, the delta method can be applied to permit estimation of smooth nonlinear functions of one or more time-average limits, i.e., performance measures of the form $\alpha = g(r(f_1), r(f_2), \ldots, r(f_l))$, where g is a differentiable nonlinear function and each $r(f_i)$ is a time-average limit of the form discussed previously.

When multiple sequences of regeneration points are available, as in the CTMC setting, a natural question is whether the choice of regeneration-point sequence makes a difference and, if so, which sequence to choose. In general, some quantities in regenerative simulation are sensitive to the particular choice of regeneration points and other quantities are not. For example, the expected length of the confidence interval for $r(f)$ based on a simulation of length t (see below) is asymptotically insensitive to the choice of regeneration points as $t \to \infty$, whereas the variance of the confidence-interval length is extremely sensitive to the choice of regeneration points. See Calvin (1994) for a detailed discussion of this issue.

Variants

Many variants of the basic method have been proposed. Some of these concern the run length of the simulation. One variant, for example, runs the simulation until a fixed (simulated) time t. Point estimates and confidence intervals are computed as above, except that statistics are computed for the random number $n(t)$ of cycles completed by time t. Other variants attempt to automatically determine the number of cycles to simulate so as to achieve a desired precision, either based on an initial pilot sample or by means of a fully sequential estimation procedure.

Various researchers have adapted the regenerative method to estimate steady-state quantities other than means, such as quantiles, central moments, discounted costs, extreme values, and mean time to failure in highly reliable systems. Regenerative simulation can also be used for estimating the gradient of a performance measure that depends on a parameter, such as $dh(\theta)/d\theta$, where $h(\theta) = E_\theta[f(X, \theta)]$; see, e.g., (Asmussen and Glynn 2007, Sec. VII.4) and (Haas 2002, Sec. 6.3.6). Here $X(t) \Rightarrow X$, the function f depends explicitly on θ, and the subscript θ on the expectation operator indicates that the distribution of the regenerative process $\{ X(t) : t \geq 0 \}$ depends on θ.

One serious concern with the standard method is that, even if a sequence of regeneration points exists, the regenerations will not occur frequently enough for the method to be practical. Several algorithm variants therefore try to increase the frequency of regenerations. Andradottir et al. (1994) provide such a scheme for Markov processes. Another example is the almost regenerative method; see Calvin et al. (2006) for a discussion and references.

A number of approaches modify the standard estimator in an attempt to reduce the bias, using, e.g., jackknifing techniques. In the case of simulation until a fixed time t, Meketon and Heidelberger (1982) show that bias can be reduced simply by continuing the simulation until the first regeneration point $T_{n(t)+1}$ after time t. Results of Awad and Glynn (2007), however, indicate that the results of such efforts can be mixed.

Similarly, a number of authors discuss various schemes for reducing the variance of the standard point estimator for the regenerative method. Some of these schemes adapt classical variance reduction methods for i.i.d. random variables – such as control variates, importance sampling, or conditional Monte Carlo – to the regenerative setting. Other approaches exploit particular properties of the system being simulated, e.g., a queueing system or a Markov model of a highly reliable system, to obtain a hybrid simulation-analytic method. Several approaches that are specific to the regenerative setting apply to simulations with multiple sequences of regeneration points, and use resampling or stratified-sampling methods to reduce the variance; see, e.g., Calvin et al. (2006).

Extensions

Some key extensions to the regenerative method, important both theoretically and practically, center around generalizations of regenerative processes that allow some limited dependence between cycles. One setting in which such processes arise is related to estimation of time-average limits for a sequence of delays that is defined in terms of an underlying regenerative process. It is common in such settings that a delay can begin in one cycle and end in another cycle, so that the delay sequence does not inherit the regenerative property. Another setting concerns Markov chains having a general (possibly uncountable) state space. Such chains can be viewed as fundamental processes underlying discrete-event systems. E.g., generalized semi-Markov processes (GSMPs) and stochastic Petri nets (SPNs) – both well-studied frameworks for specification of discrete-event systems – are defined in terms of a general state space Markov chain (GSSMC) that records the physical state of the system along with nonnegative, real-valued "clock readings," i.e., the remaining times until various events are scheduled to occur (Haas 2002; Shedler 1993). In general, sequences of regeneration points for GSSMCs cannot be directly constructed as for Markov chains having a discrete state space, since the probability that a GSSMC hits a given state is often 0.

In the case of delays, it can often be shown (Haas 2002, Sec. 8.2) that the delay sequence of interest is an od(one-dependent)-regenerative process. For such a process, the cycles are identically distributed and one-dependent, in that the ith and $(i + k)$th cycles are independent unless $k = 1$ (Henderson and Glynn 2001; Sigman 1990). The SLLN for regenerative process carries over unchanged to the od-regenerative setting.

The extended regenerative method for obtaining point estimates and confidence intervals looks almost like the standard regenerative method, but the variance constant appearing in the regenerative CLT contains a covariance term that involves adjacent cycles. Alternatively, one can apply a multiple runs method in which cycles are simulated independently and then "glued" together to form a (classically) regenerative process having the same time-average limit as the original process (Glynn 1994; Haas 2002). The standard regenerative method can then be applied to this latter process; if the cycle variables $Z_i = Y_i(f) - r(f)\tau_i$ are positively correlated, then the multiple runs method will have higher asymptotic efficiency than the extended regenerative method.

Analysis of GSSMCs, on the other hand, leads to the notion of an od-equilibrium process. Such a process satisfies all of the properties of an od-regenerative process and, moreover, the cycle lengths are i.i.d.. It follows that a SLLN holds (as with an od-regenerative process) and, in principle, the extended regenerative method and multiple runs methods may be applied to obtain point estimates and confidence intervals for steady-state means. Moreover, the regeneration points form a renewal process and – under appropriate aperiodicity and uniform integrability assumptions – the time-average limit can be interpreted as a steady-state and a limiting expected value. Thus estimation can proceed in essentially the same manner as for a classical regenerative process. The link to GSSMCs rests on a splitting argument, which shows that a GSSMC is an od-equilibrium process if it is Harris recurrent; see, e.g., Meyn and Tweedie (1993), Thorisson (2000), or Henderson and Glynn (2001) for details. Harris recurrence generalizes the notion of recurrence for a Markov chain with a discrete state space, and roughly asserts that a dense enough set of states is visited by the chain infinitely often with probability 1. The key impediment in practice is that actually identifying the od-equilibrium points (i.e., the cycle boundaries) appears to be extremely difficult in general; Henderson and Glynn (2001) explore this issue in detail. On the other hand, Glynn (1994) shows that all well-posed simulations have this type of regenerative structure, and there exist examples of Harris recurrent GSSMCs for which od-equilibrium points can be found.

If the definition of an od-equilibrium process is weakened by dropping the requirement that cycles be one-dependent (so that the cycles are merely stationary but the cycle lengths are i.i.d.), one obtains the class of equilibrium processes (Smith 1955), also called wide-sense regenerative processes (Thorisson 2000). Such processes are also related to the notion of renovating events in queueing networks (Foss and Kalashnikov 1991). Since renewal theory can be applied to such processes, results for steady-state means follow as for ordinary regenerative processes.

Conditions for Applicability

From a practical point of view, it is important to determine whether the regenerative method is applicable to a specific simulation of interest. Typically, some sort of modeling framework is used to specify a simulation model of the system under study, with either application-specific building blocks – such as robot arms and conveyor belts for manufacturing simulations – or general frameworks such as networks of queues, SPNs, event graphs, stochastic automata, and GSMPs. Thus the question is: what are the conditions on the building blocks of a simulation model under which steady state quantities are well defined and the regenerative method is valid?

For systems where the regenerative method is potentially applicable, it is often apparent that the underlying stochastic process probabilistically restarts, e.g., whenever the process is in a specified state and a specified event occurs (such as at an arrival to an empty $GI/G/1$ queue). It can be difficult, however, to verify that such restarts occur infinitely often with probability 1. It is even harder to determine whether the random time between successive regenerations has finite moments. Thus, establishing the validity of the regenerative method often amounts to establishing recurrence properties for the simulation model of interest.

Sufficient recurrence conditions for applicability of the regenerative method have been established in the specific setting of closed networks of queues – see, e.g., Kaspi and Mandelbaum (1992) – and in more general settings such as finite-state GSMPs and stochastic Petri nets. In these latter settings, one set of sufficient conditions for recurrence (Glynn and Haas 2006; Haas 2002) requires roughly that (1) the system be irreducible in that between any two states s and s' there exists a sequence of events that leads from s to s'

with positive probability, and (2) the clock-setting distributions used to stochastically schedule events have finite moments and densities that are positive on a common interval $(0, \varepsilon)$. In this case recurrence is established by first establishing Harris recurrence of the underlying GSSMC using arguments based on stochastic Lyapunov functions. An alternative approach based on geometric trials arguments (Haas 2002; Shedler 1993) avoids the positive density assumption but requires detailed knowledge of system behavior.

Relation to Other Methods

In general, many steady-state simulation methods can be viewed as exploring the behavior of the reward $r(f,t) = \int_0^t f(X(u))\,du$ as t becomes large. Under fairly general conditions, there exist constants $r(f)$ and $\sigma^2(f)$ such that $r(f,t)/t \to r(f)$ a.s. and $\mathrm{Var}[\mathrm{r(f,t)}]/\mathrm{t} \Rightarrow \sigma^2(\mathrm{f})$ as $t \to \infty$. The goal of the simulation is to estimate the time-average limit $r(f)$ and the time-average variance constant (TAVC) $\sigma^2(f)$ – given estimates $\hat{r}(f)$ and $\hat{\sigma}^2(f)$ of these quantities, the reward $r(f,t)$ and its variance can then be easily approximated as $r(f,t) \approx t\hat{r}(f)$ and $\mathrm{Var}[\mathrm{r(f,t)}] \approx \mathrm{t}\hat{\sigma}^2(\mathrm{f})$ when t is large. Examples of steady-state estimation methods include batch-means methods, spectral methods, the autoregressive method, and a variety of methods based on standardized time series, including the method of integrated paths (Calvin 2009).

Based on a simulation until time t, virtually all steady-state simulation methods estimate $r(f)$ by $\bar{r}(f,t) = r(f,t)/t$. The regenerative method essentially uses this estimator also. Indeed, the estimator of $r(f)$ based on simulation until time t is $\hat{r}(n(t))$, where $n(t)$ is the number of regenerative cycles completed by time t and $\hat{r}(n)$ is given by (2). This estimator differs from $\bar{r}(f,t)$ by a (random) remainder term that becomes negligible for large t. The key difference between estimation methods lies in how they estimate the TAVC.

When $\{X(t) : t \geq 0\}$ is a regenerative process, it can be shown that, under appropriate moment and regularity conditions, the TAVC has the representation $\mathrm{Var}[\mathrm{Y_1(f)} - \mathrm{r(f)}\tau_1]/\mathrm{E}[\tau_1]$ – see Henderson and Glynn (2001) – and can thus be consistently estimated by $s^2(n(t))/\bar{\tau}(n(t))$, where the quantities $\bar{\tau}(n)$ and $s^2(n)$ are defined as in (3) and (4). Note that the TAVC and the variance term in the previous CLT for regenerative process differ slightly, because the CLT is expressed in terms of number of cycles rather than simulated time. As discussed in Henderson and Glynn (2001), the mean squared error (MSE) of the regenerative estimate of the TAVC typically decreases at a rate of $O(t^{-1})$, whereas the MSE for all other known methods decreases at a strictly slower rate. On the other hand, these other methods can often be applied in practice to simulations for which regeneration points of sufficient frequency cannot be found.

Concluding Remarks

The standard regenerative method can only be applied to simulations having a sequence of identifiable regeneration points that occur with sufficient frequency. When applicable, however, the method provides a clean and simple solution to the problems of initialization bias and autocorrelation that are fundamental to steady-state analyses. For this reason, the regenerative method was the first mathematically rigorous method proposed for steady-state simulation analysis. Moreover, the point estimates and confidence intervals obtained from the regenerative method often have superior asymptotic properties relative to other output-analysis methods. Virtually all well-posed steady-state simulations have a form of regenerative structure, namely, the od-equilibrium property, making extensions of the standard regenerative method to this setting an important research area.

See

- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Variance Reduction Techniques in Monte Carlo Methods

References

Andradottir, S., Calvin, J. M., & Glynn, P. W. (1994). Increasing the frequency of regeneration for Markov processes. In *Proceedings of the 1994 Winter Simulation Conference*, Orlando FL (pp. 320–323).

Asmussen, S. (2003). *Applied probability and queues* (2nd ed.). New York: Springer.
Asmussen, S., & Glynn, P. W. (2007). *Stochastic simulation: Algorithms and analysis*. New York: Springer.
Awad, H., & Glynn, P. W. (2007). On the theoretical comparison of low-bias steady-state estimators. *ACM Transactions on Modeling and Computer Simulation*, (*1*), *4*.
Calvin, J. (1994). Return state independent quantities in regenerative simulation. *Operation Research, 42*, 531–542.
Calvin, J. M. (2009). Simulation output analysis using integrated paths II: Low bias estimators. *ACM Transactions on Modeling and Computer Simulation*, *9*(*3*).
Calvin, J. M., Glynn, P. W., & Nakayama, M. K. (2006). The semi-regenerative method of simulation output analysis. *ACM Transactions on Modeling and Computer Simulation, 16*(3), 280–315.
Cox, D. R., & Smith, W. L. (1961). *Queues*. London: Methuen.
Crane, M. A., & Iglehart, D. L. (1974). Simulating stable stochastic systems, I: General multiserver queues. *Journal of the Association for Computing Machinery, 21*(1), 103–113.
Crane, M. A., & Iglehart, D. L. (1975). Simulating stable stochastic systems: III. regenerative processes and discrete event simulation. *Operations Research, 23*, 33–45.
Fishman, G. S. (1974). Estimation in multiserver queueing simulations. *Operations Research, 22*, 72–78.
Foss, S. G., & Kalashnikov, V. (1991). Regeneration and renovation in queues. *Queueing Systems: Theory and Applications., 8*, 211–224.
Glynn, P. W. (1994). Some topics in regenerative steady-state simulation. *Acta Applicandae Mathematicae, 34*, 225–236.
Glynn, P. W., & Haas, P. J. (2006). Laws of large numbers and functional central limit theorems for generalized semi-Markov processes. *Communications in statistics. Stochastic models, 22*, 201–231.
Glynn, P. W., & Iglehart, D. L. (1993). Conditions for the applicability of the regenerative method. *Management Science, 39*(9), 1108–1111.
Haas, P. J. (2002). *Stochastic Petri nets: Modelling, stability, simulation*. New York: Springer.
Henderson, S. G., & Glynn, P. (2001). Regenerative steady-state simulation of discrete-event stochastic systems. *ACM Transactions on Modeling and Computer Simulation, 11*, 313–345.
Kaspi, H., & Mandelbaum, A. (1992). Regenerative closed queueing networks. *Stochastics and Stochastics Reports, 39*, 239–258.
Meketon, M. S., & Heidelberger, P. (1982). A renewal theoretic approach to bias reduction in regenerative simulations. *Management Science, 28*, 173–181.
Meyn, S. P., & Tweedie, R. L. (1993). *Markov chains and stochastic stability*. London: Springer.
Shedler, G. S. (1993). *Regenerative stochastic simulation*. New York: Academic.
Sigman, K. (1990). One-dependent regenerative processes and queues in continuous time. *Mathematics of Operations Research, 15*, 175–189.
Smith, W. L. (1955). Regenerative stochastic processes. *Proceedings of the Royal Society London: Series A, 232*, 6–31.
Smith, W. L. (1958). Renewal theory and its ramifications. *Journal of the Royal Statistical Society. Series B, 20*, 243–302.
Thorisson, H. (2000). *Coupling, stationarity, and regeneration*. New York: Springer.

Regression Analysis

Irwin Greenberg
George Mason University, Fairfax, VA, USA

Introduction

In almost all fields of study, the researcher is frequently faced with the problem of trying to describe the relation between a response variable and a set of one or more input variables. Given data on input (predictor, independent) variables labeled x_1, x_2,. . ., x_p and the associated response (output, dependent) variable y, the objective is to determine an equation relating output to input. The reasons for developing such an equation include the following:

1. To predict the response from a given set of inputs.
2. To determine the effect of an input on the response.
3. To confirm, refute, or suggest theoretical or empirical relations.

To illustrate, the simplest situation is that of a single input for which a linear relation is assumed. Thus, if the relation is exact, it is given for appropriate values of β_0 and β_1 by

$$y = \beta_0 + \beta_1 x. \tag{1}$$

The determination of β_0 and β_1 in this case is easy, requiring only two distinct pairs of observations (x_1, y_1) and (x_2, y_2).

In general, the problem is more complex in that the response is not given exactly by (1). This may be true because, although the relation is theoretically given by (1), the observations are not measured without error. Alternatively, there may be no theoretical justification for an exact linear relation but it is used as an approximation.

A model, commonly used in both cases, is

$$y = \beta_0 + \beta_1 x + e. \tag{2}$$

Here e denotes the measurement error or other random fluctuations in y which cause the response to depart from (1); it is assumed that the input variables are either specified by the user or measured without error.

The appropriate analysis of (2) is dictated by the assumptions made on the distribution of errors. Typically, it is assumed that the errors have mean zero and variance σ^2 and that the errors associated with distinct observations are uncorrelated. That is, if a very large number of pairs (x_i, y_i) were observed for a situation modeled by (2), then (a) the errors

$$e_i = y_i - \beta_0 - \beta_1 x_i \tag{3}$$

would average to zero; (b) the error associated with one observation would in no way influence any other error; and (c) the mean of the squares of the errors would be σ^2.

Based on n pairs of observations (x_i, y_i), $i = 1,\ldots,n$, the objective of the analyst is to estimate β_0, β_1, and σ^2 and to make inferences about these parameters. In addition, it may be desirable to indicate the precision of a prediction obtained for a given input when the estimates b_0 and b_1 of β_0 and β_1 are used in (1). These inferences require further specification of the distribution of the errors. The classical results are developed assuming a Gaussian (or normal) distribution.

A generalization of this simple model is the multiple linear regression model

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + e. \tag{4}$$

Here the assumption on the errors is the same as given above and the analysis is to be based on $n(p+1)$- tuples $(x_{1i}, x_{2i},\ldots,x_{pi}, y_i)$, $i = 1,\ldots, n$. The sense in which (4) is a linear model must be emphasized. As written, the average response in (2) is a linear function of x and in (4) is a linear (planar) function of $x_1,\ldots,x_p$, but this is not the essential linearity. The critical feature is that the average response is a linear function of the coefficients β_0, β_1, $\ldots$, β_p The variables indicated by y and x_i, $i = 1, \ldots, p$ may represent functions of the variables which are actually observed as long as these functions do not depend on unknown parameters. For example, the model

$$\log\ z = \beta_0 + \beta_1/w + e \tag{5}$$

does not represent z as a linear function of w, but by letting $y = \log z$ and $x = 1/w$ this model is seen to be equivalent to (2). Similarly, the polynomial model

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + e \tag{6}$$

is a special case of (4) with $x_1 = x$ and $x_2 = x^2$.

Classical Least-Squares Analysis

The estimation of the unknown parameters in the general linear regression model is most frequently achieved by the method of least squares. Given n observations (or cases) $(x_{1i}, x_{2i},\ldots,x_{pi}, y_i)$, $i = 1, \ldots, n$, let the ith residual be

$$e_i = y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}.$$

The method of least squares determines values, b_j, as estimates of β_j so as to minimize the sum of squared residuals. The estimated regression function (predicted value) for the ith set of inputs, $\hat{y}_i$, and the estimated residual r_i are given by

$$\begin{aligned} \hat{y}_i &= b_0 + \sum_{j=1}^{p} b_j x_{ij} \\ r_i &= y_i - \hat{y}_i \end{aligned} \tag{7}$$

There are essentially two major advantages of this method. The first is computational, since the method only requires the solution of a system of linear equations. The second is statistical, in that the estimates possess desirable small sample properties. In particular, the b_j are unbiased estimates of the β_j which have minimum variance in the class of estimators which are unbiased. Further, the assumption of normality allows for simple inferences on the β_j The estimate of σ^2 is also unbiased and minimum variance.

Note that these properties refer to the $(x_1, x_2,\ldots, x_p, y)$ relationship and not to the underlying variables. For example, the b_0 and b_1 values derived to estimate the β_0 and β_1 of (5) provide unbiased estimators of $\log z$, not of z, and minimize the sum of the squares of the deviations from the linear plot of $\log z$ vs. $1/w$, not from the curvilinear plot of z vs. w.

With the advent of high-speed computing, the computational advantage is less compelling than in the past. This has encouraged a study of alternatives to least squares, some of which will be described subsequently.

Departures from the Classical Assumptions

The standard analysis assumes that the model is correct and that the data are good. In practice, this is rarely the case and it is essential that the violations be detected and evaluated. Some of the main problems are the following:

1. Incorrect functional form for the regression function. Additional variables and/or different functions of the variables may be required.
2. Violations of the assumptions of independence, constant variance, and normality of errors.
3. Outliers and extreme points. The former are observations in which the response is abnormally large or small and the latter are cases in which the inputs are different from the rest of the data.
4. Multicollinearity among the input variables, that is, nearly exact linear relations among subsets of the input variables. This includes the case where one of the inputs is nearly constant.

One or more of these problems may completely invalidate the analysis. Several additional indicators have been proposed to address these possibilities. Unfortunately, there are no guaranteed solutions to any of the problems cited. The following remedies are typical but must be used with caution.

1. Nonuniform residual plots may suggest nonlinear functions. Individual points that are outstanding may suggest other variables that could be included, especially categorical variables defining subgroups of cases.
2. The most common cause of variance inhomogeneity is that the variance is proportional to one of the inputs. Division of the equation by this variable, or some power of it, will help. Normality may be achieved by transformations.
3. Outliers and extreme points may be deleted from the analysis but care must be taken, as these may be valid, informative observations. Alternatively, one of the robust procedures might be used.
4. An eigenvector analysis may identify the multicollinearity, but the action to be taken depends on the cause. If the linear relation is inherent in the system being modeled and the relation is strong, it may be appropriate to eliminate one of the variables in the relation. If the apparent linear relation is due to the peculiarities of the particular sample, then, if possible, additional data should be taken which are more uniformly spread over the sample space. Alternatively, one might simulate this by using ridge regression or a related method.

Alternatives to Classical Least Squares

Since least-squares analysis is vulnerable to departures from the basic assumptions, several alternatives have been suggested.

One of best known of these alternatives is robust regression, where the basic idea is that observations with large residuals are given less weight and hence become less influential.

When multicollinearities are present, least squares estimates of the coefficients may be abnormally large or even have the wrong sign. Ridge regression is the method that effectively adjoins fictitious data.

One of the oldest modifications of least squares is that of eliminating variables. This has been a confusing and controversial topic primarily because it has often applied indiscriminately to data that have not been subjected to proper diagnostics. Variable elimination only should be applied after the data have been examined for extremes, outliers, and multicollinearities, and appropriate action has been taken. Variables that are then not contributing to the description of the response may be eliminated.

An alternative to eliminating variables, implemented in most of the popular statistical software packages, is stepwise regression. The most significant of the x_i are determined and the parameters of (1) are estimated. New x variables are added to the equation until the resulting decrease in the portion of the variance of errors not explained by the regression becomes statistically insignificant.

See

► Exponential Smoothing
► Time Series Analysis

References

Belsley, D. A., Kuh, E., & Welsch, R. E. (1980). *Regression diagnostics*. New York: Wiley.
Daniel, C., & Woods, F. S. (1971). *Fitting equations to data*. New York: Wiley.
Draper, N. R., & Smith, H. (1966). *Applied regression analysis*. New York: Wiley.
Gunst, R. F., & Mason, R. L. (1980). *Regression analysis and its applications*. New York: Marcel Dekker.
Neter, J., & Wasserman, W. (1974). *Applied linear statistical models*. Homewood, IL: Richard D. Irwin.

Reinforcement Learning

The name used by the artificial intelligence community for approximate dynamic programming.

See

► Approximate Dynamic Programming

Relational Database

► Information Systems and Database Design in OR/MS

Relative Costs

► Prices

Relaxed Problem

The term given to a constrained optimization problem in which some of the constraints have been weakened or relaxed. In particular, it is applied to an integer-programming problem in which the variables are no longer restricted to be integer. The objective function of the relaxed problem serves as a bound for the original problem.

See

► Integer-Programming Problem

Reliability

The ability of a component or system to be operable when called upon to do its intended job. Reliability is most often quantified as the probability that the component or system has not failed (is alive) at a particular time: $R(t) = \Pr\{\text{ lifetime} > t\} = 1 - F(t)$, where F is the cumulative distribution function of the lifetime of the component or system. This reliability function is often also called the survival function.

See

► Failure-Rate Function
► Reliability Function
► Reliability of Stochastic Systems

Reliability Function

The reliability at time t, $R(t)$, is defined as $\Pr\{\text{lifetime} > t\} = 1 - F(t)$, where F is the cumulative distribution function of the lifetimes. Also called the survival function.

See

► Failure-Rate Function
► Reliability
► Reliability of Stochastic Systems

Reliability of Stochastic Systems

Donald Gross
George Mason University, Fairfax, VA, USA

Introduction

Quality is a ubiquitous concept, from newer developments such as quality circles and total quality management to old standbys such as quality control and quality assurance. Intricately related to quality,

in fact, a necessary ingredient, is reliability, loosely defined as the probability that a system, subject to random failures, will perform properly over some time span of interest. This definition shall be made more precise in the following. One might be able to have reliability without quality, but one can never have quality without reliability.

The major issue here is a consideration of the probability structure of systems made up of individual components, each with a known lifetime density, say, $f_i(t)$. The two basic combinations of system design are the series and parallel systems, with more complex structures built up from these. By series it is meant the arrangement whereby any single item's failure leads to total system failure. For the parallel case, all component devices must fail for total system failure.

There are a number of important variations on the parallel theme. They differ in the manner in which the set of devices are permitted to operate simultaneously and if not, what form of switch is necessary to call upon any alternative. When all are going together, such a system is called parallel redundant. When items are not in use but waiting to be switched to use if needed, and the items not in use do not deteriorate with age, the structure is said to be a cold standby system. The hybrid combination which finds the standby elements possibly aging at a slower pace than if they were in use is called a warm standby system. If items not in use age at the same rate as they do when in use, then the system is often referred to as a hot standby system and is equivalent to a parallel redundant system as long as the switching mechanism which brings the standby item on line when the operating item fails is itself 100% reliable (zero probability of failing).

Lifetime Probabilities

The direct application of the basic laws of probability permits the easy derivation of the lifetime probabilities associated with each of these fundamental structures. The cumulative distribution function (CDF) for the simple series system without maintenance is

$$F(t) = 1 - \prod_{i=1}^{n} [1 - F_i(t)]$$

where $F_i(t)$ is the lifetime CDF of the ith component. This result is made slightly more compact by defining a reliability function $R(t)$ as the complementary CDF, $1 - F(t)$, namely, the probability of a lifetime longer than t. Then the system reliability may be written in terms of the component device reliabilities as

$$R(t) = \prod_{i=1}^{n} R_i(t).$$

In the special case where each component's life follows the exponential distribution with parameter λ_i,

$$R(t) = \exp\left(\sum_{i=1}^{n} \lambda_i t\right).$$

For the parallel redundant (or hot standby with a 100% reliable switch) case, the system lifetime CDF is

$$F(t) = \prod_{i=1}^{n} F_i(t)$$

and thus, its reliability function is

$$R(t) = 1 - \prod_{i=1}^{n} F_i(t).$$

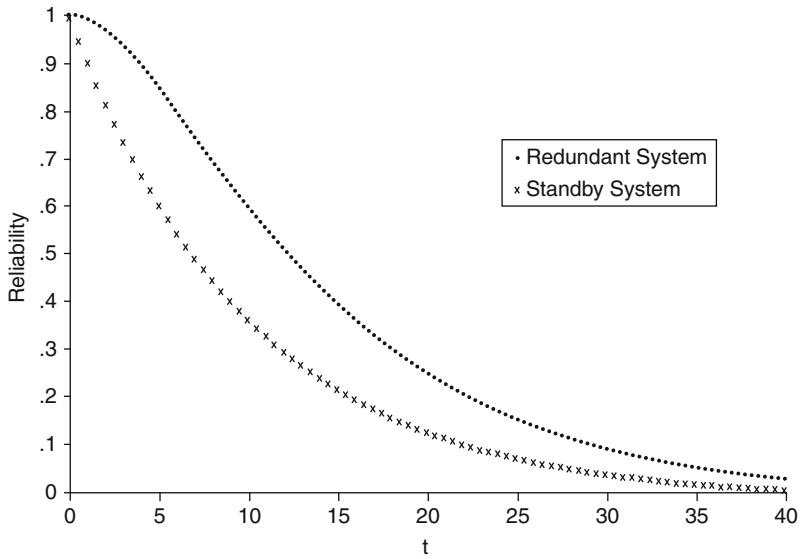
In the event that the devices are independent and identically distributed exponential distributions with parameter λ, then

$$F(t) = [1 - \exp(-\lambda t)]^n.$$

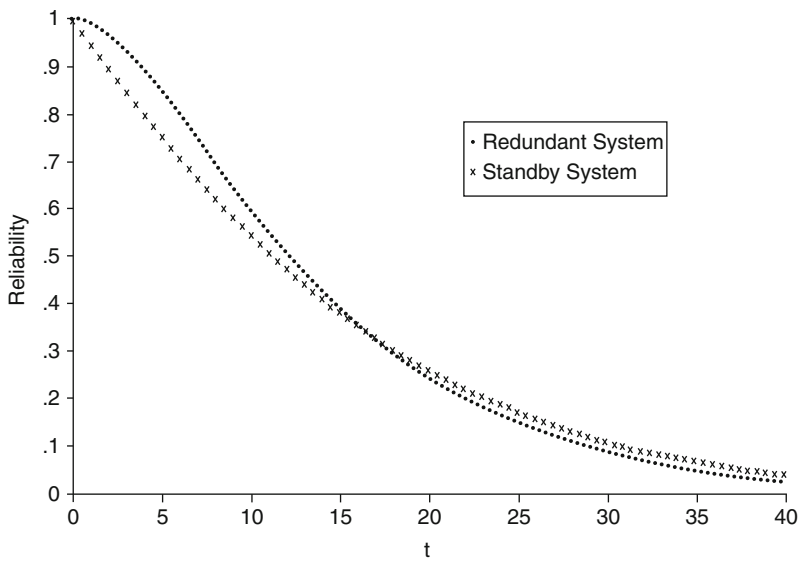
In the special case where $n = 2$, the exponential system has reliability function

$$R(t) = \exp(-\lambda t)[2 - \exp(-\lambda t)].$$

It is interesting then to compare this result to that for the two-unit exponential parallel cold standby 100% reliable switch system. The latter can be derived as the sum of two probabilities: the probability that the original component lives past time t plus the probability that the original component fails in some time v, $0 \le v \le t$, and the standby component lives longer than $t - v$, integrated over v from 0 to t.

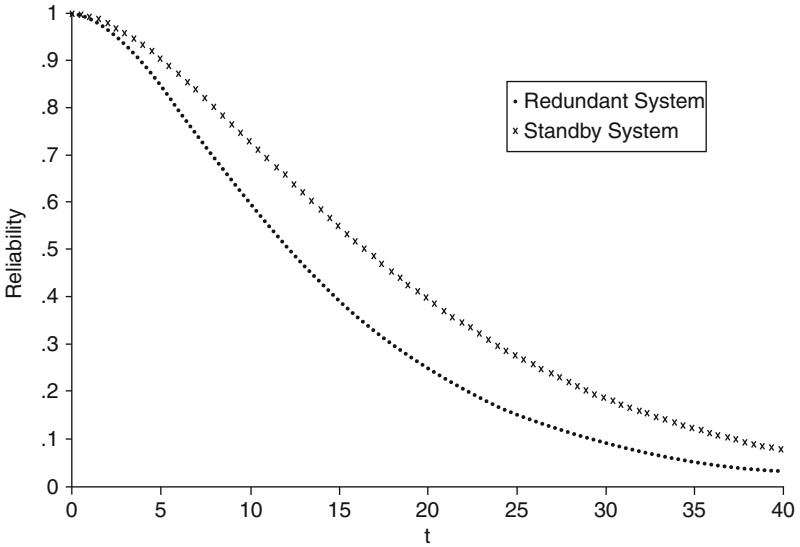


Reliability of Stochastic Systems, Fig. 1 Reliability Function for Redundant & Standby Parallel Systems (switch probability = 0)



Reliability of Stochastic Systems, Fig. 2 Reliability Function for Redundant & Standby Parallel Systems (switch probability = 0.5)

This cold-standby reliability, which, for the case of identical exponential components, turns out to be $R(t) = (1 + \lambda t)\exp(-\lambda t)$, is greater than that of the redundant structure for all values of λ. This result can be easily extended to general n. One can also build in a probability of switch failure for the standby case and observe the effects of an unreliable switch on the relative merits of standby versus redundant systems. If p is the probability that the switch will work, the reliability is adjusted to $R(t) = (1 + p\lambda t)\exp(-\lambda t)$. Figures 1, 2, and 3 show plots of $R(t)$ versus t for $p = 0.5$ and 1, respectively. When the probability of the switch working is zero, the graph shows that the parallel redundant case is superior. When the probability of the switch working is 1, the cold standby case is superior, as seen above. However, when the switch probability is between 0 and 1, as it is for the 0.5 case, there is a point of time where the reliabilities of the parallel redundant and the cold standby cases cross.



Reliability of Stochastic Systems, Fig. 3 Reliability Function for Redundant & Standby Parallel Systems (switch probability = 1)

N-Out-of-K Systems

The parallel redundant system is generalized to define failure if more than $n - k$ of the n components are not working (less than k working). So $R(t) = \Pr\{\text{at least } k \text{ of } n \text{ are up}\}$. Let $p = \Pr\{\text{component works until } t = \exp(-\lambda t)\}$. Then, from the binomial probability law,

$$R(t) = \sum_{i=k}^{n} \binom{n}{i} \exp(-\lambda t i)[1 - \exp(-\lambda t)]^{n-i}.$$

Maintained Systems

In the typical reliability application, failed units are often put into repair. As a first illustration of such a maintained system, consider a single device with time-to-failure exponential, mean $1/\lambda$, and the time to repair exponential, mean $1/\mu$. It then turns out in this simple single-component system that the probabilities that the system is operating or is down at time t, respectively, are

$$p_0(t) = \frac{\mu + \lambda \exp[-(\lambda + \mu)t]}{\lambda + \mu}$$

and

$$p_1(t) = \frac{\lambda\{1 - \exp[-(\lambda + \mu)t]\}}{\lambda + \mu}.$$

The quantity $p_0(t)$ is often called the system availability (written as $A(t)$ since it is the probability that the system is available at time t). The long-run average availability is computed from $A(t)$ as $t \to \infty$ to be $A = \mu/(\lambda + \mu)$.

Next, consider a two-item series system with identical exponential failure distributions and one exponential repair facility. The time-dependent probabilities are found as the solution to a 3×3 system of difference/differential equations; here, only the limiting probabilities are treated. The steady-state availability is the limiting fraction of time no devices are down and is given by $A = \mu^2/(2\lambda^2 + 2\lambda\mu + \mu^2)$. The limiting probabilities that one and two units are down are respectively given as

$$p_1 = \frac{2\lambda\mu}{2\lambda^2 + 2\lambda\mu + \mu^2}$$
$$p_2 = 1 - p_1 - A$$

The final maintained system discussed is the simple two-item exponential parallel redundant structure with repair. Here

$$A = \frac{\mu^2 + 2\lambda\mu}{2\mu^2 + 2\lambda\mu + 2\lambda^2}$$

which is clearly larger than that just presented for the series system.

The Structure Function

For more complicated systems of components, what is commonly called the structure function is a convenient vehicle for characterizing the reliability. First, for any component i, define a binary indicator random variable X_i as 1 if the device is operating and 0 otherwise. The structure function of a system of n components is then written as $\phi(X_1,\ldots, X_n)$ and will likewise be 1 if the system is operating and 0 otherwise. If all the components are queried at time t, then the system reliability $R(t)$ at that point is the probability that $\phi = 1$.

For a full series system then, all X_i must be 1 for ϕ to be 1, so that

$$\phi(X_1,\ldots,X_n) = \prod_{i=1}^{n} X_i.$$

In the pure parallel case, $\phi = 1$ if any $X_i = 1$, so that

$$\phi(X_1,\ldots,X_n) = 1 - \prod_{i=1}^{n} (1 - X_i).$$

The beauty of the structure function is its ability to model the most complex of systems in a fairly natural Boolean way. For example, consider a structure of five components with 1 and 4 in series, parallel with 2 and 5 in series, and also allowing the combination 1, 3, 5 for operation (together called a bridge structure). The system operates as long as at least one of these three combinations is up. Thus

$$\phi(X_1,\ldots,X_n) = 1 - (1 - X_1X_4)(1 - X_2X_5)(1 - X_1X_3X_5).$$

As a general rule, attention is limited to structures that make sense. A system is coherent if

(a) its structure function ϕ is increasing in each argument (that is, ϕ improves as X goes to 1 from 0); and
(b) each component is relevant (that is, its reliability affects system performance).

A fairly complete theory has been developed and is given in Barlow and Proschan (1975).

There is quite an extensive literature on systems reliability and related problems. Barlow and Proschan (1975) and Barlow (1998) are key references, and further material of special importance on systems problems may be found in Kaufmann et al. (1977). Introductory material on systems reliability may also be found in Chapter 12 of Hillier and Lieberman (1990) and Chapter 9 in Ross (2010), with a more advanced treatment in Chapter 9 of Crowder et al. (1991).

See

- ▶ Distribution Selection for Stochastic Modeling
- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Quality Control
- ▶ Queueing Theory
- ▶ Redundancy
- ▶ Total Quality Management

References

Barlow, R. E. (1998). *Engineering reliability. ASASIAM series on statistics and applied probability*. Philadelphia, PA: SIAM.

Barlow, R. E., & Proschan, F. (1975). *Statistical theory of reliability and life testing*. New York: Holt, Rinehart and Winston.

Crowder, M. J., Kimber, A. C., Smith, R. L., & Sweeting, T. J. (1991). *Statistical analysis of reliability data*. London: Chapman and Hall.

Hillier, F. S., & Lieberman, G. J. (1990). *Introduction to stochastic models in operations research*. New York: McGraw Hill.

Kaufmann, A., Grouchko, D., & Cruon, R. (1977). *Mathematical models for the study of the reliability of systems*. New York: Academic Press.

Ross, S. M. (2010). *Introduction to probability models* (10th ed.). New York: Academic Press.

Reneging

In queueing, when customers get impatient and leave their queue before their service is begun.

See

- ▶ Queueing Theory

Renewal Equation

▶ Renewal Process

Renewal Process

Igor Ushakov
Qualcomm Inc., San Diego, CA, USA

A renewal process is a stochastic point process $\{N(t),\ t \geq 0\}$, where $N(t)$ = number of occurrences by time t, which describes the appearance of a sequence of instant random events where the times between occurrences (e.g., called interarrival times in queueing theory) are a sequence of independent and identically distributed (i.i.d.) non-negative random variables. It is common to write the interoccurrence distribution function as $F(t)$ and its density (if it exists) as $f(t)$, with expected value $1/\mu$. The Poisson process represents a particularly important renewal process in which the intervals between occurrences are exponentially distributed (Cox 1960; Cox and Isham 1980; Feller 1966; Ross 1996; Smith 1955; Wolff 1989).

The renewal equation for the process expectation (or renewal function) $H(t) = \boldsymbol{E}[N(t)]$, plays a fundamental role in all renewal problems:

$$H(t) = F(t) + \int_0^t H(t-x)\, dF(x).$$

The derivative of $H(t)$, $h(t) = dH/dt$, is often called the intensity function and has a simple interpretation: $h(t)dt$ is the approximate probability of an occurrence within the time interval $[t, t + dt]$.

One can write an equation for the intensity function similar to the one above:

$$h(t) = f(t) + \int_0^t h(t-x)\, dF(x).$$

With t increasing, it follows that

$$\lim_{t\to\infty} \frac{H(t)}{t} = \frac{1}{\mu}$$

In a physical sense, this means that, over a large interval of size t, the mean number of events is inversely proportional to the expected interarrival time. This is usually referred to as the elementary renewal theorem (Ross 1996; Wolff 1989).

Very close to the previous statement is the following. If the renewal process is formed by continuous random variables, then

$$\lim_{t\to\infty} h(t) = \frac{1}{\mu}$$

This reflects the fact that with increasing t, the renewal process becomes stationary and its intensity becomes independent of the current time.

A further generalization comes from Blackwell's Theorem, which states for continuous interrenewal-time random variables and an arbitrary interval width $\tau \geq 0$ (Feller 1966; Ross 1996; Wolff 1989):

$$\lim_{t\to\infty} [H(\tau + t) - H(t)] = \frac{\tau}{\mu}$$

The next important result is contained in Smith's Theorem (1955), also known as the key renewal theorem (Ross 1996; Wolff 1989). If the renewal times random variables are continuous and $V(t)$ is a monotone non-increasing function, integrable on $(0, \infty)$, then

$$\lim_{t\to\infty} \int_0^t V(t-x)\, dH(t) = \frac{1}{\mu} \int_0^\infty V(t)\, dt.$$

The actual choice of the function $V(t)$ depends on the particular problem of concern.

Another special point process can be formed by two independent subsequences of random variables that alternate, where a realization of such a process has the sequence $X_1, Y_1, X_2, Y_2, \ldots$. Such a process is called an alternating renewal process when the X and Y subsequences are themselves ordinary renewal processes. An example of such a process is the modeling of equipment failure and repair over time.

See

- ▶ Point Stochastic Processes
- ▶ Poisson Process
- ▶ Queueing Theory
- ▶ Stochastic Model

References

Cox, D. R. (1960). *Renewal theory*. New York: Methuen.
Cox, D. R., & Isham, V. (1980). *Point processes*. New York: Chapman and Hall.
Feller, W. (1966). *Introduction to probability theory and its applications* (Vol. II). New York: Wiley.
Ross, S. M. (1996). *Stochastic processes* (2nd ed.). New York: Wiley.
Smith, W. L. (1955). Regenerative stochastic processes. *Proceedings of the Royal Society, Series A, 232*, 6–31.
Wolff, R. W. (1989). *Stochastic modeling and the theory of queues*. Englewood Cliffs, NJ: Prentice-Hall.

Representation Theorem for Polyhedral Set

Given a nonempty polyhedral set S, then a point X is in S if and only if X can be expressed as a convex combination of the set's extreme points plus a non-negative combination of its extreme directions.

Research Analysis Corporation (RAC)

- ▶ Operations Research Office and Research Analysis Corporation

Research and Development

John C. Papageorgiou
Wellesley, MA, USA

Introduction

Products and services have a finite life cycle, and the speed with which they go through their life cycle stages has been continuously increasing. Through the functions in an organization called research and development (R&D), new products and services are developed, existing ones are improved, and the respective transformation processes are improved to increase efficiency and minimize cost.

The worldwide distribution of R&D performance is highly concentrated in several industrialized nations, but large emerging economies like China, India and Brazil have been added to the countries engaged in R&D performance. Part of the R&D activity is aimed at pure research, i.e., research meant for pursuit of knowledge. This research takes place mainly in research laboratories of universities, research centers, government agencies, and major corporations. The other part of R&D activity is made on applied research, in which existing knowledge is used to design new products, services and processes, as well as on development, i.e., the conversion of the results of applied research into the actual transformation systems that will produce the new products and services.

In industry, ideas for R&D projects originate primarily in a firm's R&D department. However, other departments such as marketing, production and engineering are frequent contributors, as is top management. In some cases, suppliers, clients/customers, and government departments are sources of ideas. R&D project management is often difficult, due to the high degree of uncertainty involved, and OR/MS has developed several approaches to help R&D managers. OR/MS has addressed mainly two major problems in R&D management: (1) project evaluation, selection and resource allocation; and (2) project planning and control.

R&D Project Selection

The R&D project evaluation, selection and resource allocation problem deals with the evaluation of candidate R&D projects, and the selection of a subset of such projects to which available R&D resources (manpower, funds, equipment and facilities) will be allocated. Because of the investment commitment, the uncertainties involved, and the impact of the decisions upon the future of the organization, project selection is a very important and difficult problem. As a result, hundreds of papers have been published discussing the problem and suggesting various approaches for its

solution. Early approaches to the solution of the project selection problem were reviewed in several literature survey papers (Augood 1973; Baker and Pound 1964; Baker 1974; Baker and Freeland 1975; Liberatore and Titus 1983; Souder 1972; Souder and Mandakovic 1986). Since then, researchers have proposed several other approaches that introduce improvements over the earlier approaches. Only a small number of all the approaches that have been proposed can be discussed here, indicating the evolution of this area of OR/MS.

Before World War II, the project selection problem was non-existent. Companies were relatively small and competition was limited, which resulted in a limited need to develop new products. There was no distinction between "research" and "development," and the relevant function was not viewed as important. Usually, the chief technical officer would come up with a production related project and pursue it through its implementation. It was after World War II that the business environment changed, increased competition resulted in increased demand for new products/services and improved processes, and R&D project selection became a problem.

Project evaluation and selection methods started appearing in the mid-1950s. The first methods used, called checklists or profile charts, are based on a checklist of criteria. The checklist consists of factors considered to be important to the success or failure of the project, and is used as the basis on which each project is subjectively rated by one or more individuals. The checklist may include both economic and non-economic factors, such as social impacts and environmental concerns. The degree of favorableness of each criterion is checked for each project, with the objective to derive an overall pattern for each project and determine its degree of favorableness.

Because this method does not differentiate among the importance of different criteria and is based on qualitative judgment, scoring models were later developed. These models use weights assigned to both the different criteria and the degree of favorableness of each of them for each project. As a result, a weighted score is computed for each project. For the project scores to be comparable, similar criteria have to be used for all the projects. Different methods have been proposed for deriving the set of weights representative of the preference function of the particular decision maker, such as having the decision maker rank order the criteria or make comparisons of different pairs of projects.

Since these methods give dimensionless results, benefit-cost ratio approaches were developed. The different costs and benefits associated with a project, including non-economic costs and benefits, are expressed in terms of a common measure and their present value computed and expressed as a ratio. Risk factors can also be included in terms of probabilities of research, development and market success. For a project to be considered, its benefit-cost ratio should be greater than one.

The above methods, usually called classical methods, have been used extensively in R&D project evaluation and selection due to their simplicity and ease of use. They can prove useful in preliminary project analysis and screening. However, they cannot solve the project selection problem because they ignore several key aspects. For example, that projects are selected in sequences rather than individually, where the outcome of one affects that of another. Other ignored aspects include the dynamic nature of the R&D environment in terms of project funding at different levels, in which case the value and preferability of a project is a function of its funding level; dynamic resource constraints; and that the set of candidate projects continuously changes over time.

Decision trees were then introduced to deal with sequences of interrelated projects. A decision tree consists of decision nodes and event nodes from which alternative decisions and events, respectively, are branched out. Economic consequences expressed in monetary or utility values for decisions, and probabilities for events are added and, starting at the end of the branches and working backwards, the expected payoff for a sequence of decisions is computed. The optimum sequence of decisions (optimum sequence of projects/sub-projects) can thus be identified. Since the number of events that can be branched out of an event node is limited, stochastic decision trees were developed, where the event nodes are represented by a probability distribution. However, resource constraints cannot be included in a decision tree, which is a serious drawback, amongst others.

With the advent of OR/MS and the wide availability of computers, different portfolio models were developed to overcome the shortcomings of the above methods. Several types of mathematical programming (linear, integer, mixed integer,

zero-one, nonlinear, dynamic, goal, multi-objective, and stochastic) have been used to select a subset of projects that maximizes a particular objective without violating a set of constraints. The objective is usually to maximize the expected net present value of the subset of projects. In addition to providing the optimum portfolio of R&D projects and allocating the budget among them, mathematical programming presents the additional advantage of making sensitivity analysis easy, suggesting ranges of solutions, and answering what-if types of questions. However, the optimality of the portfolio is a function of the assumptions associated with each particular type of mathematical programming and the estimates used in the input data. This, combined with the difficulty on the part of decision-makers in understanding the mathematical aspects of the models, has resulted in limited reported successes of the portfolio models in terms of actual implementation.

During the 1970s, issues regarding the usefulness of the existing methods for solving real life R&D project selection problems and their acceptability by R&D managers were widely discussed. Two studies by Baker (1974) and Baker and Freeland (1975) identified several limitations of the methods that had been proposed up until that time. Such limitations included the inadequate treatment of uncertainty and risk with respect to benefit contribution and parameter estimation; project and parameter relationships with respect to both benefit contribution and resource utilization; multiple, interrelated decision criteria; the time variant property of data and criteria; and the problems associated with the continuity in the research program and the research staff.

Further limitations were the lack of explicit recognition of the experience and knowledge of the R&D manager; the non-monetary aspects such as establishing and maintaining a balance between basic and applied research, product and process development, in-house and contracted projects, improvement and breakthrough work, and different levels of risk-pay-off opportunities; the perceptions held by R&D managers that the models are difficult to understand and use; and the importance of certain individuals in the R&D organization.

Other limitations include the failure to treat the problem as an intermittent stream of investment alternatives and as a hierarchical diffuse decision process; to include in the model the timing of decisions, the generation of additional alternatives, and project recycling by gathering new information, reformulating criteria, variables and constraints, and defining new alternatives; and to recognize the diversity of projects from basic research to engineering.

Several OR/MS researchers tried to develop approaches that do not have these limitations. As a result, different models have been proposed which take into consideration a particular aspect of the shortcomings of the existing methods. Among the approaches that have been developed, the emphasis has been on multi-objective mathematical programming methods. In this respect, goal programming methods have been developed, where several goals are considered and expressed as constraints, with deviational variables used to express under-achievement or over-achievement of the goals. The objective function minimizes these deviations. The goals can be prioritized, so that their achievement is considered according to their priority sequence.

As difficulties arise in setting the aspiration levels of the goals and in including tradeoffs among goals, multi-objective linear programming methods were used, including multi-attribute utility theory (Ringuest and Graves 1989; Mehrez et al. 1982). In applying multi-attribute utility theory, utility values are assigned to each possible subset of projects for each of the goals and, using integer programming, a list of all non-dominated solutions is generated, consisting of solutions in which the performance in one goal cannot be improved without sacrifice in one or more other goals. One drawback is that the list of non-dominated solutions can be very large in a real life situation, creating a complex selection problem for the decision-maker. Screening methods that have been developed could provide some help in selecting one of the non-dominated solutions.

Several researchers have proposed multi-criteria and multi-objective approaches. Stewart (1991) developed a multi-criteria decision support system for R&D project selection. Medaglia, Graves and Ringuest (2007) have proposed an evolutionary method with partially funded projects, multiple (stochastic) objectives, project interdependencies (in the objectives), and a linear structure for resource constraints. Stewart and Mohamed (2002) have developed a decision-making framework for senior

executives when selecting innovative IT/IS projects, based on the multi-criteria utility theory combined with information economics principles. Guikema and Milke (2003) have proposed a multi-attribute optimization model based on a combination of multi-attribute utility theory, mixed-integer optimization, and statistical analysis. Gabriel et al. (2006) have developed a multi-objective, integer-constrained optimization model using probability distributions to describe costs and the Analytic Hierarchy Process to determine the criterion rank; it integrates multi-objective optimization, Monte Carlo simulation, and the Analytic Hierarchy Process.

Most of the suggested models that are based on mathematical programming techniques have almost exclusively dealt with R&D activities at the micro-level. Oral et al. (1991) have proposed a methodology for collective evaluation and selection of R&D projects at the macro-level (sectorial or national), where the experts participating in the evaluation and selection process are also stakeholders. The evaluation and selection process is based on the "relative values" of a given R&D project from the viewpoint of the other R&D projects, determined through mathematical programming. Oral et al. (2001) have developed a methodology for an international organization which has more than a dozen country members whose units or divisions have different values and preferences; the methodology is based on a multi-criteria disaggregative approach used as an instrument rather than as a descriptive tool that provides a platform to maximize the level of consensus among the member countries.

As projects may depend upon each other with respect to several factors such as cost and technology, cross impact may be significant. Cho and Kwon (2004) have developed an extended model of the Analytic Hierarchy Process called Cross-Impact Hierarchy Process through which a number of dependent technological alternatives are ranked. Kwon et al. (2004) have constructed a model which evaluates R&D projects considering cross impact among them, and selects proper projects to utilize resources efficiently as well as to maximize efficacy of investments. Wey and Wu (2007) have proposed a project selection methodology that reflects interdependence among evaluation criteria and candidate projects using the Analytic Network Process within a zero-one goal programming model.

Mittal and Kanda (2009) use two-phase heuristics based on a two-stage prioritization process of activities for resource allocation. At any decision point the projects are first prioritized as per project selection rule and eligible activities in the projects are then prioritized as per activity selection rule. These heuristics are categorized into look-ahead and non-look-ahead type based on the project selection rules used.

Mohanty et al. (2005) have applied a Fuzzy Analytic Network Process along with fuzzy cost analysis in selecting R&D projects. Fuzzy set theory has been incorporated to overcome vagueness in the preferences of the various stakeholders in an organization, which can differ and often hinder the attainment of consensus and coordination.

A different philosophy in approaching the problem of R&D project selection and resource allocation has been proposed by including in the decision making process the people at every level of the organization who would influence the project selection process. As a result, Behavioral Decision Aids (BDA) have been proposed, that use the output of project selection models, not as a solution to the problem, but as aids to communication and interaction among the parties involved to achieve a consensus. One such approach is Q-Sorting, where each individual is given a stack of cards, each bearing the title or number of one project. Through a series of sorting operations and the use of a specific criterion, the projects are sorted into five piles ranging from very high level of the criterion to very low. Another BDA approach is the Nominal Interactive Decision Process, used in combination with various other methods depending on the type of the project, where consensus is built using a modified Delphi approach.

The Analytic Hierarchy Process has also been used in what is called Decentralized Hierarchical Modeling, where the involved parties communicate electronically until they reach a consensus on the project portfolio. The dialogue takes place among the different hierarchical levels. Top management initiates the process through budgetary guidelines sent to the divisional managers; the divisional managers then send the guidelines, maybe modified, together with suggested prioritized program areas to the R&D

managers; and the R&D managers and their staff propose an R&D portfolio and send it up the hierarchy. The R&D people, in coming up with the portfolio, may use any of the available OR/MS techniques. This process may be repeated several times. Back and forth communication among the different hierarchical levels may take place at any stage of the process.

Another multi-criteria decision approach to R&D project selection has been applied by Henig and Katz (1996). It is based on an objective investigation into the impact of alternatives (portfolios of projects) on attributes and to the subjective evaluation of the decision-maker's preference system. The main stages of this application are: identifying the initial set of projects, their important attributes, and criteria associated with them; consolidating or revising attributes based on the criteria; and evaluating existing and searching for new alternatives based on the reevaluated attributes. This process goes through several iterations and helps the decision-maker to better understand the nature of the problem. The "objective hierarchy" (Keeney and Raiffa 1976) of the decision maker is thus generated and the best alternative can then be found.

There have been a few approaches that have been proposed for large-scale R&D program planning. Such programs involve multiple interdependent technologies. They are initially defined in terms of broad, qualitative policy directives, serve broad constituencies of sponsors, and R&D is performed at separate external organizations or at remote sites within an organization. Decisions regarding resource allocation among projects, establishment of objectives, assignment of projects to program sub-divisions, and scheduling of the projects need to be made. For a decision-support approach and pertinent bibliography see Mathieu and Gibson (1993).

Hueth et al. (2008) have proposed a mixed integer programming model that selects projects worthy of investment in a public utility company (a major Latin America water and sewage company). It maximizes the weighted sum of normalized economic and financial net present values and a social impact index. Buchanan and Vanderpooten (2007) describe a project selection methodology which incorporates a decision support tool (ELECTRE III), developed by a New Zealand electric generator, used in ranking and selecting projects.

Project Planning and Control

The second major area of R&D management to which OR/MS has made a significant contribution is project planning and control. Several approaches have been developed in this area as well.

One of the first approaches is Program Evaluation and Review Technique (PERT), developed in 1958 for planning and controlling the Polaris Fleet Ballistic Missile project by the Navy Special Projects Office and Lockheed Aircraft Corporation, in cooperation with Booz, Allen and Hamilton. In using PERT, the project is represented by a network, consisting of events (nodes) standing for specific accomplishments at a point in time, or milestones, and activities (arrows) representing the actual performance of a task. The events and activities follow one another in their proper technological and logical sequence, and the PERT network, also called the precedence relationships network, has a beginning event and an ending event. Activities consume time and resources such as manpower, materials, equipment, funds, and so on, and each activity is represented by the beginning and ending nodes. This means that only one activity can connect two nodes and that the network cannot have a loop.

Activity times are assumed to follow the beta distribution and three time estimates are given for each activity: an optimistic standing for the practically minimum time, a most likely standing for the best estimate of time, and a pessimistic standing for the practically maximum time. On the basis of these estimates, the mean and variance for each activity time is computed, the longest time path is determined (critical path), and the probability of reaching an event or of completing the critical path by a certain scheduled time is computed. The latter is usually taken as the probability of completing the project. Other important information derived from PERT is the earliest start and finish time, the latest start and finish time, and the slack time for each activity. Activities with zero slack time are considered critical activities that require special monitoring to avoid delays in the completion of a project.

At about the same time of the PERT development, the Critical Path Method (CPM) was developed in 1957 at the du Pont Company, in consultation with Remington Rand, in scheduling maintenance

shutdowns of chemical processing plants. PERT is a probabilistic approach whereas CPM is deterministic. CPM uses normal and crash time estimates for each activity, and their associated costs. The normal time estimate would be equivalent to PERT's expected time, and the crash time would be the minimum possible time needed to complete the activity irrespective of cost increases. Aside from these differences, CPM is almost identical to PERT, and for this reason the two techniques are referred to as the PERT/CPM technique. PERT and CPM have generated several variations, modifications, and new techniques with added capabilities and wider applications.

A number of project management software have been developed and continue being developed and updated. Mellentien and Trautmann (2001) have evaluated the resource allocation capabilities of a few of them: Acos Plus 1.8.2, CA SuperProject 5.0a, CS Project Professional 3.0, MS Project 2000, and Sctor Project Scheduler 8.0.1. Among them, Acos Plus.1 and Project Scheduler showed the best resource allocation performance. Several other software have been mentioned in the OR/MS bibliography like P3, MATLAB, Expert Choice, Time Line, Primavera Project Planner, Milestone, and others. There is a long list of current software published in Wikipedia which have been evaluated according to their capabilities. It can be accessed by searching for project management software.

PERT and CPM have been used in a variety of project planning and control situations, including R&D project management. However, some of their underlying assumptions are not always valid. For example, the originally developed network may become irrelevant in the future because of the changed content of a project. Precedence relationships cannot always be specified as they sometimes depend on the outcome of previous activities. Project completion time is not always determined by the longest time path, as a delay in a non-critical activity may result in a longer completion time of the project. The beta distribution is not the only distribution that could be used, the formulae used to estimate its mean and variance may give erroneous estimates compared with the original beta distribution formulae, and the three time estimates may include a high degree of subjectivity (Chase et al. 1998).

In addition to the criticisms of PERT and CPM, their use in R&D project planning and control includes some additional limitations (Clayton and Moore 1972; Pritsker et al. 1989). One of them is that branching from the nodes is deterministic, that is, each activity must be completed before the project is completed. In R&D projects, however, branching is usually probabilistic, for example, successful test and performance of the next stage, failure and abandonment of that part of the project, inconclusive results and repeat of the test. All the activities leading to a node must be realized before the relevant event can be realized, while in R&D projects, given the probabilistic nature of branching, not all activities leading to a node can be realized. Looping is not allowed, though an activity in an R&D project may have to be repeated, for example, a test. Activity times are assumed to be solely described by a beta distribution, while R&D activities may follow different other distributions. One terminal node is allowed (completion of the project) while in R&D one of several end events can be realized, for example, successful completion, failure and abandonment, redesign of the project.

It is obvious that these limitations render PERT inflexible in modeling complex R&D projects. To overcome PERT's limitations, the Graphical Evaluation and Review Technique (GERT) was developed under the assumption that each activity has an associated probability of being selected, ranging from zero to one. As a result, the nodes are constructed differently to denote their nature as deterministic or probabilistic. The realization of a node may be specified to occur upon the realization of one or more of the activities leading to it, it may be realized one or more times, and the first time it is realized the number of activities to be completed may be different from subsequent repeats. Looping in simple or complex forms is allowed. The network can have more than one source node and/or sink node. Modifications of the network following the completion of certain activities can be incorporated. Several types of probability distributions can be used to represent activity times. Cost can be assigned to each activity in terms of a fixed part and a variable per unit time component. Statistics on time, cost, and activity counts for specified activities can be collected for the sink as well as other designated nodes. GERT is a network-simulation approach that has been further

improved into a more powerful version, *Q-GERT*, that allows the inclusion and the simulation of more than one project, can model queues at nodes and route projects through teams based on user established decision rules (Taylor and Moore 1980; Pritsker et al. 1989).

In decision models for GERT networks, cost or time minimization is achieved under the assumption that the selection procedure of activities and the stopping rule are imposed exogenously into the model. However, in classical R&D projects, the activities and technological specifications are selected dynamically throughout the project's duration, and the stopping time of the project is unknown in advance. Granot and Zuckerman (1991) have constructed a model in which the selection procedure of activities and the stopping rule are defined as decision variables to be determined endogenously.

Another network-simulation based technique that was developed after GERT is Venture Evaluation and Review Technique (VERT). VERT, like GERT, has been developed as a technique for analyzing potential outcomes of projects, expected values of various project parameters, and criticality indices rather than for scheduling projects. It is used in assessing the risks involved in undertaking new ventures and in resource planning, control monitoring, and overall evaluation of on-going projects with respect to time, cost, and performance. It is considered to be more powerful than GERT due to the fact that performance enters the network in numerical terms. It can be modeled in terms of any unit of measurement or a dimensionless index. VERT introduced six new types of node logics and the capability of establishing a mathematical relationship between an arrow's parameter values (time, cost, performance) and any other arrow or node's parameter values, as well as mathematical relationships between the time, cost, and performance variables of a given arrow.

There have been several studies that have tried to provide solution approaches to the different complexities of the real R&D project planning and control problem. Hans et al. (2007) have looked into several viewpoints on the management of the planning complexity of multi-project organizations under uncertainty. They proposed a positioning framework to distinguish between different types of project driven organizations and thus aid project management in the choice between the various existing planning approaches. They also introduced a generic hierarchical project planning-and-control framework that serves to position planning methods for multi-project planning under uncertainty.

PERT assumes that activities are independent and they occur in linear sequence. However, R&D activity networks are recursive, reversing to earlier stages for rectification or to incorporate changes. Hardie (2001) has modeled such a network as a Markov process, where the probability of a reversion to an earlier stage is claimed to be the main factor determining project length.

Managing R&D projects under resource constraints is a usual case. Tormos and Lova (2001) have extended the concepts of activity slack and defined a new activity criticality index to classify the activities of the resource constrained project scheduling and control context. These new concepts have been integrated into standard project management software. Pantouvakis and Manoliadis (2006) have developed a heuristic method based on traditional CPM scheduling calculations and leveling algorithms for the resource constrained projects, which can be applied using normal scheduling software such as P3 and MSProject.

As a project is being implemented, it is necessary at some points that management evaluates the progress of the plan, analyze deviations and take appropriate corrective action. Falco and Macchiaroli (1998) have provided a quantitative determination of the optimum control points, based on the definition of an Effort Function. Raz and Erel (2000) have presented an analytical framework for determining the optimal timing of project control points, based on maximizing the amount of information generated by the control points. This depends on the intensity of the activities carried out since the last control point and on the time elapsed since their execution. They have used dynamic programming to solve the optimization problem. Given the uncertainty involved in planning an R&D project, Badri et al. (1997) have developed a simulation based decision support system to analyze the effect of delays in individual activities on the whole project.

A project may have to be terminated, due to technological risks, before completion, with each stage having a specific likelihood of success. Reyck and Leus (2008) have proposed an approach of scheduling projects in order to maximize their

expected net worth value when the project activities have a probability of failure and when an activity's failure leads to overall project termination. Managerial flexibility with respect to the option of abandonment of a project or taking corrective action is referred to as real option value. Huchzermeier and Loch (2001) have identified five example types of R&D uncertainty: in market payoffs, project budgets, project performance, market requirements, and project schedules. They have developed a model based on options pricing theory, which builds intuition for R&D managers as to when it is and when it is not worthwhile to delay commitments.

Al Subhi Al-Harbi (2001) presents the Analytic Hierarchy Process as a potential decision making method in project management using as an example the contractor prequalification problem. A hierarchical structure is constructed for the prequalification criteria and the contractors wishing to prequalify for a project. Then the prequalification criteria are prioritized and a descending-order list of contractors is made in order to select the best contractors. A sensitivity analysis can be performed to check the sensitivity of the final decisions to minor changes in judgments.

Kablan and Dweiri (2006) identify three criteria that may be considered as project management internal measures of efficiency: project cost, project time, and project quality. They present an approach that employs fuzzy decision making to combine these three measures into one measure they name project management internal efficiency, representing an overall estimate of how well the project was managed and executed.

Lova et al. (2000) have dealt with the problem of managing various projects that share a pool of constrained resources. They have developed a multi-criteria heuristic algorithm that improves lexicographically two criteria, each chosen by the user from two types: time type (mean project delay, multi-project duration increase), and no time type (project splitting, in-process inventory, resource leveling, idle resources). The multi-criteria heuristic consists of several algorithms based on the improvement of multi-project feasible schedules.

Wiley et al. (1998) have looked at the application of optimization techniques to the initial design and development of multi-project programs. The classic work breakdown structure is used as a framework to provide an aggregate model to investigate the effects of funding levels, resource allocation, and program, project, and component durations. Decomposition, sensitivity analysis, and parametric programming are utilized to provide the decision maker detailed information for establishing program parameters, conditions, and bounds.

Concluding Remarks

In summary, there are hundreds of OR/MS models that have been proposed for R&D management. The evolution of the field continues through newly proposed models that appear in the references, which try to improve upon existing ones by including additional aspects of the problem situation. The trend has been to recognize the multi-criteria and multi-objective nature of real world R&D projects as well as their interdependency with respect to resources, technologies, and other factors. The usage of these approaches is expected to increase in the future due to the wider exposure of R&D managers to OR/MS approaches, the wider availability of computers and user-friendly software, and the emphasis on using them "as a laboratory for testing policies, sharing opinions, asking 'what-if' types of questions and simulating interdepartmental interactions throughout the organization" (Souder and Mandakovic 1986).

Williams (2003) looked at the contribution that mathematical modeling has made to project management over the past 50 years (at the time of his study), and the contribution it is currently making and can make in the future. He maintains that project management started with well-defined foundations and modelers played an essential role in offering solutions. Since then, he concludes that much of the mathematical-modeling world continued producing ever more complex solutions to ever more complex models that did not help in solving real-world problems. However, several of the models built during the last couple of decades are systemic and dynamic and explain many of the behaviors of R&D projects.

See

- ▶ Analytic Hierarchy Process
- ▶ Analytic Network Process
- ▶ Decision Trees
- ▶ GERT

- ▶ Goal Programming
- ▶ Integer and Combinatorial Optimization
- ▶ Linear Programming
- ▶ Multi-attribute Utility Theory
- ▶ Multiobjective Programming
- ▶ Multiple Criteria Decision Making
- ▶ Network Planning
- ▶ PERT
- ▶ Portfolio Theory: Mean-Variance Model
- ▶ Project Management
- ▶ VERT

References

Al-Subhi Al-Harbi, K. M. (2001). Application of the analytic hierarchy process in project management. *International Journal of Project Management, 19*, 19–27.

Augood, D. (1973). A review of R&D evaluation methods. *IEEE Transactions on Engineering Management, EM-20*, 114–120.

Badri, M. A., Mortagy, A., Davis, D., & Davis, D. (1997). Effective analysis and planning of R&D states: A simulation approach. *International Journal of Project Management, 15*, 351–358.

Baker, N. R. (1974). R&D project selection models: An assessment. *IEEE Transactions on Engineering Management, EM-21*, 165–171.

Baker, N. R., & Freeland, J. (1975). Recent advances in R&D benefit measurement and project selection methods. *Management Science, 21*, 1164–1175.

Baker, N. R., & Pound, W. H. (1964). R&D project selection: Where we stand. *IEEE Transactions on Engineering Management, EM-11*, 124–134.

Buchanan, J., & Vanderpooten, D. (2007). Ranking projects for an electricity utility using ELECTRE III. *International Transactions in Operational Research, 14*, 309–323.

Chase, R. B., Aquilano, N. J., & Jacobs, F. R. (1998). *Production and operations management: Manufacturing and services* (8th ed.). Homewood: Irwin-McGraw Hill.

Cho, K. T., & Kwon, C.-S. (2004). Hierarchies with dependence of technological alternatives: A cross-impact hierarchy process. *European Journal of Operational Research, 156*, 420–432.

Clayton, E. R., & Moore, L. J. (1972). PERT vs. GERT. *Journal of Systems Management, 22*, 11–19.

de Falco, M., & Macchiaroli, R. (1998). Timing of control activities in project planning. *International Journal of Project Management, 16*, 51–58.

De Reyck, B., & Leus, R. (2008). R&D project scheduling when activities may fail. *IIE Transactions, 40*, 367–384.

Gabriel, S. A., Kumar, S., Ordóñez, J., & Nasserian, A. (2006). A multi-objective optimization model for project selection with probabilistic considerations. *Socio-Economic Planning Sciences, 40*, 297–313.

Granot, D., & Zuckerman, D. (1991). Optimal sequencing and resource allocation in research and development projects. *Management Science, 37*, 140–156.

Guikema, S., & Milke, M. (2003). Sensitivity analysis for multi-attribute project selection problems. *Civil Engineering and Environmental Systems, 20*, 143–162.

Hans, E. W., Herroelen, W., Leus, R., & Wullink, G. (2007). A hierarchical approach to multi-project planning under uncertainty. *Omega, 35*, 563–577.

Hardie, N. (2001). The prediction and control of project duration: A recursive model. *International Journal of Project Management, 19*, 401–409.

Henig, M. I., & Katz, H. (1996). R&D Project selection: A decision process approach. *Journal of Multi-Criteria Decision Analysis, 5*, 169–177.

Huchzermeier, A., & Loch, C. H. (2001). Project management under risk: Using the real options approach to evaluate flexibility in R&D. *Management Science, 47*, 85–101.

Hueth, D., Medaglia, A. L., Mendieta, J. C., & Sefair, J. A. (2008). A multi-objective model for the selection and timing of public enterprise projects. *Socio-Economic Planning Sciences, 42*, 31–45.

Kablan, M. M., & Dweiri, F. T. (2006). Using fuzzy decision making for the evaluation of the project management internal efficiency. *Decision Support Systems, 42*, 712–726.

Keeney, R. L., & Raiffa, H. (1976). *Decisions with multiple objectives: Preferences and value trade-offs*. New York: Wiley.

Kwon, C.-S., Park, J.-H., & Hong, S.-K. (2004). Construction of 'CIDEAR' model for selecting and evaluating cross impact R&D projects. *Journal of the Korean OR/MS Society, 29*, 41–62.

Liberatore, M. J., & Titus, G. J. (1983). The practice of management science in R&D project management. *Management Science, 29*, 962–974.

Lova, A., Maroto, C., & Tormos, P. (2000). A multi-criteria heuristic method to improve resource allocation in multi-project scheduling. *European Journal of Operational Research, 127*, 408–424.

Mathieu, R. G., & Gibson, J. E. (1993). A methodology for large-scale R&D Planning based on cluster analysis. *IEEE Transactions on Engineering Management, 40*, 283–292.

Medaglia, A. L., Graves, S. B., & Ringuest, J. L. (2007). A multi-objective evolutionary approach for linearly constrained project selection under uncertainty. *European Journal of Operational Research, 179*, 869–894.

Mehrez, A. S., Mossery, S., & Sinuany-Stern, Z. (1982). Project selection in a small R&D laboratory. *R&D Management, 12*, 169–174.

Mellentien, C., & Trautmann, N. (2001). Resource allocation with project management software. *OR-Spektrum, 23*, 383–394.

Mittal, M. L., & Kanda, A. (2009). Two-phase heuristics for scheduling of multiple projects. *International Journal of Operational Research, 4*, 159–177.

Mohanty, R. P., Agarwal, R., Choudhury, R. K., & Tiwari, M. K. (2005). A fuzzy ANP-based approach to R&D project selection: A case study. *International Journal of Production Research, 43*, 5199–5216.

Oral, M., Kettani, O., & Lang, P. (1991). A methodology for collective evaluation and selection of industrial R&D projects. *Management Science, 37*, 871–885.

Oral, M., Kettani, O., & Çinar, Ü. (2001). Project evaluation and selection in a network of collaboration: A consensual

disaggregation multi-criterion approach. *European Journal of Operational Research, 130*, 332–346.
Pantouvakis, J.-P., & Manoliadis, O. G. (2006). A practical approach to resource-constrained project scheduling. *Operational Research, 6*, 299–309.
Pritsker, A. A. B., Sigal, C. E., & Hammesfahr, R. D. F. (1989). *SLAM II–network models for decision support*. Englewood Cliffs: Prentice Hall.
Raz, T., & Erel, E. (2000). Optimal timing of project control points. *European Journal of Operational Research, 127*, 252–261.
Ringuest, J. L., & Graves, S. B. (1989). The linear multi-objective R&D Project selection problem. *IEEE Transactions on Engineering Management, 36*, 54–57.
Schroder, H. H. (1971). R&D project evaluation and selection models for development: A survey of the state of the art. *Socio-Economic Planning Sciences, 5*, 25–39.
Souder, W. E. (1972). A comparative analysis of R&D investment models. *AIIE Transactions, 4*, 57–64.
Souder, W. E., & Mandakovic, T. (1986). R&D project selection models. *Research Management, 29*, 36–42.
Stewart, T. J. (1991). A Multi-criteria decision support system for R&D project selection. *Journal of Operational Research Society, 42*, 17–26.
Stewart, R., & Mohamed, S. (2002). IT/IS projects selection using multi-criteria utility theory. *Logistics Information Management, 15*, 254–270.
Taylor, B. W., & Moore, L. J. (1980). R&D project planning with Q-GERT network modeling and simulation. *Management Science, 26*, 44–59.
Tormos, P., & Lova, A. (2001). Tools for resource-constrained project scheduling and control: Forward and backward slack analysis. *Journal of the Operational Research Society, 52*, 779–788.
Wey, W.-M., & Wu, K.-Y. (2007). Using analytic network process priorities with goal programming in resource allocation in transportation. *Mathematical and Computer Modeling, 46*, 985–1000.
Wiley, V. D., Deckro, R. F., & Jackson, J. A. (1998). Optimization analysis for design and planning of multi-project programs. *European Journal of Operational Research, 107*, 492–506.
Williams, T. (2003). The contribution of mathematical modeling to the practice of project management. *IMA Journal of Management Mathematics, 14*, 3–30.

Resource Aggregation

In a project network, a method of scheduling activities within their available float times according to a specific rule, for example, at their earliest start times, and determining the consequent total units of each resource required in each time period.

See

▶ Network Planning

Resource Leveling

A method of scheduling activities of a project to meet a limit in the amount of a resource that is available. This may mean that the project completion date is allowed to slip.

See

▶ Network Planning
▶ Project Management

Resource Smoothing

A method of scheduling activities of a project within their available float times to minimize fluctuations in day-to-day resource requirements. This approach would be used when the project completion time is not allowed to slip.

See

▶ Network Planning
▶ Project Management

Response Surface Methodology

Russell R. Barton
The Pennsylvania State University, University Park, PA, USA

Introduction

Response surface methodology (RSM) is a technique to determine design factor settings to improve or optimize the performance or response of a process or

product. It combines design of experiments, regression analysis and optimization methods in a general purpose strategy to optimize the expected value of a stochastic response. In their landmark paper, Box and Wilson (1951) describe the development and application of this sequential method to chemical process design, in which yields of particular compounds were maximized. Since that time the method has been applied successfully in many areas. Recent texts devoted to RSM include Myers et al. (2009) and del Castillo (2007).

Problem Setting and Background

Mathematically, RSM solves:

$$\max f(x) \equiv \mathrm{E}(Y(x))$$

where Y is a random variable whose mean is an unknown function of the d-dimensional factor vector x and whose variance (arising from experimental error) is an unknown constant value, denoted σ^2; and where the maximization is over x in some region R. Generally the constraints describing R are not modeled, so the setting is usually considered an unconstrained optimization problem. RSM fits a sequence of local regression models, at first linear, and later quadratic. The models are fitted to experimental data based on a set of prescribed x vectors – also called the experiment design. The linear model has the form:

$$Y(x_i) = \beta_0 + \sum_{j=1}^{d} \beta_j x_{ij} + \varepsilon_i, \{\varepsilon_i\} \sim i.i.d.N\left(0,\sigma^2\right) \quad (1)$$

where i indexes the experimental run and j the component of the x_i vector. The fitted model is represented as

$$E(Y(x_i)) = \hat{y} = b_0 + \sum_{j=1}^{d} b_j x_{ij} = b_0 + b' x_i. \quad (2)$$

The quadratic model has the form:

$$Y(x_i) = \beta_0 + \sum_{j=1}^{d} \beta_j x_{ij} + \sum_{j=1}^{d} \sum_{k=j}^{d} \beta_{jk} x_{ij} x_{ik} + \varepsilon_i, \{\varepsilon_i\} \sim i.i.d.N\left(0,\sigma^2\right) \quad (3)$$

with fitted model

$$E(Y(x)) = \hat{y} = b_0 + \sum_{j=1}^{d} b_j x_j + \sum_{j=1}^{d} \sum_{k=j}^{d} b_{jk} x_j x_k = b_0 + b' x + x' B x. \quad (4)$$

The fitted coefficients b_0, b_j and b_{jk} are usually calculated via least squares. Note that in equation (4), the off-diagonal elements of B are half the magnitude of the b_{jk} values, i.e. $B[j,k] = B[k,j] = b_{jk}/2$. Also, $\{\varepsilon_i\} \sim i.i.d.N(0,\sigma^2)$ can be represented in multivariate form. If the experiment consists of n instances or runs, then ε is assumed to be a normally distributed random n-vector with mean the zero vector and covariance matrix $\Sigma_\varepsilon = \sigma^2 I$. Each fitted local model is used to determine a search direction or subregion in R where an increase or optimum of $\mathrm{E}(Y(x))$ is expected.

RSM is preferred over a simple grid search over R under the following conditions:

1. The response function f is complex: it is not well approximated by a single quadratic over all of R.
2. Experimental error (hence σ^2) is small enough to permit local characterization of f with relatively few experiment runs.
3. Derivatives of f are continuous.
4. Experiments can be carried out sequentially without undue delay or cost.

The first condition means that a grid search over d variables would require a fine rather than coarse grid. With l levels for each variable, the total number of experiments for a grid search would be l^d, i.e., a large number of different experimental conditions or *points*. The second condition says that a small number of experiment points will be sufficient to provide local approximation of the response by a low-order polynomial, without requiring many *replications* (repeated experiments under the same experimental conditions). The third condition permits local characterization of the response function by a low-order polynomial, typically linear or quadratic. The fourth condition allows each local approximation to be used to identify a next local subregion to explore. Since this sequential approach avoids local subregions that are not promising, fewer experiments are required to characterize the optimum subregion, compared with a grid search that characterizes all of R.

RSM Algorithm

At the highest level, RSM has four activities: scaling and transforming, screening, Phase 1 modeling and search, and Phase 2 modeling and search.

1. Scale the x variables (for example, to provide comparable units of change), and transform Y values (for example, to provide responses with homogeneous variance across different x values).
2. Screen for important components of the x vector (called factors - explained below).
3. Phase 1:
   a. Select an experiment design appropriate for fitting a first-order regression model.
   b. Conduct the experiment, fit the appropriate model, assess significance and fit.
   c. If the model is satisfactory, identify a search direction and conduct a sequence of experiments in this direction.
   d. Return to step 3a.
4. Phase 2:
   a. Select an experiment design appropriate for fitting a second-order regression model (this may be an augmentation of an existing first-order design).
   b. Conduct the experiment, fit the appropriate model, assess significance and fit.
   c. If the model is satisfactory, identify whether an optimum, saddle point or a ridge has been identified.
   d. If a ridge or optimal solution is contained in the experimental range, continue to step 5.
   e. Otherwise, identify a search direction and conduct a sequence of experiments in this direction.
   f. Return to step 4a.
5. Communicate the optimum settings or a ridge of near-optimum settings in the original units of x.

Scaling of the x variables is equivalent to determining the size of the local region for each variable, since the low and high values in a local region are generally scaled to +/− 1 for regression model fitting and analysis. The steepest ascent direction depends on this scaling. Transformation of Y values can be helpful in stabilizing the variance of $Y(x)$ across different x values, i.e., when $\mathrm{Var}(Y(x)) = \sigma^2(x)$. Since the models imply $\mathrm{Var}(Y(x)) = \mathrm{Var}(\varepsilon(x))$, transforming Y for homogeneous variance ($\sigma^2(x) = \sigma^2$ for every x) permits the use of statistical methods that assume $\{\varepsilon_i\} \sim i.i.d.N(0,\sigma^2)$, given independence of experimental runs. Nonhomogeneous variance is common in simulation settings, often having a power relationship with components of x. The Box and Cox (1964) power transformations are useful in this case. The family of transformations is parameterized by λ:

$$y(\lambda) = \begin{cases} \dfrac{y^\lambda - 1}{\lambda} & \lambda \neq 0 \\ \ln(y) & \lambda = 0 \end{cases}$$

where the natural log function provides continuity at $\lambda = 0$ and the parameter λ can be chosen by maximum likelihood or estimated as one minus the slope of a plot of the ln(estimated standard deviation of $Y(x)$) on the vertical axis vs. ln(estimated mean of $Y(x)$) on the horizontal axis (Montgomery 2009). Selecting a member of this transformation family requires replicated experiments at each design point. Transformation of Y can produce an additional benefit: the response surface is often less complex after the variance stabilizing transformation. This is because the nonlinearity of the response and the nonhomogeneity of the variance are often related, and so they are both removed or reduced by the same transformation. This allows the size of the local region to be increased, which permits faster progress for the optimization.

Alternatively, to achieve homogeneous variance, it is possible to conduct replications at each x point in the design, with the number or replications proportional to the variance at that x value. For large differences in the variance of $Y(x)$, many replications could be required, increasing the experimental costs of RSM.

Since the number of design points needed to fit a first-order (second-order) regression model increases linearly (quadratically) with the number of factors, screening out unimportant factors at the start of the response surface optimization can significantly reduce experimental effort. Screening experiment designs are typically fractional factorial designs, and often have $d + 1$ or fewer points for d factors. These designs are discussed by Satterthwaite (1959), Plackett and Burman (1946) and Lin (1993). Screening is performed by examining the regression equation. If fewer than $d + 1$ runs are conducted, this can be done using stepwise regression or other strategies described in Li and Lin (2003). Sequential screening methods

group factors together in setting their high and low values, to eliminate large numbers of factors in a relatively small number of runs. This strategy requires a priori knowledge of the sign of the factor, which is practical in some physical settings and uncertain in others. Sanchez et al. (2009) describe a strategy for dealing with uncertain signs, and summarize the original developments in this area by other researchers, including Bettonvil, Cheng and Kleijnen.

Phase 1 consists of building and exploiting local linear approximations to the response function. Typically a factorial or fractional factorial design is used for experimentation, along with a center point, with replications that allow a statistical test of lack of fit of the first-order model. The region for experimentation is local rather than all of R. The local region must be chosen small enough so that a linear approximation is likely to be adequate, but large enough so that the linear effects can be detected as statistically significant without large numbers of replications. The results of the regression analysis ideally provide a statistically significant model with no significant lack of fit. In this case a sequence of experiments are conducted, from the design center in the direction of the gradient. The increment between experimental points along this line is usually chosen to give a magnitude of 1 for the largest gradient component, so the increment in this direction would be $b/\max\{|b_j|\}$. Steps continue until the observed response decreases, but various rules have been proposed to use additional steps to ensure that random error does not cause premature termination. These rules include i) using multiple replications and testing for statistical significance of the change from one step to the next, ii) fitting a univariate polynomial to the responses observed along the search direction and stopping when the fitted maximum lies within the scope of the search, iii) stopping after three consecutive failures to increase the response. These and other stopping rules are compared in del Castillo (2007).

Outcomes other than a statistically significant model with no significant lack of fit are possible in Phase 1. Table 1 summarizes the possibilities and the corresponding actions as presented in Barton and Meckesheimer (2006).

Entry to Phase 2 is based on lack of fit in Phase 1. The local region then usually remains the same, since the Phase 1 factorial or fractional factorial design is usually retained, and augmented with design points at +/-δ on the factor axes. A design having a full factorial augmented by center and axis points is called a central composite design. For central composite designs, δ can be chosen to make the design rotatable. Rotatable designs give regression models with equal prediction variance at all points a fixed distance from the center of the design, and have some advantages, as discussed by Box and Wilson (1951) and others. Other types of rotatable (or near-rotatable) designs have been used, for example, Box-Behnken designs (Box and Behnken 1960). Donohue et al. (1995) construct designs that minimize sensitivity to model misspecification. These designs provide coefficient estimates that are least sensitive to the presence of higher-order terms in the true local response. As for Phase 1, the results of a Phase 2 regression analysis ideally provide a statistically significant quadratic model with no significant lack of fit. In this case, a canonical analysis is usually performed. This analysis transforms x-space to be centered at the zero slope of the fitted quadratic: $x_0 = -1/2\ B^{-1}b$ and a transformed factor vector $w = M'(x\text{-}x_0)$, where M is the matrix of normalized column vectors corresponding to the eigenvectors of B. When the fitted quadratic has all negative eigenvalues, the location of the origin for w is x_0, the maximum of the fitted quadratic, and the w axes correspond to the major and minor axes of the ellipsoidal contours of the quadratic. The canonical model is:

$$E(Y(w)) = \hat{y} = w_0 + \sum_{j=1}^{d} \lambda_j w_j,$$

Response Surface Methodology, Table 1 Phase 1 Assessment of Significance and Fit Results and Actions

	Linear Effects Not Significant	Linear Effects Statistically Significant
Lack of Fit of Linear Regression	Augment design and fit quadratic regression model, go to Phase 2.	Augment design and fit quadratic regression model, go to Phase 2.
No Significant Lack of Fit	Choose a larger local region for experimentation or increase the number of replications at each design point. Go to step 3a.	Go to step 3c: identify a search direction and conduct a sequence of experiments in this direction.

Response Surface Methodology, Table 2 Phase 2 Assessment of Significance and Fit Results and Actions

	Quadratic Effects Not Significant	Quadratic Effects Statistically Significant
Lack of Fit of Quadratic Regression	Unlikely. If this occurs, one could select a higher-order model (cubic) but instead one would usually reduce the size of the local region and go to step 4a.	Reduce the size of the local region and go to step 4a.
No Significant Lack of Fit	Unlikely. If this occurs, increase the number of replications and go to step 4b.	Perform canonical analysis. If result is an optimum within the range of the local design, go to step 5. Otherwise, conduct a search or identify a best value within a fixed distance from the center of the design, then go to step 4a.

where the λ_j are the eigenvalues corresponding to the eigenvectors of B. Canonical analysis allows easy identification of the nature of the fitted quadratic. If all λ_j are positive, a local minimum has been fitted; if all are negative, a local maximum. When some are positive and some negative, a saddle point is identified. In this case, a search proceeds in a direction that will increase the objective, for example by the largest amount for a given step length, r.

Small eigenvalues correspond to directions of little change in the objective, usually called ridges. The ridges identified through canonical analysis provide a set of alternative factor settings (along the ridge) that provide nearly the same response. When a ridge exists, the decision maker has added flexibility in choosing among factor settings along the ridge based on secondary criteria.

Confidence regions for the optimal value x^* and optimal value of $E(Y(x^*))$ can be constructed from the fitted regression model. These are presented in the original paper by Box and Wilson (1951) and the discussion accompanying the paper. Subsequent enhancements to these methods, in particular to the constrained case, are explained in detail in del Castillo (2007).

As for Phase 1, outcomes other than a statistically significant model with no significant lack of fit are possible in Phase 2. Table 2 summarizes the possibilities and the corresponding actions as presented in Barton and Meckesheimer (2006), which also contains a completely executed example of RSM.

Application of RSM to Discrete-Event Simulation

The output of discrete-event dynamic simulations are stochastic responses, and RSM was recognized as a tool for simulation optimization in the early development of that field (Mihram 1970). In discrete-event simulation, random variation is introduced using pseudorandom number generation. This control permits deliberate introduction of correlation in responses across design points, which was exploited by Schruben and Margolin (1978) for the case where the design can be decomposed into orthogonal blocks. For the central composite design, for example, inducing positive correlation across factorial points, positive correlation across axis points, and negative correlation between these blocks, greater precision is possible in the regression model. This advantage depends on the ability to induce correlation in the output values based on common and antithetic input pseudorandom number streams, which can be hard to achieve. Successful correlation-induction strategies are discussed in simulation texts, for example Law and Kelton (2000) and Banks et al. (2009). Statistical issues associated with common and antithetic variate strategies for response model fitting were examined by a number of researchers, including Nozari et al. (1987), Tew and Wilson (1992, 1994) and Donohue (1995).

Variants and Properties

Many variants have been proposed to the RSM structure described above. Many recent contributions have come from research in the discrete-event simulation community, but with broad applicability to RSM outside the simulation domain, so they are not identified separately.

A number of modifications relate to the models and estimators in equations (1–4). A general variance structure implying $\Sigma_\varepsilon \neq \sigma^2 I$ occurs for the correlation-induction strategies developed for simulation applications, but can also occur in

physical experimentation. In this case, weighted least squares or generalized least squares can be used to estimate the regression coefficients in equations (2) and (4). Independent nonhomogeneous variance is a special case of general variance. In this case Σ_ε is diagonal but with unequal diagonal values. Consequently weighted least squares provides an alternative to the strategies of transformation of Y or differential replication that were described above. Bayesian estimation can be used to determine regression coefficients (Cheng and Currie 2004; del Castillo 2007). This can be applied in physical or computer experimentation if prior experience or theory suggest appropriate ranges for β_j values.

The models in (1) and (3) can be replaced by generalized linear models which provide more flexible distribution forms for the random component and more general modeling terms as well. Staum (2009) describes weighted least squares, generalized least squares and generalized linear models for RSM. A broad class of model-based optimization methods use global approximations, for example, Kriging, splines, or neural networks. Barton and Meckesheimer (2006) review global model-based optimization and contrast it with RSM.

If the Phase 1 search is repeated several times, one can use a modified search direction to avoid the zigzagging that may occur. Conjugate gradient directions from nonlinear optimization were applied in the RSM setting by Joshi et al. (1998). The Phase 1 design is often not rotatable, and so the prediction variance in the gradient direction may increase more rapidly than in other directions. Kleijnen et al. (2004) used this concept to develop a search direction different from b and an interval different from $b/\max\{|b_j|\}$. For a fitted first-order model, they identify the point x^+ that maximizes a lower 1-α confidence interval for the predicted mean. This "adapted steepest ascent" search direction and interval is invariant to the scaling of the x components, unlike steepest descent, and the authors find that the performance is usually improved over steepest ascent.

Although the original form of RSM does not consider explicit constraints on the value of x, constraints must usually be taken into account (see Kleijnen 2008). When constraints are linear, linear programming may be used in Phase 1 optimization, and quadratic programming in Phase 2. For nonlinear constraints, other techniques have been proposed. Biles (1974) used a gradient projection search for RSM applied to the optimization of discrete-event simulation output. Robust design methods present an RSM situation with constraints that are themselves based on a model (del Castillo 2007). For this setting Bettonvil et al. (2009) developed statistical tests based on the Karush-Kuhn-Tucker conditions for nonlinear programming to check for optimality.

Box, in the discussion section of Box and Wilson (1951) warned against automating RSM. He felt that the specifics of each application would require the judgment of statisticians and process experts as decisions were made on experiment design choice, search direction choice and model choice. Nonetheless it is tempting, when applying RSM to the optimization of a computer simulation response function, to formalize and code the RSM process to permit an automated algorithm. Formal RSM algorithms were described by Neddermeijer et al. (2000), Nicolai et al. (2004), and Barton and Meckesheimer (2006). A formal structure naturally leads to a question of the convergence properties of the method. No convergence results were available for RSM until recently. Chang and Wan (2009) developed a version of RSM that chooses the step size and direction based on a trust region for the approximating model. They prove convergence under assumptions related to boundedness of the expected value of the maximum response, a mean response that is twice differentiable, bounded below and has uniformly bounded Hessian and gradient, and that for sufficiently small region the quadratic regression model (3) holds (possibly with nonhomogeneous variance). Convergence in this case means that with probability 1 the norm of the gradient of the $E(Y(x))$ with respect to x will go to zero as the number of successful iterations goes to infinity.

See

- ▶ Nonlinear Programming
- ▶ Regression Analysis
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Simulation Metamodeling
- ▶ Simulation Optimization
- ▶ Variance Reduction Techniques in Monte Carlo Methods

References

Banks, J., Carson, J. S., II, Nelson, B. L., & Nicol, D. M. (2009). *Discrete-event system simulation* (5th ed.). Upper Saddle River, NJ: Prentice Hall.

Barton, R. R., & Meckesheimer, M. (2006). Metamodel-based simulation optimization. Chapter 18. In S. G. Henderson & B. L. Nelson (Eds.), *Simulation: Handbooks in operations research and management science*. Elsevier B.V.

Bettonvil, B., del Castillo, E., & Kleijnen, J. P. C. (2009). Statistical testing of multiresponse simulation-based optimization. *European Journal of Operational Research, 199*, 448–458.

Biles, W. E. (1974). A gradient regression search procedure for simulation experimentation. In H. J. Highland, H. Steinberg, & M. F. Morris (Eds.), *Proceedings of the 1974 winter simulation conference* (pp. 491–497). New York: Association for Computing Machinery.

Box, G. E. P., & Behnken, D. W. (1960). Some new three-level designs for the study of quantitative variables. *Technometrics, 2*, 455–475.

Box, G. E. P., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society, Series B, 26*, 211–252.

Box, G. E. P., & Wilson, K. B. (1951). On the experimental attainment of optimum conditions. *Journal of the Royal Statistical Society, Series B, 13*, 1–45.

Chang, K.-H., & Wan, H. (2009). Stochastic trust region response surface convergent method for generally distributed response surface. In M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, & R. G. Ingalls (Eds.), *Proceedings of the 2009 winter simulation conference* (pp. 563–573). Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Cheng, R. C. H., & Currie, C. S. M. (2004). Optimization by simulation metamodelling methods. In R. G. Ingalls, M. D. Rossetti, J. S. Smith, & B. A. Peters (Eds.), *Proceedings of the 2004 winter simulation conference* (pp. 485–490). Piscataway, NJ: Institute of Electrical and Electronics Engineers.

del Castillo, E. (2007). *Process optimization: A statistical approach*. New York: Springer (softcover reprint version 2010).

Donohue, J. M. (1995). The use of variance reduction techniques in the estimation of simulation metamodels. In C. Alexopoulos, K. Kang, D. Goldsman, & W. Lilegdon (Eds.), *Proceedings of the 1995 winter simulation conference* (pp. 194–200). Piscataway, NJ: Institute of Electronic and Electrical Engineers.

Donohue, J. M., Houck, E. C., & Myers, R. H. (1995). Simulation designs for the estimation of quadratic response surface gradients in the presence of model misspecification. *Management Science, 41*, 244–262.

Joshi, S., Sherali, H. D., & Tew, J. D. (1998). An enhanced response surface methodology (RSM) algorithm using gradient deflection and second-order search strategies. *Computers and Operations Research, 25*, 531–541.

Kleijnen, J. P. C. (2008). Response surface methodology for constrained simulation optimization: An overview. *Simulation Modelling Practice and Theory, 16*, 50–64.

Kleijnen, J. P. C., den Hertog, D., & Angün, E. (2004). Response surface methodology's steepest ascent and step size revisited: Correction. *European Journal of Operational Research, 170*, 664–666.

Law, A. M., & Kelton, W. D. (2000). *Simulation modeling and analysis* (3rd ed.). New York: McGraw-Hill.

Li, R., & Lin, D. K. J. (2003). Analysis methods for supersaturated design: Some comparisons. *Journal of Data Science, 1*, 249–260.

Lin, D. K. J. (1993). A new class of supersaturated designs. *Technometrics, 35*, 28–31.

Mihram, G. A. (1970). An efficient procedure for locating the optimal simular response. In P. J. Kiviat & M. Araten (Eds.), *Proceedings of the fourth annual conference on the applications of simulation* (pp. 154–161). New York: Association for Computing Machinery.

Montgomery, D. C. (2009). *Design and analysis of experiments* (7th ed.). New York: John Wiley and Sons.

Myers, R. H., Montgomery, D. C., & Anderson-Cook, C. M. (2009). *Response surface methodology* (2nd ed.). New York: John Wiley and Sons.

Neddermeijer, H. G., van Oortmarssen, G. J., Piersma, N., & Dekker, R. (2000). A framework for response surface methodology for simulation optimization. In J. A. Joines, R. R. Barton, K. Kang, & P. A. Fishwick (Eds.), *Proceedings of the 2000 winter simulation conference* (pp. 129–136). Piscataway, NJ: Institute of Electronic and Electrical Engineers.

Nicolai, R. P., Dekker, R., Piersma, N., & van Oortmarssen, G. J. (2004). Automated response surface methodology for stochastic optimization models with unknown variance. In R. G. Ingalls, M. D. Rossetti, J. S. Smith, & B. A. Peters (Eds.), *Proceedings of the 2004 winter simulation conference* (pp. 491–499). Piscataway, NJ: Institute of Electronic and Electrical Engineers.

Nozari, A., Arnold, S. F., & Pegden, C. D. (1987). Statistical analysis for use with the Schruben and Margolin correlation induction strategy. *Operations Research, 35*, 127–139.

Plackett, R. L., & Burman, J. P. (1946). The design of optimum multifactorial experiments. *Biometrika, 33*, 305–325.

Sanchez, S. M., Wan, H., & Lucas, T. W. (2009). Two-phase screening procedure for simulation experiments. *ACM Transactions on Modeling and Computer Simulation, 19*, 1–24.

Satterthwaite, F. E. (1959). Random balance experimentation. *Technometrics, 1*, 111–137.

Schruben, L. W., & Margolin, B. H. (1978). Pseudorandom number assignment in statistically designed simulation and distribution sampling experiments. *Journal of the American Statistical Association, 73*, 504–520.

Staum, J. (2009). Better simulation metamodeling: The what, why and how of stochastic kriging. In M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, & R. G. Ingalls (Eds.), *Proceedings of the 2009 winter simulation conference* (pp. 119–133). Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Tew, J. D. A., & Wilson, J. R. (1992). Validation of simulation analysis methods for the Schruben-Margolin correlation-induction strategy. *Operations Research, 40*, 87–103.

Tew, J. D., & Wilson, J. R. (1994). Estimating simulation metamodels using combined correlation-based variance reduction techniques. *IIE Transactions, 26*, 2–16.

Response Time

Often used to describe the time between the arrival of a new queueing customer (e.g., as in the receipt of a call by an emergency dispatcher) and the initiation of service (as in the arrival of the emergency unit at the scene of the call), thus equal to the queueing delay.

Restricted-Basis Entry Rule

In the adaptation of the simplex algorithm for solving separable-programming problems in which variables are approximated by a set of grid variables, the restricted basis entry rule only allows, for each original variable, no more than two such neighboring grid variables to be in a solution. Such a rule is also used in solving quadratic-programming problems to force certain complementarity conditions to hold.

See

- ▶ Separable-Programming Problem
- ▶ Special-Ordered Sets (SOS)
- ▶ Wolfe's Quadratic-Programming Problem Algorithm

Retailing

Hiroaki Sandoh[1] and Kaoru Tone[2]
[1]Osaka University, Toyonaka, Osaka, Japan
[2]National Graduate Institute for Policy Studies, Minato-ku, Tokyo, Japan

Introduction

Retailing can be seen as the third phase in the flow of goods, following production and logistics. Retail shops are traditionally and historically classified into several categories: independent stores, department stores, supermarkets, discount stores and convenience stores. The advent of the Internet had a drastic impact on retailing, as it has alleviated restrictions in relation to time as well as space on the globe. As a result, online stores with global reach have led to even more intense competition between retailers.

Retailing has a wide variety of decision-making phases, and therefore OR/MS has been applied in many aspects with a view to obtaining scientific solutions to such areas as store location problems, product assortment along with shelf-space allocation, inventory control and advertising/sales promotion. Since the mid-1980s, applications of scientific methods have increased in retailing against growing competition among retailers in pursuit of larger profit. Many useful software packages have also been developed for the purpose of surviving in these extremely competitive environments.

Retail Store Location Problems

Decisions regarding store location are preliminary but vital for the retail management. Traditionally, OR/MS has addressed this problem along with general facility location problems. Models for a facility location problem often confine themselves to obtaining an optimal solution assuming the attractiveness of the facilities is known. For store location problems, however, how attractiveness can be measured is a critical concern, since development of accurate sales forecast is central to successful retail-site selection (Kotler 1984). For this reason, the following gravitational models are mainly used for retail store location problems.

Huff (1963) utilizes the conceptual properties of the gravity model, which is from the laws of Newtonian physics, to provide a probability that a customer patronizes a specific retail store. This gravity-based formula suggests that the probability is proportional to the attractiveness of the retail store and inversely proportional to the distance from it. More precisely, the attraction felt by customer i towards retail store $j(= 1, 2, \cdots, n)$ is expressed by S_j / T_{ij}^{λ}, where S_j denotes the selling space devoted to the sale of a specific class of goods by store j, T_{ij} the travel time of a customer S_j to retail store j, and λ a parameter which is to be estimated empirically to reflect the effect of travel time on various kinds of shopping trips. Then the probability that customer i chooses store j is given by $\left(S_j / T_{ij}^{\lambda}\right) \Big/ \sum_{k=1}^{n} \left(S_k / T_{ik}^{\lambda}\right)$. Once such

a probability is obtained, it serves as a useful index to determine the location of a new store. This model is called a gravitation model.

Nakanishi and Cooper (1974) generalize this gravitational model to include other properties of stores by suggesting techniques to estimate the attractiveness of stores, and since then retail location problems have been studied extensively. The gravitational models and, more generally, spatial interactions models receive more interest as more data and global information system technologies are becoming available (Birkin et al. 2002).

Prediction of the Number of Customers Visiting a Store

The number of customers visiting a retail store will influence the sales volume of individual goods at the store. If management can predict with a fair degree of accuracy the future number of customers, be it 1 day, a week or a month in advance, it then becomes possible to map out an efficient operations program. Regular and part-time employees can be scheduled effectively to reduce operating costs and improve customer service. On the product side, stock orders can be gauged more accurately, reducing the risk of lost sales due to shortages and wastage due to an overstocking of goods.

The number of customers is usually affected by the day of the week, weather, temperature, sales campaigns and economic conditions. Multiple regression analysis can be used to arrive at a formula that explains the volume and distribution of customers; see Chatterjee and Price (1991) for details of statistical methods. By using new data, the formula can be updated on a daily or weekly basis, or when the difference between the predicted number and the actual number falls out of a predetermined range.

Classification of Goods by Sales Volume

In an average size supermarket, several thousands of goods are displayed. They can be divided into three classes according to sales (or gross sales). This is known as ABC analysis or the Pareto chart. Class *A* goods, while accounting for around 50% of sales, constitute only a small portion of the entire range of goods, usually about 10%. Class *B* comprise approximately 40% of both the range of goods and sales value. Class *C* goods comprise about half of the goods available but only around 10% of the sales.

Empirical studies have shown that the sales of Class *C* goods reflect a Poisson-type distribution, while that of Class *B* goods is represented by a normal or log-normal distribution. The sales of Class *A* goods are well explained by regression analysis using causal models (including price and sales campaigns).

Retailers, wholesalers and producers will be able to get valuable information by classifying goods in this way and extend their understanding of their characteristics as represented by the set of parameters. This information can then be used to make appropriate choices for items in the store. For Class *A* goods, the effects of price and sales campaigns in the regression formula provide useful information for strategic management. In addition, as will be discussed below, goods procurement and inventories will be more efficiently controlled.

Product Assortment and Shelf-Space Allocation

One of the primary concerns of retail management involves determining the variety of products to offer, and the allocation of limited shelf-space among the selected products so as to maximize the store's profit. Most of the shelf-space research indicates that the proportion of total product shelf-space received by a particular product is important since it influences the brand's aggregate sale and market share.

Anderson and Amato (1974) develop a model for simultaneous decision making for a brand assortment and shelf-space allocation problem, assuming the shelf-space large enough to contain at least one facing of each available brand. Let B and S, respectively, denote a set consisting of all the available brands and a subset of B. In addition, let n_b ($b \in S$) signify the number of facings of brand b to be displayed. For a subset S of B, seeking an optimal value for n_b is a knapsack problem. They also provide a method for obtaining an optimal set of brands to be displayed.

However, their model does not take into account the cross-effect among products within the store. For this reason, Corstjens and Doyle (1981) develop a more

comprehensive model, considering the main effects and the cross-effect of demand with the cost effects, which is a generalized geometric programming problem. Hansen et al. (2010), furthermore, propose a retail shelf-space decision model that incorporates a nonlinear profit function, vertical and horizontal location effects, and product cross-elasticity together with comprehensive survey of shelf-space decision models.

Inventory Control

Future sales can be estimated with a certain degree of accuracy based on the results of the type of data analysis described before. The estimated values relate to the buying-in amount and the inventory of goods. It is especially important to predict and control the inventories of non-durable goods delivered daily, such as fresh foods and dairy products.

A basic inventory policy for a retail store can be described as follows. Let Z denote the remaining amount of a commodity at the store's closing time, of which a part, D, is to be scrapped due to expiration. Then, the stock at the end of the day is $U = Z - D$. If the predicted sales for the following day is Y, and the leadtime for buying-in is one night, then the order volume, P, is determined by $P = \max\{Y - U + \alpha, 0\}$, where α corresponds to the safety stock, the slack which prevents opportunity loss. The safety stock α relates directly to the trade-off between the probability of loss and the holding cost. An inventory simulation can be applied using past data to estimate an adequate α. It should be noted that the classical inventory control problem where the remaining amount Z should always be disposed, i.e., $D = Z$ or $U = 0$, is called a newsboy problem.

Inventory control for a retail store is significantly relevant to that for a warehouse as well as a manufacturer in the context of supply chain management.

Analysis of Movement of Customers

Although a point-of-sales (POS) record of a customer tells what kinds of goods were purchased, his or her movements through the store are not clear from the record alone. However, by comparing the layout of the store and the POS record, it becomes possible to deduce the route taken via a 'traveling salesman' scenario. By superimposing the solutions for a given number of customers, the congestion likely in each pathway of the store can be estimated. In addition, changing the distance table to correspond with the new assignment makes it much easier to analyze the effects of display changes on congestion. Traditional methods, such as the use of video and first-hand observations, are less efficient in terms of cost and precision. Using analysis such as that described above will result in less dead corners and fewer busy corners.

Pricing Strategies for Consumer Sales Promotion

Since the mid-1980s, consumer sales promotion has particularly been emphasized in retailing due to severe competition. Under these circumstances, retailers have actively introduced a variety of scientific models, mainly through software packages, for pricing strategies for consumer sales promotion with a view to attracting more customers to their shops.

Research on pricing strategies has been conducted by economists, marketing scientists, and operations researchers from a wide range of perspectives. Eliasberg and Steinberg (1993) present a comprehensive survey on this topic, and Nagle et al. (2002) develop excellent overview of decision makings for pricing management.

Markdown

Markdown pricing is one of the simplest and most popular pricing strategies for consumer sales promotion. Retailers of fresh foods use markdown pricing to sell out excess inventory before their expiration date. Retailers of apparel and seasonal goods likewise rely on markdown pricing. When goods have low salvage values once the sales season is over, retailers have incentive to sell the remaining goods while they can even at a low price due to their low salvage values.

Markdown pricing is advantageous when retailers are uncertain which products will be popular with customers. Retailers set high prices for all items initially to identify which products are popular for which customers have high reservation prices since

popular products sell out at the high initial price. The retailers then notice the remaining products as low-reservation price products and mark them down. These observations reveal that markdown pricing is a learning process of the market for retailers to sell a new item (Lazear 1986). Markdown pricing can also be interpreted as a segmentation mechanism since customers purchasing early have higher willingness to pay (Talluri and van Ryzin 1994). Heching et al. (2002) show significant improvements in revenues using model-based markdown optimization.

Decisions in markdown pricing involve optimization of the timing and magnitude of a markdown. It is essential for these decision models to identify a demand function, which explains the relationship between the price and the demand quantity. Usually, retailers themselves or related software packages seek a suitable form for the demand function by fitting historical POS data involving inventory transitions. Regression analysis will be effective when a linear demand function can be assumed.

Dynamic Pricing

Dynamic pricing is a flexible and efficient extension of markdown pricing. Under a dynamic pricing strategy, retailers of fashion and seasonal products, for example, reduce the selling price of items gradationally over time according to customers' reservation prices. Compared with markdown pricing described above, dynamic pricing is advantageous since it can explore the market in more detail with more accurate segmentation. However, dynamic pricing does not always introduce cost reduction, as airlines often raise prices over time.

There is an extensive literature on dynamic pricing. Bitran and Caldentey (2003) and Elmaghraby and Keskinocak (2003) provide an excellent survey on dynamic pricing in academic fields. Talluri and van Ryzin (2004) extensively discuss dynamic pricing within the framework of revenue management.

In the research literature, various models have been proposed for dynamic pricing from a wide range of perspectives. They are, first of all, categorized according to the level of competition of retail stores, monopoly, duopoly, oligopoly or perfect-competition. They are also classified depending on the population size, finite or infinite. Most important for theoretical understanding of dynamic pricing is a model of how demand responds to changes in price. Demand models can be for individual customers or for more aggregate classes, and can also be classified according to whether they are continuous or discrete, deterministic or stochastic, static or dynamic, without replenishment or with replenishment, etc. (Talluri and van Ryzin 2004). For example, a logit model or a discrete choice model (see Anderson et al. 1995) is a useful tool for describing demands by individual consumers under stochastic demand.

See

- ▶ Data Mining
- ▶ Facility Location
- ▶ Geometric Programming
- ▶ Inventory Modeling
- ▶ Knapsack Problem
- ▶ Newsboy Problem
- ▶ Regression Analysis
- ▶ Revenue Management
- ▶ Supply Chain Management
- ▶ Traveling Salesman Problem

References

Anderson, E. E., & Amato, H. N. (1974). A mathematical model for simultaneously determining the optimal brand-collection and display-area allocation. *Operations Research, 22*, 13–21.

Anderson, S. P., de Palma, A., & Thisse, J.-F. (1992). *Discrete choice theory of product differentiation*. Cambridge, MA: MIT Press.

Birkin, M., Clarke, G., & Clarke, M. (2002). *Retail geography and intelligent network planning*. Chichester: John Wiley.

Bitran, G., & Caldentey, R. (2003). An overview of pricing models for revenue management. *Manufacturing and Service Operations Management, 5*, 203–229.

Chatterjee, S., & Price, B. (1991). *Regression analysis by example* (2nd ed.). New York: John Wiley.

Corstjens, M., & Doyle, P. (1981). A model for optimizing retail space allocations. *Management Science, 27*, 822–833.

Eliashberg, J., & Steinberg, R. (1993). Marketing-production joint decision making. In J. Eliasberg & G. L. Lilien (Eds.), *Handbook in operations research and management science. Vol. 5: Marketing*. Amsterdam: North Holland.

Elmaghraby, W., & Keskinocak, P. (2003). Dynamic pricing in the presence of inventory considerations: Research overview, current practices, and future directions. *Management Science, 49*, 1287–1309.

Hansen, J. M., Raut, S., & Swami, S. (2010). Retail shelf allocation: A comparative analysis of heuristic and meta-heuristic approaches. *Journal of Retailing, 86*, 94–105.

Heching, A., Gallego, G., & van Ryzin, G. J. (2002). Mark-down pricing: An empirical analysis of policies and revenue potential at one apparel retailer. *Journal of Revenue and Pricing Management, 1*, 139–160.
Huff, D. L. (1963). A probabilistic analysis of shopping trade areas. *Land Economics, 39*, 81–90.
Kotler, P. (1984). *Marketing management: Analysis, planning, and control* (5th ed.). Englewood Cliffs: Prentice-Hall.
Lazear, L. (1986). Retail pricing and clearance sales. *American Economic Review, 76*, 14–32.
Nagle, T. T., Hogan, J., & Zale, J. (2002). *The strategy and tactics of pricing: A guide to profitable decision making*. New Jersey: Prentice Hall.
Nakanishi, M., & Cooper, L. G. (1974). Parameter estimate for multiplicative interactive choice model: Least square approach. *Journal of Marketing Research, 11*, 303–311.
Talluri, K. T., & van Ryzin, G. J. (2004). *The theory and practice of revenue management*. New York: Kluwer.

Revenue Equivalence Theorem

Revenue equivalence theorems in bidding theory establish conditions under which the expected revenue from various auction types (e.g., standard sealed bidding sales and progressive oral auctions) is the same.

See

► Bidding Models

Revenue Management

Costis Maglaras
Columbia University, New York, NY, USA

Introduction

Revenue management focuses on how a firm should set and update pricing and product availability decisions across its various selling channels in order to maximize its profitability. A familiar example comes from the airline industry, where tickets for the same flight may be sold at many different fares, the availability of which is changing as a function of purchase restrictions, the forecasted future demand, and the number of unsold seats. Indeed, the airline deregulation act in the late 1970s motivated the rapid development and deployment of revenue management tools to manage the sales process of airline tickets, and such systems have been adopted and have transformed the transportation and hospitality industries, and is increasingly important in retail, telecommunications, entertainment, financial services, health care and manufacturing. In parallel, pricing and revenue optimization has become a rapidly expanding practice in consulting services, and a growing area of software and information technology (IT) development.

Revenue management, or yield management, as it was originally called focused on tactical optimization of capacity allocation decisions. This was motivated by the airline industry, where ticket prices – or fare classes – were determined early in the sales horizon, but where airlines could tactically decide which fare classes to make available to consumers at different points in time. The implementation of such systems was facilitated from the presence of industry-wide electronic reservation systems that allowed airlines to "push" their capacity allocation decisions to the travel agents that used to interact with potential passengers. Similar models and systems were applied in many other industries successfully starting with hotels and car rental companies in the late 1980s and the 1990s.

In other application settings, such as retailing, instead of tactical capacity allocation decisions, revenue management systems focus on tactical pricing and markdown decisions. Specifically, in retailing, such systems are used to choose which product pricing decisions to change, either through temporary promotions or permanent markdowns, so as to optimize the overall profitability of a product, a store, or a chain of stores. The same product may be offered at different initial prices at different geographic locations, reflecting local demand conditions, and can be differentially priced throughout its sales season across stores. Moreover, the initial inventory and product assortments at each store may themselves be optimized using to a large extent inputs from such a revenue management system. Revenue management systems have been widely adopted in retailing after their introduction around 2000, and similar tactical price optimization systems have been applied in many other settings such as sports event pricing, entertainment,

telecommunications, and financial services. And while most of the above applications are typically concerned with the interface between businesses and consumers, similar techniques have been adopted in many business-to-business (B2B) settings in customized pricing of B2B transactions.

In all of the above settings, some common characteristics can be identified, such as the large scale of potential transactions; markets with imperfect competition that allows firms to at least partially discriminate potential customers via differential pricing or capacity allocation decisions; the availability of data that is used to quantify market response and estimate demand models, form demand forecasts, characterize consumer choice behavior, etc.; and the use of quantitative methods and IT systems to implement and deploy such solutions.

Consider a firm that owns a fixed capacity of a certain resource that is consumed in the process of producing or offering multiple products or services over a finite time horizon. The firm's problem is to maximize its total expected revenues by dynamically selecting the price of each of its products over time. Four variants will be considered. In the first two, the prices of the products are assumed to be fixed and the firm controls the allocation of capacity at the different price levels. The first variant will focus on the basic setting of Littlewood's problem of a low-fare and a high-fare demand class that arrive sequentially over time, and the firm controls how much capacity to protect for the high-fare class. The second variant assumes that there are more than two demand classes, and where demand requests are not sequenced in time from lower to higher fare, but instead arrive stochastically over time. In such a setting the firm needs to dynamically control which set of fare class requests to be accepted at any point in time. In the next two variants, the firm controls the pricing of the product. In one case the firm offers just one product and is assumed to be a monopolist or to operate in a market with imperfect competition, and thus to have power to influence the demand for each product by varying its price. In this setting, the firm's problem is to choose a dynamic pricing strategy for its product in order to optimize expected revenues. In the second case, the firm offers many products that consume capacity of the same scarce resource, and again has pricing power and seeks to optimize its expected revenues by choosing a multi-dimensional dynamic pricing policy. Finally, the first two are referred to as capacity control problems, and the last two as dynamic pricing problems.

The approach taken here will highlight how the last three problems can be reduced to a common formulation, thus connecting prior results that have appeared in the literature under a unified framework, and explores some of the consequences of this formulation. Specifically, it is shown that the multi-product dynamic pricing and the capacity control problems can be recast within this common framework, and be treated as different instances of a single-product pricing problem for appropriately selected concave revenue functions. Broadly speaking, this is done by decoupling the revenue maximization problems in two parts: first, at each point in time the firm selects an aggregate capacity consumption rate from all products, and second, it computes the vector of demand rates to maximize instantaneous revenues subject to the constraint that all products jointly consume capacity at the aforementioned rate. The latter is akin to the basic microeconomics problem of resource allocation subject to a budget constraint, and gives rise to an appropriate aggregate revenue rate function in each case.

Adopting this common formulation, it will be straightforward to review some of the key structural results regarding the monotonicity properties of the value function and the associated controls, which were derived in the literature (Gallego and van Ryzin 1994, 1997; Lee and Hersh 1993; Lautenbacher and Stidham 1999; Zhao and Zheng 2000; Maglaras and Meissner 2006). Subsequently, the deterministic and continuous (fluid) approximations of the dynamic pricing problems described above are presented, reviewing their solution, the pricing heuristics that can be gleaned from them, and the performance bounds that they offer for the expected revenues of the stochastic and discrete revenue maximization of original interest.

The next section formulates and solves Littlewood's two-fare-class capacity control problem, and briefly discusses its extension to multiple fare classes. After that, the single product dynamic pricing problem is formulated, some known structural results are reviewed, and the deterministic and continuous (fluid) analog of the dynamic pricing problem presented. Multi-product variants and extensions to a network setting are then each described in separate sections.

Single-Resource Capacity Control with Two Fare Classes

Model

In its simplest form, this model can be described as follows: an airline has a fixed capacity for a flight to sell to the market; there is a low-fare and a high-fare class, and low-fare demand is realized before the high-fare demand; the key decision is to select how many units of capacity to reserve for the high-fare demand (i.e., make them unavailable for the low-fare demand that gets realized first), so as to maximize the total expected revenue per flight.

A firm has C identical units of a good to sell over two time periods to two demand classes indexed by $i = 1, 2$. The class-2 demand, denoted by D_2, arrives first and pays a price of p_2, followed by the class-1 demand, denoted by D_1, which pays $p_1 > p_2$. The salvage value is assumed without loss of generality to be 0. The two demands are discrete random variables that are independent of each other, and independent of any capacity control decisions made by the system manager, drawn from some distributions F_i for $i = 1, 2$. The firm controls whether to accept or reject each class- i request for one unit of its capacity, and its objective is to allocate the available capacity to the two demand streams described above so as to maximize its total expected revenue over the entire selling horizon. It is well known that the structure of the firm's optimal policy takes the form of a threshold, or protection level, denoted by L, which sets the number of units of capacity to be reserved for the high-fare class demand, D_1, i.e., class 2 demand requests are accepted as long as it the remaining capacity left for period 1 for the high-fare demand stream is greater than L, and are rejected otherwise. In summary, the firm's problem is to choose the protection level L to maximize its expected revenue:

$$\max_{0 \le L \le C} \mathbb{E}[p_1 \min(D_1,\ \max(C - D_2, L)) + p_2 \min(D_2, C - L)]. \tag{1}$$

where the expectation is taken with respect to the two demand distributions.

Littlewood's Formula

In the capacity control problem in (1), the term $\min(D_2,\ C - L)$ is the sales for the low-fare class, which arrives first; and consequently, the high-fare class sales is the minimum of demand D_1 and the remaining number of seats $C - \min(D_2,\ C - L) = \max(C - D_2,\ L)$. If D_1 and D_2 were continuous random variables and partial sales were allowed, then the optimal protection-level L^* would be given by the following equality:

$$\begin{aligned} p_1\, \mathbb{P}(D_1 \ge L^*) &= p_2 \\ \text{if and only if } \ F_1(L^*) &= \gamma := 1 - p_2/p_1. \end{aligned} \tag{2}$$

This condition is commonly referred to as Littlewood's rule. The left hand side of the above expression equates the marginal expected revenues from an immediate sale at price $\$p_2$ versus a potential sale in the next period at the higher price $\$p_1$. For discrete demand distributions, the optimal protection level satisfies

$$\begin{aligned} p_2 < p_1\, \mathbb{P}(D_1 \ge L^*) \ \text{ and } \ p_2 \ge p_1\, \mathbb{P}(D_1 \ge L^* + 1) \\ \Leftrightarrow\ \gamma > F_1(L^* - 1) \ \text{ and } \ \gamma \le F_1(L^*), \end{aligned} \tag{3}$$

i.e., the optimal protection level is given by

$$L^* = \inf\{L\ :\ F_1(L) \ge \gamma\}. \tag{4}$$

The above expression is known as Littlewood's rule, following Littlewood's 1972 paper that formulated and solved that problem. There is a natural connection between this problem and the well-known newsvendor problem in operations management.

Littlewood's rule has been extended into multiple fare classes, still under the assumption that lower fare demand is realized before higher fare demand. Two heuristics were developed in the literature under the EMSR(a) and EMSR(b), where the acronym EMSR stands for Expected-Marginal-Seat-Revenue. Both heuristics are based on recursive reductions of the multiple fare class problem into two fare classes looking at a marginal class against all downstream fare classes. The nature of the solution is a sequence of nested protection levels, where L_1 units are protected for class 1, L_2 are protected for the two highest fare classes 1,2, etc. The optimal solution of the multiple fare class problem is given in Brumelle and McGill (1993).

Single-Product Dynamic Pricing Problem

Model

Consider a firm endowed with C units of capacity of a product that is to be sold over a finite horizon τ, and where capacity cannot be replenished up to that time. The salvage value of remaining capacity at time τ is assumed to be zero. (A constant per-unit salvage value would also result to formulations similar to those developed below.) The firm is either a monopolist or is assumed to operate in a market with imperfect competition, and, in that, has power to influence the demand for each product by varying its menu of prices. Let $p(t)$ denote the price posted at time t. The demand process is assumed to be a non-homogeneous Poisson process with rate vector λ determined through a demand function $\lambda(p(t))$, where $\lambda : \mathcal{P} \to \mathcal{L}$, $\mathcal{P} \subseteq \mathbb{R}$ is the set of feasible prices, and $\mathcal{L} = \{x \geq 0 : x = \lambda(p),\ p \in \mathcal{P}\} \subseteq \mathbb{R}_+$ is the set of achievable demand rate vectors. Assume that $\mathcal{L}$ is a convex set. For ease of exposition the demand function $\lambda(\cdot)$ is assumed to be stationary. Consider regular demand functions that satisfy some additional conditions. In the sequel, x' denotes the transpose of any matrix x, for any real number y, $y^+ := \max(0, y)$, e is the vector of ones of appropriate dimension and a.s. stands for almost surely.

Definition 1. *A demand function is said to be regular if it is a continuously differentiable, bounded function, and (a) $\lambda(p)$ is strictly decreasing in p, (b) $\lim_{p\to\infty} \lambda(p) = 0$, and (c) the revenue rate $p\lambda(p)$ is bounded for all $p \in \mathcal{P}$ and has a finite maximizer $\hat{p}$.*

Assuming there exists a continuous inverse demand function $p(\lambda)$, $p : \mathcal{L} \to \mathcal{P}$, which maps an achievable demand rate λ to the corresponding price $p(\lambda)$, the demand rate can be taken as the firm's control, from which the appropriate price can be inferred using the inverse demand function. The expected revenue rate can be expressed as a function of the vector of demand rates λ as $R(\lambda) := \lambda p(\lambda)$, and is assumed to be continuous, bounded and strictly concave.

Ex.1 *Linear demand model:* the demand for the product at price p is given by $\lambda(p) = \Lambda - bp$, where $\Lambda > 0$ is the market potential for the product and $b > 0$ is the price sensitivity parameter. The inverse demand and revenue functions are $p(\lambda) = (\Lambda - \lambda)/b$ and $R(\lambda) = \lambda(\Lambda - \lambda)/b$, respectively.

Ex. 2 *Logit demand model:* $\lambda(p) = \Lambda e^{-bp}/(1 + e^{-bp})$, $p(\lambda) = (1/b)ln(\Lambda/\lambda - 1)$ and $R(\lambda) = (\lambda/b)$ $ln(\Lambda/\lambda - 1)$, respectively.

The problem addressed is roughly described as follows: given an initial capacity C, a selling horizon τ, and a demand function that maps a posted price to a corresponding instantaneous demand rate, the firm's goal is to choose a non-anticipating dynamic pricing strategy in order to maximize its total expected revenues.

A discrete-time formulation is adopted, i.e., one where time has been discretized in small intervals of length δt, indexed by $t = 1, \ldots, T$, such that $\mathbb{P}(\text{one product request in } [0, \delta t]) = \lambda\delta t + o(\delta t)$, $\mathbb{P}(\text{two product requests in } [0, \delta t]) = \lambda^2(\delta t)^2 + o((\delta t)^2)$, and so on, where the notation $f(x)$ is of order $o(x)$ implies that $f(x)/x \to 0$ as $x \to 0$. In addition, $T = \tau/\delta t$. With slight abuse of notation, write λ in place of $\lambda\delta t$, and refer to λ either as the demand or the buying probability. The random demand in period t, denoted by $\xi(t; \lambda)$, is Bernoulli with probability $\lambda(t) = \lambda(p(t))$, i.e., $\mathbb{P}(\xi(t; \lambda) = 1) = \lambda(p(t))$ and $\mathbb{P}(\xi(t) = 0) = 1 - \lambda(p(t))$. Treating the demand rate λ as the control variables (prices are inferred via the inverse demand relationship), the discrete-time formulation of the dynamic pricing problem of Gallego and van Ryzin (1994) is:

$$\max_{\{\lambda(t),\, t=1,\ldots,T\}} \left\{ \mathbb{E}\left[\sum_{t=1}^{T} p(\lambda(t))\xi(t; \lambda)\right] : \sum_{t=1}^{T} \xi(t; \lambda) \leq C \text{ a.s. and } \lambda(t) \in \mathcal{L}\ \forall t \right\}. \tag{5}$$

Analysis of the Dynamic Program

Let x denote the number of remaining units of capacity at the beginning of period t, and $V(x, t)$ be the expected revenue-to-go starting at time t with x units of capacity left. Then the Bellman equation associated with (5) is:

$$V(x, t) = \max_{\lambda \in \mathcal{L}} \{\lambda[p(\lambda) + V(x - 1, t + 1)] + (1 - \lambda)\, V(x, t + 1)\}, \tag{6}$$

with the boundary conditions

$$V(x, T + 1) = 0\ \forall x \quad \text{and} \quad V(0, t) = 0\ \forall t. \tag{7}$$

Letting $\Delta V(x,t) = V(x,t+1) - V(x-1,t+1)$ denote the marginal value of one unit of capacity as a function of the state (x,t), (6) can be rewritten as

$$V(x,t) = \max_{\lambda \in \mathcal{L}} \{R(\lambda) - \lambda \Delta V(x,t)\} + V(x,t+1), \quad (8)$$

and the optimal control $\lambda^*(x,t)$ is given by

$$\lambda^*(x,t) = \operatorname*{argmax}_{\lambda \in \mathcal{L}} \{R(\lambda) - \lambda \Delta V(x,t)\},$$

where $R(\cdot)$ is a concave increasing revenue function. Using the properties of $R(\cdot)$, one can show that $\lambda^*(x,t)$ is decreasing in $\Delta V(x,t)$, which using a backwards induction argument in t gives that $\Delta V(x,t)$ is decreasing in x and t. These monotonicity results are the key structural properties that one can extract from the dynamic program and are summarized in the following results.

Proposition 1. *(Talluri and van Ryzin 2004b, Prop. 5.2 Ch. 4) For the problem defined in (5):*

1. $\lambda^*(x,t)$ is decreasing in the marginal value of capacity $\Delta V(x,t)$, and
2. $\Delta V(x,t)$ is decreasing in x and t.

The dynamic program in (6) and (7) admits a closed-form solution for the special case of the exponential demand model, and is fairly easy to solve numerically in other cases. The optimal policy takes the form of a two-dimensional table that specifies a price for each (remaining inventory, time) pair. The optimal price path continuously decreases price between sales, and jumps up at every sales epoch.

The Fluid Model

The fluid model has deterministic and continuous dynamics, and in broad terms is obtained by replacing the Poisson demand process with non-homogeneous rate $\lambda(t)$, where demand requests arrive stochastically over time and require discrete units of capacity, by a deterministic and continuous process where demand for the product arrive continuously at the deterministic rate $\lambda(t)$. The resulting inventory dynamics (in discrete time) are given by

$$X(t+1) = X(t) - \lambda(t), \;\; x(0) = C, \;\; X(T) \geq 0.$$

This deterministic analog is a simplification of the discrete and stochastic model from the previous subsections. It can be rigorously justified as a limit under a law-of-large-numbers type of scaling as the potential demand and the capacity grow proportionally large, and, as such, one would expect to provide more useful analysis and policy recommendations in settings where the firm has many units to sell and operates in a market with high demand. For example, one would expect that the discrete and stochastic nature of the pricing problem to be relevant when selling four newly constructed single family homes over the course of 24 weeks, but it may be less critical when selling 4000 pairs of skis over a similar time duration from, say, October to March.

The fluid model formulation of the dynamic pricing problem is one where the firm selects a demand rate $\lambda(t)$ (or a price) at each time t to:

$$\max_{\{\lambda(t),\, t=1,\ldots,T\}} \left\{ \sum_{t=1}^{T} R(\lambda(t))dt \;:\; \sum_{t=1}^{T} \lambda(t)dt \leq C \text{ and } \lambda(t) \in \mathcal{L} \; \forall t \right\}. \quad (9)$$

Optimal policy for fluid pricing problem. An important result derived in Gallego and van Ryzin (1994) is that a constant price (and thus a constant demand rate) is optimal for (9). Specifically, let $\hat{\lambda} = \operatorname{argmax}\{R(\lambda) : \lambda \in \mathcal{L}\}$ and $\hat{p} = p(\hat{\lambda})$ be the demand rate and price that maximize the revenue rate disregarding any capacity considerations, respectively. Also, let $\lambda^0 = C/T$ be the run-out rate that depletes capacity at time T, and $p^0 = p(\lambda^0)$.

Proposition 2. *(Talluri and van Ryzin 2004b, § 5.2.1.2) The optimal solution for (9), denoted by $\bar{\lambda}$ and $\bar{p}$, are given by*

$$\bar{\lambda}(x,\, t) = \min(\lambda^0,\; \hat{\lambda}), \;\; \bar{p} = \max(p^0,\; \hat{p}), \quad t = 1, \ldots, T. \quad (10)$$

Intuitively, the firm uses the revenue-maximizing price $\hat{p}$ unless this would deplete the capacity too soon, in which case it increases its unit price to p^0 and sells its capacity by time T, while accruing higher total revenues. The proof is simple and exploits the structure of (9) that seeks to maximize a concave function over the capacity constraint; the first-order

optimality conditions set the marginal revenue rates at each time t to be equal, which in turn is achieved via a static pricing policy.

The static nature of the optimal policy for the fluid control problem is simple and intuitive, but also lacks the capability of corrective action against stochastic fluctuations. This does not arise in the fluid formulation, where the capacity is drained along the optimal deterministic trajectory, but it is relevant for the stochastic problems of original interest. Alternatively, $\bar{\lambda}$ can also be expressed in feedback form. Note that the deterministic trajectory of the fluid model is such that $x/(T-t) = C/T$ for all t if $\hat{\lambda} \geq C/T$, and $x/(T-t) = (C-\hat{\lambda}t)/(T-t) \geq C/T$ if $\hat{\lambda} < C/T$. Given this observation, $\bar{\lambda}$ can be expressed as

$$\bar{\lambda}(x,t) = \min\left(\hat{\lambda}, \frac{x}{T-t}\right). \tag{11}$$

Equation (11) illustrates that the optimal pricing policy extracted from the fluid model is essentially a tracking or feedback policy that continuously re-optimizes its decision so as to achieve a sales rate that is given by the minimum between the capacity unconstrained revenue maximizing rate and the rate that would deplete all inventory exactly at time T. This interpretation motivates the common practical heuristic that periodically resolves the fluid model so as to adjust its prevailing price, which according to (11) is exactly what the fluid model prescribes; it is worth noting that the resolving heuristic is one of the most practical and widely adopted approaches to price optimization.

An upper bound on achievable performance. Apart from good policy recommendations, the fluid model offers a useful upper bound on the achievable expected revenue in the underlying stochastic and discrete problem in (5). Specifically, Gallego and van Ryzin (1994) showed the following:

Proposition 3. *(Talluri and van Ryzin 2004b, §5.2.2.3)* $V(C,1) \leq (\bar{\lambda}\bar{p})T$.

This result establishes a tractable limit for the best achievable performance that is useful in establishing sub-optimality gaps for heuristics that one may wish to use, and to prove asymptotic optimality results of candidate policies.

Multiple Products, Single Resource

This section studies multi-product dynamic pricing and capacity control problems.

Problem Formulations

Dynamic pricing problem. The basic elements of the problem as similar to those in the previous section, with the key difference that the firm is now selling multiple products or services, indexed by $i = 1, \ldots, n$ that consume the capacity C. Each product i request requires one unit of capacity. Let $p(t) = [p_1(t), \ldots, p_n(t)]$ denote the vector of prices at time t. The demand process is assumed to be n -dimensional non-homogeneous Poisson process with rate vector λ determined through a demand function $\lambda(p(t))$, where $\lambda : \mathcal{P} \to \mathcal{L}$, $\mathcal{P} \subseteq \mathbb{R}^n$ is the set of feasible price vectors, and $\mathcal{L} = \{x \geq 0 : x = \lambda(p),\ p \in \mathcal{P}\} \subseteq \mathbb{R}^n_+$ is the set of achievable demand rate vectors, assumed to be a convex set. As in the previous section, the demand function $\lambda(\cdot)$ is assumed to be stationary. The definition of regular demand functions is as follows:

Definition 2. *A demand function is said to be regular if it is a continuously differentiable, bounded function, and (a) for each product i,* $\lambda_i(p)$ *is strictly decreasing in* p_i*, (b)* $\lim_{p_i \to \infty} \lambda_i(p) = 0$ *(i.e., consumers have bounded wealth), and (c) the revenue rate* $p'\lambda(p) = \sum_{i=1}^n p_i \lambda_i(p)$ *is bounded for all* $p \in \mathcal{P}$ *and has a finite maximizer* $\bar{p}$.

Assuming there exists a continuous inverse demand function $p(\lambda)$, $p : \mathcal{L} \to \mathcal{P}$, which maps an achievable vector of demand rates λ into a corresponding vector of prices $p(\lambda)$, allows one to view the demand rate vector as the firm's control, and infer the appropriate prices using the inverse demand function. The expected revenue rate can be expressed as a function of the vector of demand rates λ as $R(\lambda) := \lambda' p(\lambda)$, and is assumed to be continuous, bounded and strictly concave.

Ex. 1 *Linear demand model:* the demand for product i is given by

$$\lambda_i(p) = \Lambda_i - b_{ii}p_i - \sum_{j \neq i} b_{ij}p_j,$$

$$\text{or (in vector form) } \lambda(p) = \Lambda - Bp,$$

where Λ_i is the market potential for product i and b_{ii}, b_{ij} are the price and cross-price sensitivity parameters. The inverse demand and revenue functions are $p(\lambda) = B^{-1}(\Lambda - \lambda)$ and $R(\lambda) = \lambda' B^{-1}(\Lambda - \lambda)$, respectively. Definition 1 requires that $b_{ii} > 0$ for all i. To ensure that the inverse demand function is well defined and the revenue function is concave, it is required that either $b_{ii} > \sum_{j \neq i} |b_{ji}|$ or $b_{ii} > \sum_{j \neq i} |b_{ij}|$ for all i; both conditions guarantee that B is invertible and that its eigenvalues have positive real parts (Horn and Johnson 1994, Thm. 6.1.10). The linear demand model has the obvious shortcoming that for some prices it may generate negative demand values. This could be corrected by taking $\lambda(p) = (\Lambda - Bp)^+$, but such a change does not preserve the concavity of the revenue function. In practical applications, it is appropriate to use the linear demand model in settings where the set of reasonable prices would ensure that the demand rates are positive, or in applications where the pricing manager can constrain the prices so as to ensure this property.

Ex. 2 *Multinomial Logit (MNL) model:* Potential customers arrive with rate Λ and have utilities for each product i given by $v_i - p_i + \xi_i$, where v_i is the deterministic portion that is common to all customers, p_i is the price, and ξ_i is the random term (that differentiates customers) that is drawn from a Gumbel distribution with mean zero and parameter one (the latter is assumed w.l.o.g.), and is IID across products and customers. The no-purchase option has utility $u_0 + \xi_0$, ξ_0 is IID with the ξ_i's and $u_0 = 0$. The demand rates for product i is given by

$$\lambda_i(p) = \Lambda \frac{e^{v_i - p_i}}{1 + \sum_j e^{v_j - p_j}}.$$

Adopting again a discrete-time formulation, the random demand vector in each period t, denoted by $\xi(t;\lambda)$, is Bernoulli with probabilities $\lambda(t) = \lambda(p(t))$, and $\mathbb{P}(\xi_i(t;\lambda) = 1) = \lambda_i(p(t))$ and $\mathbb{P}(\xi_i(t;\lambda) = 0) = 1 - \lambda_i(p(t))$ for all i. Treating the demand rates λ_i as the control variables (prices are inferred via the inverse demand relationship), the discrete-time formulation of the dynamic pricing problem of Gallego and van Ryzin (1997) is:

$$\max_{\{\lambda(t),\, t=1,\ldots,T\}} \left\{ \mathbb{E}\left[\sum_{t=1}^{T} p(\lambda(t))' \xi(t;\lambda)\right] : \sum_{t=1}^{T} e' \xi(t;\lambda) \leq C \text{ a.s. and } \lambda(t) \in \mathcal{L}\ \forall t \right\}. \tag{12}$$

Capacity control problem. The next variant considered is the one studied by Lee and Hersh (1993), where the price vector p and the demand rate vector $\lambda = \lambda(p)$ are fixed, and the firm optimizes over capacity allocation decisions. For this problem and without any loss of generality it is assumed that products are labelled such that $p_1 \geq p_2 \geq \cdots \geq p_n$. The firm has discretion as to which product requests to accept at any given time. This is modeled through the control $u_i(t)$ that is equal to the probability of accepting a product i request at time t. It is customary to assume that the firm is "opening" or "closing" products, thus considering controls $u_i(\cdot)$ that are 0 or 1, but this need not be imposed as a restriction. The dynamic capacity control problem is the following:

$$\max_{\{u(t).t=1,\ldots,T\}} \left\{ \mathbb{E}\left[\sum_{t=1}^{T} p' \xi(t;u\lambda)\right] : \sum_{t=1}^{T} e' \xi(t;u\lambda) \leq C \text{ a.s. } 1 \text{ and } u_i(t) \in [0,1]\ \forall t \right\}, \tag{13}$$

where $u\lambda$ above denotes the vector with coordinates $u_i\lambda_i$.

The remainder of this section describes how to reduce (12) and (13) into dynamic optimization problems where the control is the (one-dimensional) aggregate capacity consumption rate. The reduced problems can be studied through a unified analysis.

A Common Formulation in Terms of the Aggregate Capacity Consumption

Dynamic pricing problem. Let x denote the number of remaining units of capacity at the beginning of period t, and $V(x,t)$ be the expected revenue-to-go

starting at time t with x units of capacity left. Then, the Bellman equation associated with (5) is:

$$V(x,t) = \max_{\lambda \in \mathcal{L}} \left\{ \sum_{i=1}^{n} \lambda_i [p_i(\lambda) + V(x-1,t+1)] + (1 - e'\lambda)\, V(x,t+1) \right\}, \tag{14}$$

with the boundary conditions

$$V(x,T+1) = 0 \ \forall x \quad \text{and} \quad V(0,t) = 0 \ \forall t. \tag{15}$$

Letting $\Delta V(x,t) = V(x,t+1) - V(x-1,t+1)$ denote the marginal value of one unit of capacity as a function of the state $(x,\ t)$, (14) can be rewritten as

$$\begin{aligned} V(x,t) &= \max_{\lambda \in \mathcal{L}} \left\{ R(\lambda) - \textstyle\sum_{i=1}^{n} \lambda_i \Delta V(x,t) \right\} \\ &\quad + V(x,t+1) \\ &= \max_{\rho \in \mathcal{R}} \{ R^r(\rho) - \rho \Delta V(x,t) \} \\ &\quad + V(x,t+1), \end{aligned} \tag{16}$$

where $\rho := \sum_{i=1}^{n} \lambda_i$ is the aggregate rate of capacity consumption, $\mathcal{R} := \left\{ \rho : \sum_{i=1}^{n} \lambda_i = \rho,\ \lambda \in \mathcal{L} \right\}$ is the set of achievable capacity consumption rates, and

$$R^r(\rho) := \max_{\lambda} \left\{ R(\lambda) : \sum_{i=1}^{n} \lambda_i = \rho,\ \lambda \in \mathcal{L} \right\} \tag{17}$$

is the maximum achievable revenue rate subject to the constraint that all products jointly consume capacity at a rate ρ. Note that (17) is a concave maximization problem over a convex set, and its solution is readily computable, often in closed form. The aggregate revenue function $R^r(\cdot)$ is concave and satisfies the conditions of Definition 1. The optimal vector of demand rates, denoted by $\lambda^r(\rho)$, is unique and continuous in ρ.

Ex. 1 *Linear demand model:* For $\lambda(p) = \Lambda - Bp$, the associated aggregate revenue function $R^r(\rho)$ defined through (17) can be expressed as

$$R^r(\rho) = -\alpha_i \rho^2 + \beta_i \rho + \gamma_i \quad \text{for} \quad \rho \in [r_{i-1}, r_i),$$

for $0 = r_0 \le r_1 \le r_2 \le \cdots \le r_I$, and constants $(\alpha_i, \beta_i, \gamma_i)$ and r_i that depend on the model parameters Λ, B, μ, and are such that $R^r(\rho)$ is continuous, almost everywhere differentiable, and increasing for all $\rho \le \hat{\rho} := \text{argmax}_{\rho} R^r(\rho)$. The value of r_{i-1} is that of the smallest capacity consumption rate above which it is optimal to start offering the i most profitable products. The derivation of the constants $(\alpha, \beta, \gamma, r)$ can be found in the Appendix of Maglaras (2005).

Ex. 2 For the MNL model, straightforward manipulations show that

$$R(\rho) = \rho \ln\left(\sum_j e^{v_j} \right) - \rho \ln(\rho/(\Lambda - \rho)),$$

$$\lambda_i(\rho) = \rho \frac{e^{v_i}}{\sum_j e^{v_j}} \quad \text{and} \quad p_i(\rho) = \ln\left(\textstyle\sum_j e^{v_j} \right) - \ln(\rho/(\Lambda - \rho)).$$

Proposition 4. *The dynamic pricing problem (12) can be reduced to the dynamic program (15)/(16) expressed in terms of the aggregate consumption rate. In particular, if $\rho^*(x,t)$ denotes the associated optimal control and $\lambda^*(x,t)$ and $p^*(x,t)$ denote the respective optimal demand rate and price vectors associated with (12), then, $\lambda^*(x,t) = \lambda^r(\rho^*(x,t))$ and $p^*(x,t) = p(\lambda^r(\rho^*(x,t)))$.*

The capacity control problem. Similarly, the Bellman equation associated with (13) is

$$V(x,t) = \max_{u_i \in [0,\ 1]} \left\{ \sum_{i=1}^{n} \lambda_i u_i [p_i + V(x-1,t+1)] + (1 - u'\lambda)\, V(x,t+1) \right\} \tag{18}$$

with the boundary condition (15), which using the marginal value of capacity ΔV becomes

$$\begin{aligned} V(x,t) &= \max_{u_i \in [0,\ 1]} \left\{ \textstyle\sum_{i=1}^{n} \lambda_i u_i p_i - u'\lambda \Delta V(x,t) \right\} \\ &\quad + V(x,t+1) \\ &= \max_{0 \le \rho \le \sum_{i=1}^{n} \lambda_i} \{ R^a(\rho) - \rho \Delta V(x,t) \} \\ &\quad + V(x,t+1), \end{aligned} \tag{19}$$

where $\rho = u'\lambda$ and $R^a(\rho) = \max_u \{\sum_{i=1}^n u_i\lambda_i p_i : u'\lambda = \rho,\ u_i \in [0,\ 1]\}$ is the maximum revenue rate when the capacity is consumed at rate equal to ρ, and $u^a(\rho)$ is the corresponding control.

Proposition 5. *The capacity control problem (13) can be reduced to the dynamic program (15)/(19) expressed in terms of the aggregate consumption rate ρ. In particular, if $\rho^*(x,t)$ denotes the optimal solution of (15) and (19) and $u^*(x,t)$ denote the optimal policy for (13), then $u^*(x,t) = u^a(\rho^*(x,t))$.*

A similar result was derived by Talluri and van Ryzin (2004a) for a capacity control problem for a model with customer choice.

A Unified Analysis of the Pricing and Capacity Control Problems

The preceding analysis illustrates that both problems can be reduced to appropriate single-product pricing problems, highlighting their common structure and enabling a unified treatment. As a starting observation, for both (16) and (19), the optimal control $\rho^*(x,t)$ is computed from

$$\rho^*(x,t) = \underset{\rho\in\mathcal{R}}{\text{argmax}}\ \{R(\rho) - \rho\Delta V(x,t)\}.,$$

where $R(\cdot)$ is a concave increasing revenue function. Using Proposition 1 and the properties of $R(\cdot)$, one gets that $\rho^*(x,t)$ is decreasing in $\Delta V(x,t)$, which using a backwards induction argument in t gives that $\Delta V(x,t)$ is decreasing in x and t.

Structural results for the pricing and capacity allocation policies follow from the properties of R^r, λ^r and R^a, u^a, respectively. For example, consider the pricing problem for the case where the products are non-substitutes, i.e., the demand for product i is only a function of the price for that product p_i. In that case, the Lagrangian associated with (17) is $L(\lambda,x,y) = R(\lambda) + x(\rho - \sum_{i=1}^n \lambda_i) - y'\lambda$, with first order conditions given by $\partial R(\lambda)/\partial\lambda_i = x + y_i$, for some $x \geq 0$ and $y_i \leq 0$ with $y_i = 0$ if $\lambda_i > 0$. It is easy to show that x is decreasing in ρ (i.e., the shadow price for the capacity consumption constraint decreases as ρ increases), and that $\lambda_i^r(\rho)$ is decreasing in x.

Corollary 1. *Consider the problem specified in (12) and further assume that the products are non-substitutes, i.e., $\lambda_i(p) = \lambda_i(p_i)$ for all i. Then, $\lambda_i^*(x,t)$ is non-decreasing in $\rho^*(x,t)$ (and non-increasing in $\Delta V(x,t)$).*

A similar result can be obtained when products are substitutable provided that the demand model satisfies certain conditions analogous to those of the sensitivity matrix B of the linear model described earlier in this section.

For the capacity control problem, it is easy to recover some well-known structural properties of the optimal policy (e.g., Lee and Hersh (1993)). The derivation based on the capacity consumption rate offers new intuition as to why they hold. Specifically, $R^a(\cdot)$ is a knapsack solution for which

$$\begin{aligned} R^a(\rho) &= \min_i\ c_i + p_i, \\ u_k^a(\rho) &= \min\left(\frac{(\rho - \sum_{i<k}\lambda_i)^+}{\lambda_k}, 1\right), \end{aligned} \tag{20}$$

where $c_1 = 0$ and $c_i = \sum_{k<i}\lambda_k(p_k - p_i)$, and for any $x \in R$, $x^+ := max(x,0)$, and the optimal control $\rho^*(x,t)$ reduces to the solution to $\max\{\min_i c_i + (p_i - \Delta V(x,t))\rho\ :\ 0 \leq \rho \leq \sum_{i=1}^n \lambda_i\}$. Let $i^*(x,t) = \max\ \{i \geq 1 : p_i \geq \Delta V(x,t)\}$. Then, by inspecting the form of the piecewise linear objective function involved in the calculation of $\rho^*(x,t)$, it follows that $\rho^*(x,t) = \sum_{i\leq i^*(x,t)}^n \lambda_i$, i.e., the solution is "bang-bang" in the sense that the form of the optimal control is such that $u_i^*(x,t)$ is 0 if $i > i^*(x,\ t)$ and 1 if $i \leq i^*(x,\ t)$. In addition, from Proposition 1 part 1, $i^*(x,t)$ is decreasing in the marginal value of capacity $\Delta V(x,t)$. Therefore:

Corollary 2. *For the capacity control problem (13) or equivalently, (15)/(19), the optimal allocation policy is nested, in that $u_i^*(x,t) = 1$ if $i \leq i^*(x,t)$, and $u_i^*(x,t) = 0$ otherwise, and $i^*(x,t)$ is decreasing in the marginal value of capacity $\Delta V(x,t)$.*

Efficient frontier. The subproblem of computing the optimal revenue subject to a constraint on the aggregate capacity consumption rate specified in (17) and (20) defines an efficient frontier $(\rho,\ R^r(\rho))$ and $(\rho,\ R^a(\rho))$ for the dynamic pricing and capacity allocation problems, respectively. As in the context of portfolio optimization, the efficient frontier provides a systematic framework for comparing different policies and highlights the structure of the respective optimal controls. Some of the direct insights extracted form the efficient frontier calculation is that the optimal

capacity control policy in a multi-product setting is nested, and that the optimal dynamic pricing policy is the solution of a problem that strives to equalize the marginal revenue rates across products.

It may also lead to computational improvements if this subproblem can be solved efficiently, which is indeed the case for some common demand models such as the linear and the multinomial logit. The preceding discussion is from Maglaras and Meissner (2006); see also Feng and Xiao (2000, 2004) and Talluri and van Ryzin (2004a). Finally, note that the structure of the dynamic programs studied in this section has been observed in other papers, such as Lin et al. (2003) and their study of single-resource capacity control problems where each arrival may request multiple units of capacity, and Vulcano et al. (2002) and their analysis of optimal dynamic auctions. The latter involves an analysis of a discrete-time, batch demand analog to the dynamic program studied here.

Solution to the Deterministic Multi-product Pricing Problem

As before, the fluid model has deterministic and continuous dynamics, and is obtained by replacing the discrete stochastic demand process by its rate, which now evolves as a continuous process. The realized instantaneous demand for product i at time t is deterministic and given by $\lambda_i(t)$. It is allowed for product i requests to consume capacity at a rate of $a_i > 0$ units per unit of demand, and denote by a the vector $[a_1, \ldots, a_n]$. This is a generalization of the model considered thus far that assumed uniform capacity requirements (all equal to 1). With a general capacity requirement vector a, the capacity consumption rate is defined by $\rho = a'\lambda$, and the definitions of R^r and λ^r can be appropriately adjusted to reflect that change. The system dynamics are given by $X(t+1) = X(t) - \sum_{i=1}^{n} a_i \lambda_i(t), \quad X(0) = C$, together with the boundary condition that $X(T) \geq 0$. The firm selects a demand rate $\lambda_i(t)$ (or a price) at each time t. The fluid formulation of the multi-product pricing problem is the following:

$$\max_{\{\lambda(t), t=1,\ldots,T\}} \left\{ \sum_{t=1}^{T} R(\lambda(t))dt \;:\; \sum_{t=1}^{T} a'\lambda(t)dt \leq C \text{ and } \lambda(t) \in \mathcal{L}\ \forall t \right\}. \tag{21}$$

Gallego and van Ryzin (1997, §4.5) partially extended their single product results to multiple products, but without providing such a succinct solution as the one presented in the previous section. An alternative approach that exploits the action space reduction described earlier was described in Maglaras and Meissner (2006) and is reviewed below. Specifically, recalling the definitions of the aggregate revenue function $R^r(\rho)$ and optimal demand rate vector $\lambda^r(\rho)$ in (17) adjusted for the fact that $\rho = a'\lambda$, (21) can be rewritten as:

$$\max_{\{\rho(t),\, t=1,\ldots,\, T\}} \left\{ \sum_{t=1}^{T} R^r(\rho(t))dt : \sum_{t=1}^{T} \rho(t)dt \leq C,\ \rho(t) \in R\ \forall t \right\}. \tag{22}$$

Note that (22) is identical to a single-product problem with revenue function R^r, and thus is solvable using the approach described above. Let $\rho^0 := C/T$ and $\hat{\rho} = \text{argmax}_\rho R^r(\rho)$. Then, the optimal solution to (22) is to consume capacity at a constant rate $\bar{\rho}$ given by

$$\bar{\rho}(t) := \min(\hat{\rho}, \rho^0) \quad \forall t, \tag{23}$$

the corresponding vector of demand rates is $\lambda^r(\bar{\rho})$, while the price vector is $p(\lambda^r(\bar{\rho}))$. A direct verification that this solution satisfies the optimality conditions for (21) establishes the following:

Proposition 6. *Let $\bar{\lambda}(\cdot)$ and $\bar{p}(\cdot)$ denote the optimal vectors of demand rates and prices for (21). Then, $\bar{\lambda}, \bar{p}$ are constant over time and are given by $\bar{\lambda}(t) = \lambda^r(\bar{\rho})$ and $\bar{p}(t) = p(\lambda^r(\bar{\rho}))$.*

Asymptotically Optimal Heuristics Extracted from the Deterministic Model

Finally, three heuristics for the revenue management problems studied in this section are presented. For each of these policies, one could show that they achieve the optimal asymptotic performance (Maglaras and Meissner 2006).

a. **The static pricing (make-to-order) heuristic of Gallego and van Ryzin (1997).** This policy implements the static prices $\bar{p}$ specified in Proposition 1.

b. **A List Price Capacity Control (LPCC) heuristic.** One way to implement (3) is by introducing capacity control capability on top of the static prices given in (a), specifically,
   1. price according to $\bar{p}$ and label products such that $\bar{p}_1/a_1 \geq \bar{p}_2/a_2 \geq \cdots \geq \bar{p}_n/a_n$, and
   2. compute $\bar{\rho}(x,t)$ and use the capacity controls $u_1(x,t) = 1$ if $x > 0$, $u_1(0,t) = 0$, and for $i \geq 2$,

$$u_i(x,t) = \begin{cases} 1 & \text{if } \bar{\rho}(x,t) - \sum_{j<i} a_j \bar{\lambda}_j \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

   Note that this policy can only reduce the aggregate capacity consumption rate from its nominal value of $\sum_{i=1}^{n} a_i \bar{\lambda}_i$, but can never increase it. A product is made available only if the fluid solution starting from that state would choose to sell this product in all future time periods, and closes the product if the fluid solution would dictate only partial acceptance of the associated demand.
   This policy was described in Maglaras and Meissner (2006). It is a refinement of the static pricing policy in (a) and the make-to-order heuristic of Gallego and van Ryzin (1997). Other examples of joint pricing and capacity controls include Vulcano et al. (2002), Lin et al. (2003), and Feng and Xiao (2004).

c. **A dynamic pricing heuristic.** The solution of the fluid pricing problem studied earlier can be described in feedback form as

$$\bar{\rho}(x,t) = \min\left(\hat{\rho}, \frac{x}{T-t}\right), \tag{25}$$

where x is the remaining capacity at time t. The third heuristic translates the aggregate control $\bar{\rho}(x,t)$ into product-level rates (and prices) through

$$\begin{aligned} \lambda(x,t) &= \lambda^r(\bar{\rho}(x,t)) \quad \text{and} \\ p(x,t) &= p(\lambda(x,t)), \end{aligned} \tag{26}$$

where the mapping $\lambda^r(\cdot)$ was the maximizer in (17) and it is continuous in ρ. This corresponds to the idea of resolving the fluid problem while stepping through time. This is widely applied in practice, where, however, the resolving occurs at discrete points in time, e.g., daily or weekly, depending on the application setting. These resolving policies seem to have been first analyzed in Maglaras and Meissner (2006), which show that the idea of resolving applied in the context of the dynamic pricing or the LPCC heuristics is fluid-scale asymptotically optimal, as is the static policy (a). The single-product version of this feedback pricing policy has also been studied by Reiman (2002). These result show that the suboptimal behavior demonstrated by a resolving policy in the negative example studied in Cooper (2002) does not persist in systems with large capacity and large demand. Intuitively, resolving is nothing but implementing the fluid policy in feedback form. Numerical experiments documented in many papers (e.g., Maglaras and Meissner (2006)) demonstrate that such feedback heuristics tend to outperform policies that are static.

Dynamic Pricing Network Revenue Management Problems

This section offers a glimpse of network revenue management problems, and specifically a brief review of its associated fluid model formulation; see Gallego and van Ryzin (1997) and Talluri and van Ryzin (2004b) for a more detailed treatment. Suppose that the firm is operating a network of resources, indexed by $j = 1, \ldots, m$, and that each product i request consumes A_{ij} units of resource j capacity. Let $A := [A_{ij}]$ denote the associated capacity consumption matrix, and assume that the initial capacity for each resource j is C_j. Then, the fluid model formulation of the network dynamic pricing problem is:

$$\max_{\{\lambda(t),\ t=1,\ldots,\ T\}} \left\{ \sum_{t=1}^{T} R(\lambda(t))dt : \sum_{t=1}^{T} A\lambda(t)dt \leq C \text{ and } \lambda(t) \in \mathcal{L}\ \forall t \right\}. \tag{27}$$

As before, this problem can be expressed in terms of ρ, which is defined by $\rho := A\lambda$. Specifically, let

$$R^r(\rho) := \max_{\lambda} \{R(\lambda) : A\lambda = \rho,\ \lambda \in \mathcal{L}\}, \tag{28}$$

be the maximum achievable revenue rate when resource capacity is consumed at a rate ρ, and $\lambda^r(\rho)$

denote the corresponding vector of optimal demand rates. Then, (27) can be reduced to

$$\max_{\{\rho(t),\ t=1,\ldots,\ T\}} \left\{ \sum_{t=1}^{T} R^r(\rho(t))dt : \sum_{t=1}^{T} \rho(t)dt \le C \text{ and } \rho(t) \in \mathcal{R}\ \forall t \right\}. \quad (29)$$

Let $\bar{\rho}$ denote the solution to (29). Then, $\lambda^r(\bar{\rho})$ is the vector of optimal demand rates for (27). This reduction could prove computationally beneficial, since as is often the case the number of products (e.g., the number of fare-class and origin–destination pairs) tends to be greater than the number of resources (e.g., number of flights in a hub-and-spoke network). However, unlike the treatment of the single-resource models in the previous section, this reduction need not necessarily apply to the discrete and stochastic formulation of the underlying revenue management problem.

Concluding Remarks

The papers by Elmghraby and Keskinocak (2003), Bitran and Caldentey (2003), and McGill and van Ryzin (1999), and the book by Talluri and van Ryzin (2004b) provide comprehensive overviews of the areas of dynamic pricing and revenue management. The modeling framework adopted here closely matches that of Gallego and van Ryzin (1994, 1997). Additional references on the capacity control formulation are Brumelle and McGill (1993) and Lautenbacher and Stidhman (1999). The reduction of the multi-product and capacity control problems to single-product pricing problems and their subsequent solutions are from Maglaras and Meissner (2006). The asymptotic analysis that is briefly reviewed here builds on the setup used in Gallego and van Ryzin (1994, 1997) and Cooper (2002), and the results reviewed here are adopted from Maglaras and Meissner (2006). The idea of efficient controls and that of a notion of an efficient frontier on how to choose the product level pricing and capacity control decisions to achieve a desired capacity absorption rate is from Maglaras and Meissner (2006). Related ideas have appeared in slightly different settings in Talluri and van Ryzin (2004a), in the context of a capacity control problem for a model with customer choice among products, and in Feng and Xiao (2000, 2004), while studying pricing problems with a predetermined set of price points; the presentation of this topic here follows Maglaras and Meissner (2006). The pricing and capacity control heuristics have appeared in many references such as Gallego and van Ryzin (1994, 1997), McGill and van Ryzin (1999), Feng and Xiao (2004), Lin et al. (2003), and Maglaras and Meissner (2006). The feedback form of the static pricing policy that is optimal for the deterministic and continuous (fluid) analog of the stochastic and discrete dynamic pricing problem and the property of asymptotic optimality of the resolving pricing heuristic that is based on that feedback policy are both from Maglaras and Meissner (2006).

The are several extensions, generalizations, and new directions of work in the area of revenue management that were not reviewed here. Some may be the result of practical considerations. For example, one may want to consider pricing policies where the feasible price grid is discrete, say in $10 increments. Another extension would incorporate costs incurred when the price is changed. Other important research directions include revenue maximization problems for which the seller does not have accurate information about the underlying demand model, where in such settings, the seller uses its pricing decisions to simultaneously learn the demand and optimize revenues; the effect of strategic consumer behavior on revenue maximization practices, such as intentional waiting for sales to make retail purchases; and multi-product and multi-firm pricing problems under competition.

See

- Bellman Optimality Equation
- Dynamic Programming
- Inventory Modeling
- Markov Decision Processes
- Yield Management

References

Bitran, G., & Caldentey, R. (2003). An overview of pricing models for revenue management. *Manufacturing & Service Operations Management, 5*(3), 203–229.

Brumelle, S., & McGill, J. (1993). Airline seat allocation with multiple nested fare classes. *Operations Research, 41*(1), 127–137.
Cooper, W. (2002). Asymptotic behavior of an allocation policy for revenue management. *Operations Research, 50*(4), 720–727.
Elmaghraby, W., & Keskinocak, P. (2003). Dynamic pricing: Research overview, current practices and future directions. *Management Science, 31*, 47–66. To appear.
Feng, Y., & Xiao, B. (2000). A continuous-time yield management model with multiple prices and reversible price changes. *Management Science, 46*(5), 644–657.
Feng, Y., & Xiao, B. (2004). *Integration of pricing and capacity allocation for perishable products*. Chinese University of Hong Kong, Preprint.
Gallego, G., & van Ryzin, G. (1994). Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Management Science, 40*(8), 999–1020.
Gallego, G., & van Ryzin, G. (1997). A multiproduct dynamic pricing problem and its applications to network yield management. *Operations Research, 45*(1), 24–41.
Horn, R. A., & Johnson, C. R. (1994). *Matrix Analysis*. Cambridge: Cambridge University Press.
Lautenbacher, C., & Stidham, S. (1999). The underlying Markov Decision Process in the single-leg airline yield-management problem. *Transportation Science, 33*(2), 136–146.
Lee, T., & Hersh, M. (1993). A model for dynamic airline seat inventory control with multiple seat bookings. *Transportation Science, 27*(3), 252–265.
Lin, G. Y., Lu, Y., & Yao, D. D. (2003). *The stochastic knapshack revisited: structure, switch-over policies, and dynamic pricing*. Columbia University, Preprint.
Maglaras, C. (2005). Revenue management for a multi-class single-server queue via a fluid model analysis. *Operations Research, 54*(5), 914–932.
Maglaras, C., & Meissner, J. (2006). Dynamic pricing strategies for multi-product revenue management problems. *Manufacturing & Service Operations Management, 8*(2), 136–148.
McGill, J., & van Ryzin, G. (1999). Revenue management: reserach overview and prospects. *Transportation Science, 33*(2), 233–256.
Reiman, M. I. (2002). Asymptotically optimal dynamic pricing for a wholesale - retail telecommunications service provider. Presentation, 2nd INFORMS Conference on Revenue Management, New York.
Talluri, K., & van Ryzin, G. (2004a). Revenue management under a general discrete choice model of consumer behavior. *Management Science, 50*(1), 15–33.
Talluri, K., & van Ryzin, G. (2004b). *The theory and practice of revenue management*. Boston: Kluwer Academic.
Vulcano, G., van Ryzin, G., & Maglaras, C. (2002). Optimal dynamic auctions for revenue management. *Management Science, 48*(11), 1388–1407.
Zhao, W., & Zheng, Y.-S. (2000). Optimal dynamic pricing for perishable assets with nonhomogeneous demand. *Management Science, 46*(3), 375–388.

Revenue Neutrality Theorem

▶ Revenue Equivalence Theorem

Reversible Markov Chain/Process

A stationary Markov process whose infinitesimal generator (rate matrix) has elements given by

$$q(k,j) = \frac{\pi_j q(j,k)}{\pi_k} \quad \text{for } j,k \in E,$$

where π_j is the steady-state probability that the chain is in state j and $q(j, k)$ is the rate at which the chain goes from state j to k, i.e., the mean flow rates or probability flux satisfies detailed balance equations for every pair of nodes.

See

▶ Detailed Balance Equations
▶ Markov Chain Monte Carlo
▶ Markov Chains
▶ Markov Processes
▶ Networks of Queues
▶ Queueing Theory
▶ Rate Matrix

Revised Simplex Method

A version of the simplex method that uses an explicit or implict expression of the inverse of the current basis to calculate the simplex multipliers (prices) and related information.

See

▶ Product Form of the Inverse (PFI)
▶ Simplex Method (Algorithm)
▶ Simplex Tableau

RHS

▶ Right-Hand Side

Right-Hand Side

The column vector of coefficients $\boldsymbol{b}$ in a system of linear constraints $\boldsymbol{Ax} = \boldsymbol{b}$.

Right-Hand-Side Ranging

▶ Sensitivity Analysis

Risk

The risk of a decision d is the expected value of the loss incurred using d taken over all possible states of nature. A risk-averse person is one who prefers to behave conservatively. A decision maker (DM) is said to be risk averse if the DM prefers the expected consequence of a nondegenerate lottery to that lottery (a nondegenerate lottery is one where no single consequence has a probability of one of occurring). A DM is risk averse if and only if the DM's utility function is concave. In contrast, a risk-prone (or risk-seeking) person is one who does not prefer to behave conservatively. A DM is said to be risk prone (or risk seeking) if the DM prefers any nondegenerate lottery to the expected consequences of that lottery. A DM is risk prone if and only if the DM's utility function is convex. Finally, a DM is risk neutral if and only if the DM's utility function is linear.

See

▶ Lottery
▶ Risk Assessment
▶ Utility Theory

Risk Assessment

Clyde G. Chittister[1] and Yacov Y. Haimes[2]
[1]Carnegie Mellon University, Pittsburgh, PA, USA
[2]University of Virginia, Charlottesville, VA, USA

Introduction

To the layman, risk is often quantified in terms of probabilities, whereby it might be said that gambling on any event with a low probability of occurring is a risky proposition. Or, the mere existence of a catastrophic event with non-zero probability exposes those involved to risk. The risk of nuclear plant failure, global warming, or the depletion of the ozone layer are examples. One might say that the risk of not carrying an automobile insurance policy is not worth the risk. One result of these simple considerations is that the standard components of risk are the chance of a loss, the possible magnitude of the loss, and the exposure to that loss.

Public interest in the field of risk analysis has grown and expanded in leaps and bounds since 2000. Furthermore, since 1990, risk analysis has emerged as an effective and comprehensive procedure that supplements and complements the overall management of almost all aspects of people's lives. Managers of health care, the environment, and physical infrastructure systems (e.g., water resources, transportation, and electric power) all incorporate risk assessment in their decision-making processes. The omnipresent adaptation of risk analysis by many disciplines and its deployment by industry and government agencies in decision making have led to an unprecedented development of theory, methodology, and practical tools. Technical articles on risk assessment address concepts, tools, and technologies that have been developed and practiced in such areas as design, development, system integration, prototyping, and construction of physical infrastructure; in reliability, quality control, and maintenance; and in the estimation of cost and schedule and in project management (Haimes 2009).

Risk, a measure of the probability and severity of adverse effects, is a concept that many find

difficult to comprehend, and its quantification has challenged and confused lay persons and professionals alike. There are myriad reasons for this state of affairs. One of the fundamental elements that causes so much confusion and misunderstanding of the concept of risk is that it is composed of two diverse constructs. It is a complex composition and amalgamation of two components, one real (the potential damage, or unfavorable adverse effects and consequences), the other an imagined, mathematical human construct, termed probability. Probability, per se, is intangible, yet its omnipresence in risk-based decision making is indisputable. Furthermore, the measure of the probability that dominates the measure of risk is itself uncertain, especially for rare and extreme events, e.g., when there exists an element of surprise. Furthermore, what is meant by the terms probability (or likelihood) and adverse effects? Consider the interpretation of the term likelihood in isolation (for now) of its probable consequences: Is it the likelihood of the occurrence of any kind of threat (or other initiating event), at any level or magnitude, and when, and of what duration? Or, is it the likelihood of the level and magnitude of the consequences (for every element of the vector of consequences)? Thus, the phrase "*probability and severity of adverse effects*" can be interpreted in two ways at the same time: (1) in terms of the probability of the occurrence of adverse effects, and (2) in terms of the probability of the severity of adverse effects, given their occurrence. Both interpretations are valid; however, each represents varied conceptual and theoretical challenges (Haimes 2009).

In the first issue of *Risk Analysis*, Kaplan and Garrick (1981) set forth the following "set of triplets" definition of risk, R:

$$R = \{<S_i,\ L_i,\ X_i>\} \tag{1}$$

where S_i denotes the ith risk scenario, L_i denotes the likelihood of that scenario, and X_i the "damage vector" or resulting consequences. This definition has served the field of risk analysis well since then, and much early debate has been thoroughly resolved about how to quantify the L_i and X_i, and the meaning of probability, frequency, and probability of frequency in this connection (Kaplan 1993).

In Kaplan and Garrick (1981), the S_i themselves were defined, somewhat informally, as answers to the question, "What can go wrong?" with the system or process being analyzed.

Subsequently, a subscript "c" was added to the set of triplets by Kaplan (1991, 1993):

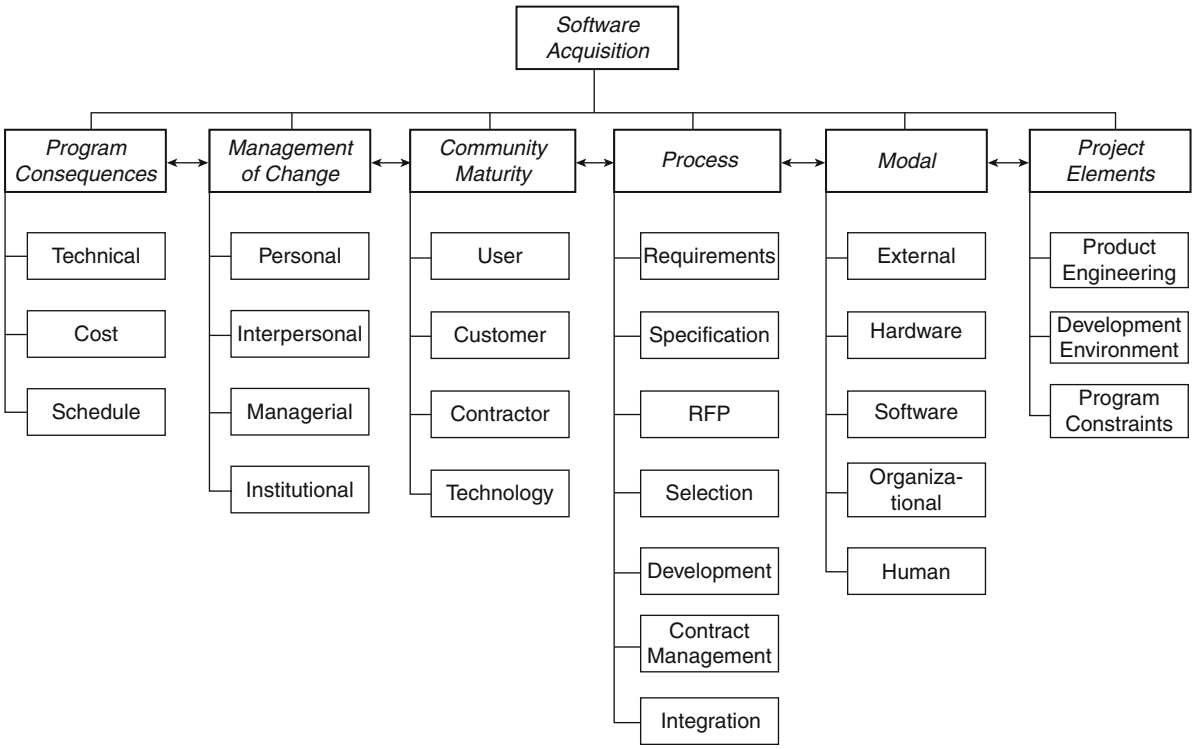
$$R = \{<S_i,\ L_i,\ X_i>\}_c \tag{2}$$

to denote that the set of scenarios, $\{S_i\}$, should be complete, meaning it should include all the possible scenarios, or at least all the important ones.

Also in Kaplan (1991, 1993), the idea of the "success," or "as-planned," scenario was introduced and denoted by S_0. The risk scenarios S_i could then be visualized as deviations from S_0. Thus the idea began to gel that the various risk analysis methods used in different industries (e.g., failure mode and effects analysis (FMEA), fault trees, and event trees) could be viewed as just different systematic ways of identifying and categorizing these deviations, S_i. When these methods became generalized and when the Russian method of anticipatory failure determination (AFD) was added, this idea matured into what is now called the theory of scenario structuring (TSS) (Kaplan et al. 2001).

At about the same time that the definition of risk article (Kaplan and Garrick 1981) was published, so too was the first article on hierarchical holographic modeling (HHM) (Haimes 1981). Central to the HHM method is a particular form of diagram, which is particularly useful for the analysis of systems with multiple, interacting (perhaps overlapping) subsystems such as a regional transportation or water supply system (Haimes 2009). Figure 1 presents an HHM for software acquisition (Schooff et al. 1997; Haimes 2009). The different columns in the diagram reflect different perspectives on the overall system.

The HHM methodology recognizes that most organizational as well as technology-based systems are hierarchical in structure, and thus the risk management of such systems must be driven by and responsive to this structure. The intent is that from this perspective, multiple methods can be compared, and thus be better understood. The risk analyst then can be more confident and flexible when choosing, mixing, and designing the method applicable to a specific problem.



Risk Assessment, Fig. 1 HHM for software acquisition (Schooff et al. 1997; Haimes 2009)

HHM can be seen as part of the TSS and vice versa. Under the sweeping generalization of the HHM method, the different methods of scenario structuring can lead to seemingly different sets of scenarios for the same underlying problem. This fact is a bit awkward from the standpoint of the set-of-triplets definition of risk (Kaplan and Garrick 1981). To eliminate this awkwardness, this definition of risk is refined to make explicit what was only implicit before: The set of risk scenarios used in a quantitative risk analysis should be (1) complete, (2) finite, and (3) disjoint. These three properties can be achieved by first noting that in realistic problems, there is always an underlying continuum of possible scenarios; this continuum is then divided into a finite set of nonoverlapping subsets. Thus, recognizing that each such subset is itself a scenario leads to a complete, finite, and disjoint set. The mathematical term for this dividing process is partitioning.

The HHM approach divides the continuum but does not necessarily partition it. In other words, it allows the set of subsets to be overlapping, i.e., non-disjoint. It argues that disjointedness is required only when the likelihood of the scenarios is going to be quantified, and even then, only if these likelihoods are going to be added up (in which case the overlapping areas would end up counted twice). Thus, if the risk analysis seeks mainly to identify scenarios rather than to quantify their likelihood, the disjointedness requirement can be relaxed somewhat, so that it becomes a preference rather than a necessity.

With this understanding, the risk identification and scenario structuring dimensions of HHM take their place within the TSS as an extremely general scenario identification process, alongside the other well-known but more specific processes: FMEA, hazard and operations analysis (HAZOP), fault and event trees, and AFD (Haimes 2009).

In seeing how HHM and TSS fit within each other, one key idea is to view the HHM diagram as a depiction of the success scenario S_0. Each box in the diagram may then be viewed as defining a set

of actions or results required of the system, as part of the definition of success. Conversely then, each box also defines a set of risk scenarios; the set of scenarios in which there is failure to accomplish one or more of the actions or results defined by that box. The union of all these sets of risk scenarios is then "complete" in that it contains all possible risk scenarios.

In (1) the choice of the subscript i, on the S_i, carries with it, by conventional usage, the implicit assumption that the set of scenarios is denumerable (i.e., countable). Moreover, because (1) is intended to describe the result of an actual risk analysis, there is the further implicit assumption that the number of scenarios in the set $\{S_i\}$ is finite. To release both these assumptions, revise (2) to read:

$$R = \{<S_\alpha,\ L_\alpha,\ X_\alpha>\}, \alpha\, \varepsilon\, A \tag{3}$$

where the index α now ranges over a set A, which in general is non-denumerable. The set A is therefore infinite and non-denumerable. It has the same order of infinity as the real number continuum.

From the perspective of this framework, TSS can now be viewed as a study of the various techniques for achieving such a partitioning. Having defined the success scenario S_0, the process of finding the risk scenarios, S_i, consists of decomposing S_0 into parts or components. Then, putting a magnifying glass over each part in turn, it is asked, "What could go wrong in this part?" In this way the S_i is generated.

Now (2) and (3) can be connected by recalling the principle that every scenario, S_i, that can be described with a finite number of words is itself a set of scenarios (Kaplan 1991, 1993). Thus, each S_i in (2) can be visualized as a subset of S_A. For practical purposes, the set of scenarios in the risk analysis, $\{S_i\}$, should be

1. Complete, in the sense that $\cup_i S_i = S_A$;
2. Finite; and
3. Disjoint, meaning that $S_i \cap S_j, = \emptyset$ for all $i \neq j$.

Such a set of subsets of S_A is termed a partitioning, P, of S_A. Thus, the goal of risk analysis can be viewed as identifying a partitioning of the underlying risk space S_A. The individual sets in this partitioning are the scenarios S_i, which are finite in number, disjoint, and together cover the underlying space S_A. It may then be written

$$R_P = \{<S_i,\ L_i,\ X_i>\}_P \tag{4}$$

R_P is thus an approximation to R based on the partition P:

$$R_P \approx R \tag{5}$$

Within the field of risk assessment, TSS and HHM aspire to be a comprehensive treatment of the process of finding, organizing, and categorizing the set of risk scenarios. As such, each should include within itself the well-known standard methods of scenario identification such as fault trees, FMEA, and failure mode, effects, and criticality analysis (FMECA) (Haimes 2009).

See

- ▶ Risk Management for Software Engineering
- ▶ Stochastic Programming

References

Haimes, Y. Y. (1981). Hierarchical holographic modeling. *IEEE Transactions on Systems, Man, and Cybernetics, 11*(9), 606–617.

Haimes, Y. Y. (2009). *Risk modeling, assessment, and management* (3rd ed.). New York: Wiley.

Kaplan, S. (1991). The general theory of quantitative risk assessment. In Y. Haimes, D. Moser, & E. Stakhiv (Eds.), *Risk based decision making in water resources V* (pp. 11–39). New York: American Society of Civil Engineers.

Kaplan, S. (1993). The general theory of quantitative risk assessment—Its role in the regulation of agricultural pests. *Proceedings of the APHIS/NAPPO International Workshop on the Identification, Assessment and Management of Risks due to Exotic Agricultural Pests, 11*(1), 123–126.

Kaplan, S., & Garrick, B. J. (1981). On the quantitative definition of risk. *Risk Analysis, 1*, 11–27.

Kaplan, S., Haimes, Y. Y., & Garrick, B. J. (2001). Fitting hierarchical holographic modeling into the theory of scenario structuring and a resulting refinement of the quantitative definition of risk. *Risk Analysis, 21*(5), 807–815.

Lowrence, W. W. (1976). *Of acceptable risk: Science and determination of safety*. Los Altos: William Kaufman.

Schooff, R. M., Haimes, Y. Y., & Chittister, C. (1997). A holistic management framework for software acquisition. *Acquisition Review Quarterly, 4*(1), 35–85.

Risk Management for Software Engineering

Clyde G. Chittister[1] and Yacov Y. Haimes[2]
[1]Carnegie Mellon University, Pittsburgh, PA, USA
[2]University of Virginia, Charlottesville, VA, USA

Introduction

Most, if not all, engineering systems are conceived, designed, constructed, marketed, and maintained under great unknowns and immense uncertainties. This lack of knowledge is not limited to technological issues such as strength of material, functionality, performance, accuracy, and quality of the components and the total product, but in fact spans a diversity of non-technical areas as well, such as predictions of customer, competitor, and market behaviors, or the anticipation of the product's impact on the organization that manufactures it. Thus, there are risks (measured by the probability and severity of adverse effects) associated with engineering systems; these risks must be managed. Computer software plays a critical role in the operation of engineering systems, due to the pervasive nature of computers in society, plus the integrality of software to the business of OR/MS.

Because software has its foundation in mathematics and logic and not in physical laws, the ability of a software engineer to introduce uncertainty into a software system is greater than in any other field. Only through very stringent management can those uncertainties introduced during the software development cycle be effectively controlled. The increased influence of software in decision making has introduced a new dimension to the way business is done in engineering quarters: many of the used-to-be-engineering decisions have been or transferred and transformed, albeit in a limited and controlled manner, to the software function. This power shift in software functionality, the explicit responsibility and accountability of software engineers, and the expertise required on the job of technical professionals, has interesting manifestations, implications, and challenges for the software engineers to adapt to new realities and to change — all of which affect the assessment and management of risk associated with software development (Chittister and Haimes 1994; Haimes and Chittister 1996).

Perhaps one of the most striking manifestations of this power shift relates to real-time control systems. Quality control in the manufacture of an engineering component, for example, is no longer primarily the responsibility of the operator; instead, the software controlling the process also controls the quality. Thus, in many respects, the software, which is designed and developed by software engineers, actually controls the process, not the engineers who originally designed the product. This implies that a shift has taken place from a strictly hardware-engineering perspective to a hardware-and software-engineering perspective. Software now fundamentally influences the design of the system. For example, the C-17 transport aircraft has been called the most computerized, software-intensive transport aircraft ever built (General Accounting Office 1992). Similarly, the Space Station has "on-board computers . . . critical to space craft safety and mission" (General Accounting Office 1989). Likewise, neither the A320 Airbus nor a number of high technology civilian and military systems can perform their functions without software. Such examples illustrate the difference between software as amanufacturing or implementation mechanism and software as a system-design component. A comprehensive discussion of software acquisition is presented by Boehm and Lane (2007) and Haimes and Chittister (1996).

For example, the decision to update or change operating parameters or entire algorithms based on real-time sensor data received from other sources is now embedded in the software system design. As another example, the data selected to be displayed on one system may be based on information received from other systems. Indeed, the types of changes or updates being implemented by software today would have, in the past, required either system hardware modification or a fundamental redesign of the system. In spite of these examples and others like them, software risk assessment and management, as a specialized entity, with all its importance and implications on other engineering systems and humans, remains an emerging rather than a well-understood activity.

Since software development, in the majority of cases, is an ad hoc process (Humphrey 1990), it is

not surprising that the risk identification and management process has been by and large ad hoc also. That process, however, can be made systematic and structured even if the software development process is not. The advances in hardware technology and reliability and the seemingly unlimited capabilities of computers render the reliability of most systems to be more heavily dependent on the integrity of the software used. Thus, software failure must be scrutinized with respect to its contribution to overall system failure and with the same diligence and tenacity that have been devoted to hardware failure.

Software Risk

In the software development process, the following three basic questions must be posed and answered at each stage regarding risk: (1) What can go wrong? (2) What is the likelihood that it will go wrong? (3) What are the consequences? (Kaplan and Garrick 1981). Now it can be added: what is the time frame? (Haimes 2009a). Only after these questions have been answered can the final question be asked: What can be done? Determining what can be done entails developing alternative design options; evaluating trade-offs; selecting one or more acceptable options (in terms of cost, reliability, performance, total quality, and safety); and evaluating the impact of current policies on future options. To answer the first three questions in the risk assessment process, however, one may benefit from knowledge of the four major sources of failure of systems in general, as well as in software development (Haimes 1991, 2009a, b):

1. Hardware failure
2. Software failure (which includes software used in the development of software)
3. Organizational failure
4. Human failure

The evolving role of the software engineer in decision making has created and continues to create enormous new challenges. The risk of not meeting specified product quality has also shifted. What was once solely the responsibility of traditional engineers who had technical know-how, expertise, and experience is now responsibility shared with software engineers, who design and develop the controlling software (Haimes and Chittister 2005, 2006; Chittister and Haimes 2004).

Although all engineering managers practice risk management in one way or another, only a minority follow this systemic process by looking for sources of failure across the entire system. The intricacy and complexity of the risk assessment and management process (when applied to complex engineering systems) and the need for quantitative analysis (which requires knowledge in probability and statistics, and frequently other content knowledge) have contributed to the emergence of the subspecialization of risk management in engineering. Thus, seeds for two seemingly distinct groups — engineers as managers of risk, and risk experts as managers of engineering systems — have been sown. In a parallel way, one may trace the distinction between (a) the engineer as a technical expert, one primarily concerned with the technical aspects of a project and to a lesser degree with managerial issues; and (b) the manager, one primarily concerned with management (in the broader and more encompassing sense of the term) and to a lesser degree with technical aspects.

Here again, the engineer (as a local manager) and the manager (as a more global manager with a broader vision and perspective) share responsibilities, tools, and methodologies, yet at the same time, each performs distinct functions, matures in different professional cultures, often uses a different jargon, and communicates with a different language. Understanding this emerging paradigm surrounding the three entities — software engineering, management, and risk analysis — is at the heart of understanding the emergence of software technical risk management.

Furthermore, to appreciate the connectedness among the three elements of this paradigm, one must also understand the hierarchical managerial structure and the consequences of its divisions:

1. *Upper management*: This group views risk almost exclusively in terms of profitability, schedule, and quality. Risk is also viewed in terms of the organization as a whole, and the effect on multiple projects or a product line.
2. *Program management*: Although this group is concerned with profitability, it concentrates more on cost, schedules, product specificity, quality, and performance, usually for a specific program or project.

3. *Technical staff* (software engineers, hardware engineers, etc.): This group of professionals concerns itself primarily with technical details of components, subassemblies, and products for one or more projects.

Clearly, differences among the risk managers at each level of this hierarchical decision making structure are caused by numerous factors, including the scope and level of responsibilities, time horizon, functionality, as well as requirements of skill, knowledge, experience, and expertise. Consequently, these differences determine, to a large extent, the tools and methodologies employed by risk managers at various levels. The management of risk associated with the development of software is governed by the same hierarchical decision making structure and by the same interconnected engineering-management-risk subspecialization paradigm.

Technical vs. Non-technical Risk

The increase in the influence and dominance of software on the system necessarily accompanies an increase in the elements of risk and uncertainty. Although no single classification of risk associated with software development has been developed, a dichotomous model of software technical risk vs. software non-technical risk is adopted here (Chittister and Haimes 1994).

This dichotomy between software technical and non-technical risk is introduced not for the purpose of distinguishing between two types of software products; rather, this classification distinguishes various functions in the developmental process of software, and thus, is concerned with the expertise required to deliver each function. Clearly, software technical and non-technical risks are dependent on and influence one another. For example, during a systems integration phase, the developed software may not meet some performance criteria or requirements. In this case, management has several options, including fixing the product and thus delaying the delivery time (and possibly exceeding the budgeted cost) or shipping the product as-is on time. In either case, however, the sources of software technical risk have not changed: only the consequences have been altered.

Software technical risk is defined as a measure of the probability and severity of adverse effects inherent in the development of software that does not meet its intended functions and performance requirements. Thus, software technical risk connotes the risk associated with those aspects in the software developmental process that are concerned with the quality, precision, accuracy, and performance over time of the developed software. In other words, software technical risk connotes the risk associated with building a software product that meets intended functions and performance.

On the other hand, software non-technical risk connotes the risk associated with the programmatic aspects in the developmental process of software that are concerned with general management, that is, with personnel, contractor selection, scheduling, budget, and marketing.

Software non-technical risk is defined as a measure of the probability and severity of adverse effects that are inherent in the development of software and are associated with the programmatic aspects in the development process of software. Although each type of risk may have an impact on the other, this distinction is still useful because it improves the process of risk assessment and management by establishing causality. Indeed, the distinction between software technical risk (e.g., noncompliance with expected product quality) and software non-technical risk (e.g., cost overruns and delays in scheduled delivery of the product) is helpful in many ways. Indeed, cognizance of the differences between the types of risks should improve their assessment and management, not serve as a detriment to dealing with them. In other words, while the distinction among the multifarious sources and types of risks is important only to the extent that the totality of these sources and type can be accounted for through their inherent differences, the successful management of risk can be achieved only through an integrated and holistically-based approach. Since software development is an intellectual, labor-intensive activity, the role of humans and human factors must be carefully understood to properly assess and manage software technical risk.

The sources of risk associated with software development are many and varied. Indeed, at each stage of the software life-cycle (design, development, testing, installation, integration into a larger system,

and its ultimate use), one can identify numerous sources of risk.

The road to building a software system is full of surprises. Software often goes through numerous changes, upgrades, fixes, recompiles, and system builds, etc., to address problems; nevertheless, new problems invariably arise. These changes take place because requirements change, people make mistakes, hardware manufacturers make changes to the system in response to marketing information, engineers introduce improvements, and software vendors upgrade their tools. In addition, often there is a break in communication, all these changes necessarily introduce uncertainties into the software development process and into the road to building software.

The ability to predict software problems before-hand has three major components:

1. Identifying and anticipating problems before they happen.
2. Determining the magnitude of potential or existing problems or risks.
3. Communicating the problems or risks to the appropriate people (people who cause, fix, are affected by, or are responsible for the problems).

The Role of Software in a Larger System

To understand what software development risk is, and to contribute to its assessment and management through the transfer of knowledge from the hardware engineering field to the software engineering field, one must (a) recognize the salient features and differences between the development processes of hardware engineering and software engineering; (b) understand the role of software engineering within the entire system; (c) appreciate, in the context of design and development, the uniqueness of software failure as juxtaposed against hardware failure, recognizing the importance of all four sources of system failure — hardware, software, organizational, and human; and (d) be familiar with the process of risk assessment and management from a total systems viewpoint.

There would be, in general, a finite and unambiguous number of fundamentally different paths or design options for hardware development to meet a given set of design specifications. Indeed, the extensive use of fault-tree analysis builds on this premise of finiteness. This is not so in the case of the architectural design of software; the number of significantly distinguishable paths or design options of software, for any given specifications, is significantly larger, more ambiguous, and broader. This inherently large number of degrees of freedom in design defies attempts to rely on historical statistics in predicting potential defects, faults, and errors in the development of software. A case in point is the development of high performance computing technology (Chittister and Haimes 2010). Significant challenges are facing the following three different groups within the professional community that support the development of large-scale scientific and engineering software applications. The first of the groups encompasses the application developers of large-scale scientific and engineering software systems, especially those requiring high-performance computing (HPC). The second group covers the HPC software development and run-time environments. The third consists of the integrators of the first two groups, with a focus on the systems engineers whose task is to bridge the technical and cultural gap between the other two. These challenges reside in several areas, the most important being the educational and cultural backgrounds that are reflected in the knowledge, expertise, and experience of the principals involved (Chittister and Haimes 2010).

The design and development of software do not typically follow well-established protocol and commonly accepted procedures. Indeed, in most cases, each software development is envisioned as a unique and distinct product. This lack of a well-developed and acceptable protocol has major implications on several dimensions for the assessment of software development risk.

Hardware has been increasingly taking the component role, whereas software has been forcefully assuming the overall systems role. Clearly, however, this is not the case in all organizations. This seemingly pivotal development has significant implications for the evolving influence of software engineers on important and critical decisions concerning product design, development, and marketing. The coordination of myriads of components in one system can often be accomplished more cost effectively and with higher reliability through software; this is a marked departure from

past practices. This fact has also brought the role of systems engineers to greater prominence in formulating policy affecting product design and development. Furthermore, the software engineers' evolving role in implementation, more than the knowledge that they bring into the project, constitutes another important force in the power shift from hardware to software engineers. Indeed, the last step in the development of a system always involves software engineers, a fact that carries with it more responsibility and implied authority in final product development. It is important to recognize, however, that both hardware and software engineers play an equally important role in the over all system design. In this sense, in the evolution of the power shift from hardware to software one must keep in mind that software development is an intellectually intensive activity where human factors are central. In his chapter on cognitive ergonomics, Sage (1992) makes a forceful argument about the centrality of human interaction with various aspects of the system throughout all phase so fa systems engineering life cycle. Therefore, to assess and manage software technical and non-technical risks, one must explicitly address the human element. Sage argues that "a systematic study of human error and approaches to ameliorate the effects of human error in systems and in organizations" is essential in this regard.

As the role that software is assuming in meeting system requirements grows, the impact of software on system risk grows. To be effective and meaningful, risk management must be an integral part of overall system management. This is particularly important in the management of technological systems, especially software-intensive systems, where the failure of a system can be caused by the failure of hardware, software, the organization, or its people.

Hierarchical Holographic Modeling for Risk Assessment

In the quest to develop an analytical framework for risk management of software engineering it is important to focus on the *sources and causes* of these problems, attempt to group them into a meaningful, yet manageable, number of categories, and then develop a comprehensive framework for dealing with the causes rather than the symptoms. To streamline the discussion and add order to it, a hierarchical structure is adopted.

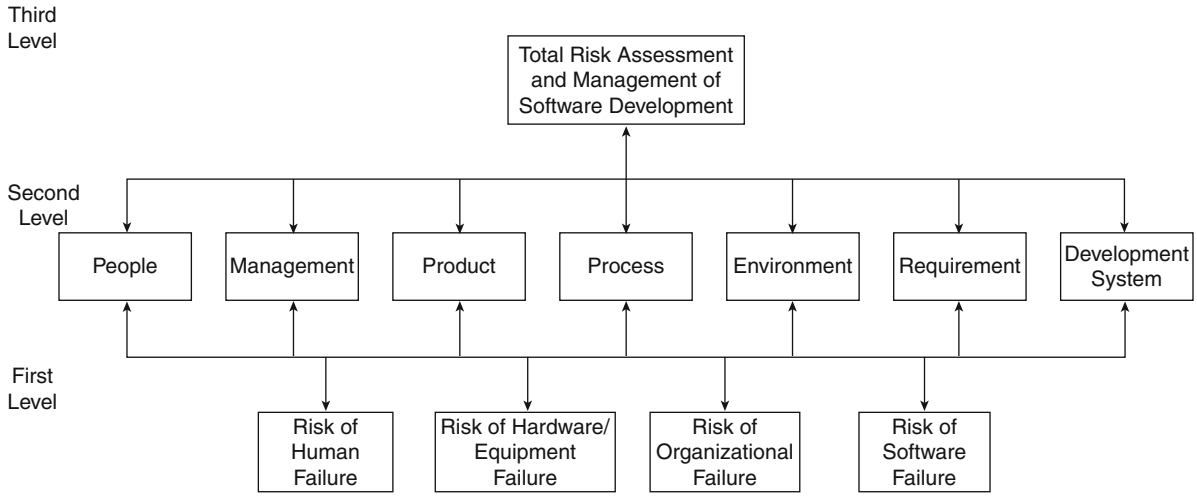
Indeed, it is impossible to do justice to a comprehensive framework for the risk assessment and management of software development by boxing it into one planar structure (model). By allowing cross-representations and overlapping models of the various facets and dimensions of the process, hierarchical holographic modeling (HHM) alleviates some of the limitations of a single schema or a single vision of the complex system (Haimes 1981, 2009a, b; Haimes et al. 1990).

Fundamentally, HHM is grounded on the premise that large-scale and complex systems, such as software development, should be studied and modeled by more than a single representation, vision, or schema. And, because such complexities cannot be adequately modeled or represented through a planar or a single vision, overlapping among these visions is not only unavoidable, but can be helpful in a holistic appreciation of the interconnectedness among the various components, aspects, objectives, and decision makers associated with such systems.

The stratagem presented here for risk identification evolves around three hierarchical levels (Chittister and Haimes 1994; Haimes 2009a, b). The three major decompositions, visions, or perspectives include: the functional perspective, the source-based perspective, and the temporal perspective.

From a functional perspective, the software development process may be decomposed into the following seven subsystems: requirement, product, process, people, management, environment, and the development system (Fig. 1). These terms may be defined as follows:

1. *Requirement*: The highest-level definition of what the product is supposed to do: what needs it must meet, how it should behave, and how the customer will use it. It corresponds to the production perspective.
2. *Product*: The output of the project that will be delivered to the customer. It includes the complete system: hardware, software, and documentation.
3. *Process*: The way by which the contractor proposes to satisfy the customer's requirement. The process is the sequence of steps — their inputs, outputs, actions, validation criteria, and monitoring activities — that leads from the initial requirement



Risk Management for Software Engineering, Fig. 1 Total risk assessment and management of software engineering: Functional-based hierarchical holographic structure

to the final delivered product. It includes such phases as requirements analysis, product definition, product creation, testing, and delivery. It includes both general management processes, such as costing, schedule tracking, and personnel assignment, and project-specific processes, such as feasibility studies, design reviews, or regression testing.

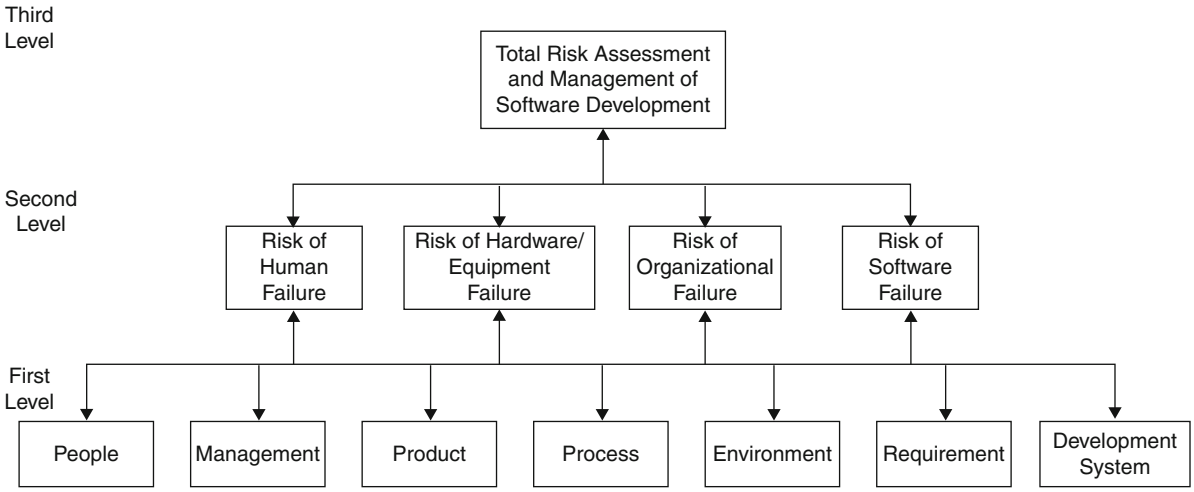
4. *People*: All those who will be associated with the technical work on the project and all the support staff. It also includes the technical advisers, overseers, and experts, whether in the chain of command or matrixed.
5. *Management*: The line managers at every level who have authority over the project, including those responsible for budget, schedule, personnel, facilities, and customer relations.
6. *Environment*: The "externals" of the project: the factors that are outside the control of the project but can still have major effects on its success or be sources of substantial risk.
7. *Development system*: The methods, tools, and supporting equipment that will be used in the product development. This includes, for instance, CASE tools, simulators, design methodologies, compilers, and host computer systems.

Another vision of the HHM can be obtained through the four sources of system failure discussed earlier (Fig. 2):

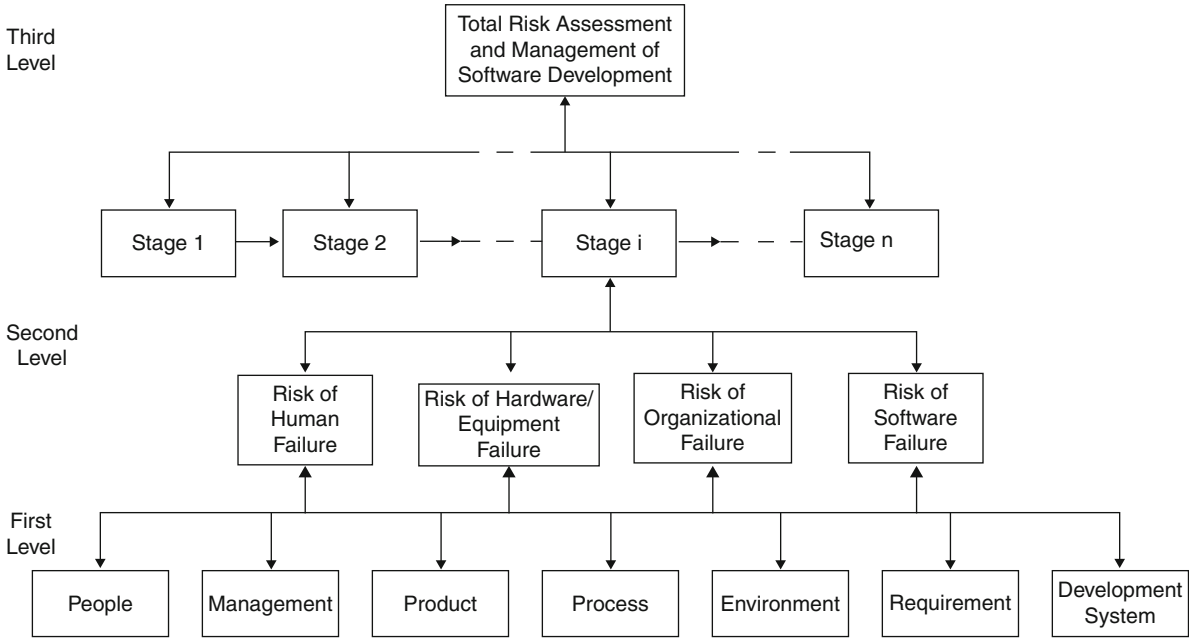
1. Hardware failure
2. Software failure (software used in the development of software)
3. Organizational failure
4. Human failure

These four sources of failure are not necessarily independent of each other. Just as the distinction between software and hardware is not always straightforward, neither is the separation between human and organizational failure. Nevertheless, these four categories of sources of failure provide a meaningful foundation upon which to build the decision-making hierarchy for the proposed framework. Note that software development is an intellectual, labor-intensive activity that must be streamlined through a well-managed organizational infrastructure and nurtured by an organizational culture and vision that are conducive to and driven by a continuous improvement philosophy.

The third vision of the HHM relates to the evolution of software development over time. Each of the various stages of software development, although often not sharply distinguishable, overlapping, and iterative, constitutes a subsystem in the temporal decomposition (Figs. 3 and 4). For purposes of this section, the temporal stages are identified as in Humphrey (1990) as: (1) system requirements; (2) software requirements; (3) analysis; (4) program design; (5) coding; (6) testing; and (7) operations.

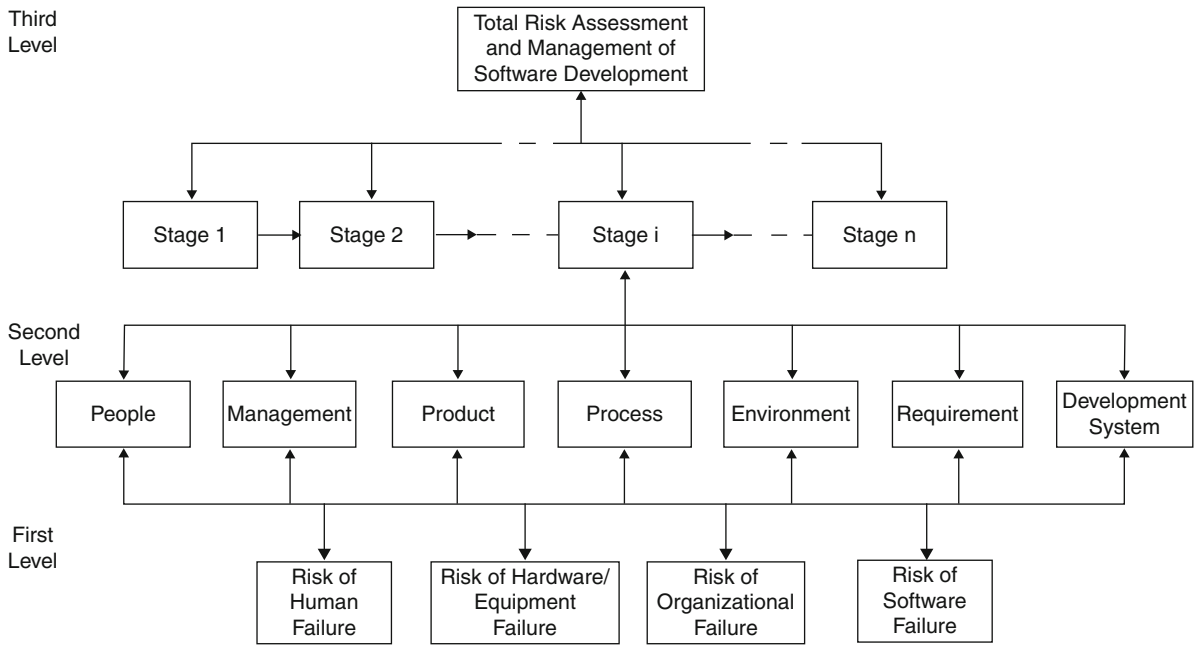


Risk Management for Software Engineering, Fig. 2 Total risk assessment and management of software engineering: Source-based hierarchical holographic structure



Risk Management for Software Engineering, Fig. 3 Total risk assessment and management of software engineering: Temporal-based hierarchical holographic structure

Each stage (subsystem) in the temporal decomposition can be viewed as one frame in a fixed time (e.g., testing) during the software development process. It is at this fixed-time frame that risks associated with the functional decomposition (e.g., requirement) and with the source-based decomposition (e.g., organizational failure) are identified and articulated. As another example, consider the following four risks that are common during each stage of software development: cost overrun, time delay, not meeting requirements, and not meeting technical quality specifications.



Risk Management for Software Engineering, Fig. 4 Total risk assessment and management of software engineering: Temporal-based hierarchical holographic structure

The temporal domain has significance far beyond the schedule of the project; it articulates how risks change and evolve over time.

Each of the three hierarchical holographic (HH) submodels developed here contributes to the identification of risk associated with software development. The overlaps among these HH submodels mimic the fuzziness that characterizes the real world of software development with respect to the inability to make a clear distinction among the various causes of failures. Figure 1, which is an inverted image of Fig. 2, presents an entirely different perspective in answering the set of triple questions: What can go wrong? What is the likelihood that it will go wrong? And what would the consequences be? Since a central objective of risk assessment is to identify, to the extent possible, everything that can go wrong, then a hierarchical holographic modeling structure is superior to a planar single model in this respect. Note, for example, that in Fig. 1, the four sources of risk (human, hardware, organizational, and software) are investigated for each subsystem of the functional decomposition (people, management, product, process, environment, requirement, and development systems). On the other hand, Fig. 2 depicts a different perspective; namely, the seven functional de-compositions are investigated for each of the four sources of risk of failure. Figures 3 and 4 incorporates the temporal decomposition that captures the stage-wise evolutionary process of software development, and thus the risk associated with each stage and for each subsystem of the functional and source-based decompositions. In particular, Figs. 3 and 4 extend Figs. 1 and 2 by incorporating the temporal domain into the HHM.

Bases for Variances in Software Cost Estimation

Most developers of large complex software systems use cost models to estimate their costs and to assess the risk of cost overrun. These models are structured on a set of relationships based on such parameters as the size and complexity of the software, the experience level of the software developer, and the type of application within which the software will be used. Different models generate different weights or levels of importance for these parameters, and not all models

use the same parameters. Radically different cost estimates can result merely on the basis of which parameters are used in the models and how they are implemented. Even when the parameters are consistent, different developers will probably not agree on the value or weight of the parameter in the first place. In fact, many organizations consider their interpretations of these parameters to contribute to their "competitive edge" because the definition affects their ability to determine costs accurately. For example, an organization that has little experience in developing space system software may not have the same perception of difficulty when developing a complex avionic software system as would an organization that has significant experience in that area. Their understanding of space systems, however, will alter their definition of the avionic system parameters. Do developers with little experience overestimate or underestimate the complexity of the task because of how they define these parameters? The central questions are, "What are the sources of risk associated with project cost estimation? How can such risk be quantified?" (Schooff 1996; Schooff et al. 1997; Haimes 2009a, b).

Although creating, maintaining, and updating project cost-estimation metrics and parameters are extremely important for an organization, it is nevertheless unlikely that a future project will be similar enough to previous projects to merit directly importing these metrics or parameters; such metrics and parameters may not be directly applicable without appropriate modifications. Indeed, cost estimators must use judgment when applying these parameters to a new project requirement. Furthermore, cost estimation constitutes a critical area with regard to the sources of risk for software development, which is without parallel to other fields. An analogy would be a contractor estimating the cost to construct a 50-story building. If the contractor had previously built only structures with a maximum of ten stories, he would not just increase the estimate five-fold. In fact, the contractor would probably question the basic foundations and relevance of extending the 10-story model to the new structure parameters. In software, however, it is not uncommon to increase estimates for new projects by a factor of five from previous projects of one-fifth the size and complexity. Many new systems have size estimates of over 1,000,000 lines of code even though the developers have little experience with systems of this size (Schooff et al. 1997; Haimes 2009a, b).

Another example is in the use of commercial off-the-shelf (COTS) software. The original assumption that a commercial database management system (DBMS) can be used to meet customer requirements may change if the customer requires features not supported by DBMS suppliers. Such changes may have serious ramifications for the cost estimate, depending on how the developer plans to solve the problem. If the developer chooses to deal with a subcontractor in a way similar to dealing with the DBMS vendor, there will be risk associated with the subcontractor. The alternative is for the developer to undertake the development of its own DBMS. This requires an additional set of assumptions, design parameters, and judgments regarding the architecture, size, experience level, domain knowledge, software engineering knowledge, and the support environment needed to develop the DBMS. Each of these assumptions, parameters, and judgments has some uncertainty associated with it, which contributes to the overall risk in the cost estimate. If the developer chooses to subcontract the DBMS development to an outside vendor, then the issue for the contractor is understanding and accounting for the set of assumptions that are made by the subcontractors on the DBMS and on the system architecture.

The ability of the developer to make valid assumptions and design decisions is usually based on a set of metrics; these metrics can either be based on current measurements or on past performance. Either way, however, there has to be an agreed-upon set of measures that is being evaluated (such as the number of lines of code needed to accomplish specified tasks, or productivity rates in terms of lines of code per hour). The difficulty with software development is that the community has not agreed upon basic measures, such as how to count lines of code or how to measure productivity. Using performance history is difficult because the systems under development are sufficiently different such that history may not adequately reflect the new parameters accurately.

There are many models for software acquisition (Schooff et al. 1997). In the spiral model of software development (Boehm 1988), the process consists of multiple repetitions of primary stages and often

extends over a great length of time. Lederer and Prassad (1993) reported that in practice, software estimation is most often prepared at the initial project proposal stage; then, with declining frequency, at the requirements, systems analysis, design, and development stages. However, as the software development community continues to move away from the traditional waterfall development process model to the spiral-type models, demand has increased for cost-estimation models that account for the dynamics of changing software requirements and design (and the always-present uncertainty) over multiple time periods (Schooff et al. 1997; Haimes 2009a, b). Bell's survey of software development and software acquisition professionals indicates that a vast majority believe a dynamic software-estimation model would be most applicable for their estimation requirements (Bell 1995).

At each stage of the acquisition process, decisions are made that affect the events and decision opportunities of subsequent phases. Software estimation is a required activity in every stage of the process. Applying the probabilistic cost-estimation method with multiple objective risk functions described in Schooff and Haimes (1997) and in Haimes (2009a, b) constitutes a multiple-objective decision problem that is solved over the multiple stages of the acquisition life cycle.

Concluding Remarks

Software will continue to grow in size, complexity, and importance as it assumes more functionality in large, complex systems. If engineers and managers working in the community do not embrace a risk management ethic for software development, then software problems will continue to grow as well. Although practicing risk management does not guarantee fewer problems, it does provide a structure with which to make better decisions about the uncertainty and impact of future events. If risks can be measured, then contingency strategies can be provided; however, if risks are unknown, then surprise is likely when it is least convenient (Haimes 2009a).

Although software engineering is different from other engineering disciplines, the management of risk in the developmental process is critical for all engineering disciplines. The framework for identifying and assessing risk in the software development process is grounded on the premise that software development is an intellectual, labor-intensive activity, thus making the human factor central to the assessment and management of risk.

As systems become larger and more complex, the assessment and management of risk must be a team effort. The team has to include, among others, the system developers, support staff from the organization, and management. Risk management is neither just the program manager's job nor just a technical issue. Financial and quality risks are as important as software technical risk.

The more diversified the team, the more important it is to have a common and agreed-upon risk assessment and management process. The members of the team will have their own technical jargon and their own frames of reference. If each subgroup identifies and manages risks differently, there will be no common ground for communication or measurement. A systematic and structured process that is used by everyone will provide a foundation for discussion and for mitigation strategies. This process will also greatly reduce confusion caused by misunderstanding, which is itself a source of risk in large complex systems.

Indeed, modeling and managing software technical risk must be an activity that recognizes the intricacy of the internal and external environment within which software development is practiced. Depending on the forces exerted and on the software development practice itself, two types of risks are likely to emerge — software technical and non-technical risks. Indigenous to these forces is the power shift from hardware to software; consequently, such change must be recognized and managed. In *Changed Agents*, London (1990) summarized his views on organizational change, as follows:

Incremental change merges the new with the old. It requires a willingness to be open to new ideas and to continuously refine and possibly extend the goals of the organization. Frame-breaking change is dramatic and often sudden. Though resistance is likely, the organization's survival depends on re-creating the organization's mission, structure, staff, and modes of operation.

Indeed, the risks of not meeting product quality and performance, cost, and schedule can be successfully

identified, quantified and measured, evaluated, and managed only when a systemic and holistic process of assessment and management is employed. Such a process is an organizational recipe for long-term sustainable development. Toffler (1990) best articulated the imperative of properly coping with technological change. The software engineering community is traveling in unexplored terrain; the success of its journey depends, to a large extent, on its ability to bring the larger systems community into the realization that acknowledging and responding to the power shift in the software area is a first and critical step in a successful management of software technical and non-technical risk.

No one would argue that the best way to manage problems is to keep them from happening. A risk ethic that is embraced and practiced by an entire organization will significantly reduce the chaos created by unknown risks and crisis situations.

See

- ▶ Quality Control
- ▶ Risk Assessment

References

Bell, G. A. (1995). Applying the system design dynamics technique to the software cost problem: A rationale. In *Proceedings of the tenth annual COCOMO user's group meeting*. Software Engineering Institute, Carnegie Mellon University, Pittsburgh.

Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer, 21*(5), 61–72.

Boehm, B. W., & Lane, J. A. (2007). Final report: Using the incremental commitment model to integrate system acquisition. *Cross- talk Journals, 1*(10), 1–13.

Chittister, C., & Haimes, Y. Y. (1993). Risk associated with software development: A holistic frame-work for assessment and management. *IEEE Transactions on Systems, Man, and Cybernetics, 23*, 710–723.

Chittister, C., & Haimes, Y. Y. (1994). Assessment and management of software technical risk. *IEEE Transactions on Systems, Man, and Cybernetics, 24*, 187–202.

Chittister, C. G., & Haimes, Y. Y. (2004). Risks of terrorism to information technology and to critical interdependent infrastructures. *Journal of Homeland Security and Emergency Management, 1*(4), 1–21.

Chittister, C. G., & Haimes, Y. Y. (2010). Harmonizing high performance computing (HPC) with large-scale complex systems in computational science and engineering. *Systems Engineering, 13*(1), 47–57.

General Accounting Office. (1989). *Automated information systems*. Washington, DC: GPO.

General Accounting Office. (1992). *Embedded computer systems: Significant software problems on C-17 must be addressed* (CAO/IMTEC-92-48). Washington, DC: Government Printing Office.

Haimes, Y. Y. (1981). Hierarchical holographic modeling. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-11*, 606–617.

Haimes, Y. Y. (1991). Total risk management. *Risk Analysis: An International Journal, 11*, 169–171.

Haimes, Y. Y. (2009a). *Risk modeling, assessment, and management* (3rd ed.). New York: Wiley.

Haimes, Y. Y. (2009b). On the complex definition of risk: A systems-based approach. *Risk Analysis, 29*(12), 1647–1654.

Haimes, Y. Y., & Chittister, C. G. (1995). An acquisition process of the management of non-technical risks associated with software development. *Acquisition Review Quarterly, II*(2), 121–154.

Haimes, Y. Y., & Chittister, C. G. (1996). Systems integration via software risk management. *IEEE Transactions on Systems, Man, and Cybernetics, 26*(5), 521–532.

Haimes, Y. Y., & Chittister, C. G. (2005). A roadmap for quantifying the efficacy of risk management of information security and interdependent SCADA systems. *Journal of Homeland Security and Emergency Management, 2*(2), 1–21.

Haimes, Y. Y., & Chittister, C. G. (2006). Cybersecurity: From ad hoc patching to lifecycle of software engineering. *Journal of Homeland Security and Emergency Management, 3*(4), 85–113.

Haimes, Y. Y., Tarvainen, K., Shima, I., & Thadathil, J. (1990). *Hierarchical multiobjective analysis of large scale systems*. New York: Hemisphere Publishing.

Humphrey, W. S. (1990). *Managing the software process*. Reading: Addison-Wesley.

Kaplan, S., & Garrick, B. J. (1981). On the quantitative definition of risk. *Risk Analysis, 1*, 11–27.

Lederer, A. L., & Prasad, J. (1993). Information systems software cost estimating: A current assessment. *JL Information Technology, 8*, 22–33.

London, M. (1990). *Change agents: New roles and innovation strategies for human resource professionals*. San Francisco: Jossey-Bass.

Lowrence, W. W. (1976). *Of acceptable risk: Science and determination of safety*. Los Altos: William Kaufmann.

Sage, A. P. (1992). *Systems engineering*. New York: Wiley.

Schooff, R. M. (1996). *Hierarchical holographic modeling for software acquisition risk assessment and management* (Ph.D. dissertation). Systems Engineering Department, University of Virginia, Charlottesville.

Schooff, R. M., & Haimes, Y. Y. (1997). *Dynamic multistage software estimation, technical report 15–97, center for risk management of engineering systems*. Charlottesville: University of Virginia.

Schooff, R. M., Haimes, Y. Y., & Chittister, C. G. (1997). A holistic management framework for software acquisition. *Acquisition Review Quarterly, 4*(1), 55–85.

Toffler, A. (1990). *Power shift*. New York: Batman Books.

Ritter's Partitioning Method

A procedure for decomposing and solving a linear-programming problem that has both coupling constraints and coupling variables.

See

► Block-Angular System

Robust Optimization

Optimization that takes into account data uncertainty without using probability by considering only the ranges of possible values of parameters, e.g., "best case" and "worst case" scenarios. For example, in the standard linear programming problem $\min_{\boldsymbol{x}} \boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{x}$ subject to $\boldsymbol{Ax} \leq \boldsymbol{b}$, the robust counterpart formulation would consider the input data $\boldsymbol{A}$, $\boldsymbol{b}$, and $\boldsymbol{c}$ in a given range comprising the uncertainty set for the parameters. This approach is contrasted with stochastic programming, which takes into account uncertainty explicitly by using probability distributions on scenarios.

See

► Linear Programming
► Sensitivity Analysis
► Stochastic Programming

References

Ben-Tal, A., El Ghaoui, L., & Nemirovski, A. (2009). *Robust optimization*. Princeton, NJ: Princeton University Press.

Robustness Analysis

Jonathan Rosenhead
The London School of Economics and Political Science, London, UK

Robustness Analysis is a method for evaluating initial decision commitments under conditions of uncertainty, where subsequent decisions will be implemented over time. The robustness of an initial decision is an operational measure of the flexibility which that commitment will leave for useful future decision choice.

The definition of the robustness of an initial commitment is:

> *the number of acceptable options at the planning horizon that are compatible with that commitment, as a ratio of the total number of acceptable options.*

In more formal notation, if
$\underline{S}$ is the set of all options at the planning horizon with acceptable performance;
$\underline{S}_i$ is the subset of $\underline{S}$ that is compatible with initial commitment d_i;
and n (.) denotes the number of elements in a set
then the robustness of commitment d_i is given by

$$r(d_i) = n(\underline{S}_i)/n(\underline{S})$$

All robustness scores lie in the range (0, 1). Higher scores are preferred to lower ones.

The same basic logic can be used where there is a particular concern to avoid access to future options that are assessed as likely to perform unacceptably badly. The mirror image concept of debility is defined as above, where now both numerator and denominator refer to *un*acceptable options. Lower debility scores are preferred.

A variant of the approach is multi-future robustness analysis. In this case the acceptability of an option may vary between futures, and the analysis produces a vector of robustness scores for each commitment. For a fuller description of Robustness Analysis, see Rosenhead (2001).

To employ Robustness Analysis it is necessary to:

- Identify the initial commitments to be evaluated;
- Select a planning horizon (typically 5–10 years, depending on the speed of change in the environment and the time lag in implementing decisions);
- Clarify the range and variety of future options that could result at the planning horizon;
- Establish which commitments are compatible with, ie leave available, which options; and
- Determine the acceptability of future options in the projected future(s)

The evaluations of 'acceptability' and of 'compatibility' may be model-based, or may be elicited from stakeholders. When employed in a workshop format Robustness Analysis is a member of the Problem Structuring Methods family. The conceptual simplicity of Robustness Analysis makes it easy to grasp intuitively.

It is not suggested that decision-makers should automatically select and implement the commitment that has the highest robustness score. The balance between flexibility and more short-term factors will depend on circumstances. The advantage that Robustness Analysis brings to the decision table is a language in which flexibility can participate systematically in the conversation.

See

- ▶ Problem Structuring Methods
- ▶ Sensitivity Analysis

References

Rosenhead, J. (2001). Robustness analysis: Keeping your options open. In J. Rosenhead & J. Mingers (Eds.), *Rational analysis for a problematic world revisited* (pp. 181–207). Chichester: Wiley.

Role-Playing

People are asked to adopt roles and then to act out various situations. This procedure, widely used for training and therapy, can also be used for forecasting.

See

- ▶ Forecasting

Rosen's Partitioning Method

A procedure for decomposing and solving a linear-programming problem that is a block-angular system with either coupling constraints or coupling variables.

See

- ▶ Block-Angular System

Roundoff Error

The computational error due to the significant-digit arithmetic inherent in digital calculations.

Route Construction Heuristic

A vehicle-routing heuristic that builds a feasible solution by inserting at every iteration unrouted customers into a current partial vehicle route.

See

- ▶ Vehicle Routing

Route Improvement Heuristic

A local improvement heuristic for vehicle routing.

See

- ▶ Vehicle Routing

Row Vector

One row of a matrix or a matrix consisting of a single row.

Rule

A named fragment of reasoning knowledge consisting of premise and a conclusion. In addition, a rule may have other attributes such as a priority, a cost, a preaction sequence, a premise-testing strategy, a textual description, and an internal comment.

See

- ► Artificial Intelligence
- ► Expert Systems

Rule Set

A named collection of rules that represent reasoning knowledge about some problem area. A rule set is used by an inference engine to solve specific problems in that area. In addition to rules, a rule set may also contain an initialization sequence, a completion sequence, and variable descriptions.

See

- ► Artificial Intelligence
- ► Inference Engine

Rule-Based Forecasting

Rules developed for the weighting of a set of extrapolation models based upon forecasting methodology guidelines and knowledge about the specific problem domain.

See

- ► Forecasting

Running Time of an Algorithm

- ► Computational Complexity

S

S-model

▶ Learning Curves

SA

▶ Sensitivity Analysis
▶ Simulated Annealing
▶ Stochastic Approximation

Saddle-Point of a Function

For an arbitrary payoff function $F(X, Y)$, the point (X^0, Y^0) is a saddle point if $F(X^0, Y) \leq F(X^0, Y^0) \leq F(X, Y^0)$.

See

▶ Saddle-Point Problem

Saddle-Point of a Game

For a zero-sum, two-person game, if an element a_{ij} of the payoff matrix is the minimum of its row and maximum of its column, it is a saddle point. The value of the game is equal to the value of the saddle point, with the maximizing player's optimal strategy being the pure strategy i and the minimizing player's optimal strategy being the pure strategy j.

See

▶ Game Theory
▶ Saddle-Point of a Function

Saddle-Point Problem

For the mathematical-programming problem: Minimize $f(\boldsymbol{x})$, subject to $\{g_i(\boldsymbol{x}) \leq b_i\}$, the saddle-point problem is to find vectors $\boldsymbol{x}^0$ and $\boldsymbol{y}^0$ such that $F(\boldsymbol{x}^0, \boldsymbol{y}) \leq F(\boldsymbol{x}^0, \boldsymbol{y}^0) \leq F(\boldsymbol{x}, \boldsymbol{y}^0)$, where $F(\boldsymbol{x}, \boldsymbol{y})$ is the associated Lagrangian function, $\boldsymbol{y} \geq \boldsymbol{0}$.

See

▶ Saddle-Point of a Function

Safety

Igor Ushakov
Qualcomm Inc., San Diego, CA, USA

Safety is a property of a system that permits the system to operate without dangerous consequences for people (including serving personnel) and the environment. For many systems (as aircraft, submarines, chemical plants, nuclear power stations, etc.), some kinds of failures can lead to catastrophic results. In these cases, the safety indices coincide with reliability

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

indices after the choice of the appropriate criteria for defining failure. These might be the (complementary) probability of successful operation without accident, the mean time to accident appearance, etc.

Sometimes the safety of systems (as for dams of hydro-power stations, constructions in seismic zones, etc.) is considered to be only under the influence of nature. In this case, probabilistic measures may be insufficient and one should instead consider conditional safety under some specified levels of external influence.

But many systems may be harmful even under ideal conditions, without accidents. Examples are various chemical and metallurgical technological processes, power stations, and other objects polluting the environment with various toxic substances.

To quantify, begin by letting $f(t)$ be the poisonous emission function in time. One useful index of safety of such a system is the condition that, for some specified time interval of width Δ,

$$\int_t^{t+\Delta} f(t)dt \le f^0$$

where the threshold f^0 is given.

If there is some reduction process $\phi(t)$ which lowers the harmful consequences described by $f(t)$, then an appropriate index could be

$$\int_t^{t+\Delta} [f(t) - \phi(t)]_+ dt \le f^0$$

where $[\cdot]_+$ denotes the positive part of the number in brackets.

Many harmful processes (radioactive emission, dioxide pollution, etc.) exponentially calm down (recover) with time. In this case, a good safety criterion might be

$$\int_t^{t+\Delta} f(t)e^{-\alpha t}\, dt \le f^0$$

where α is the intensity of the recovery.

For more complete discussions of the issues outlined here, see Ushakov (1994), particularly for the relationship of classical reliability modeling to the analysis of safety.

See

- ▶ Redundancy
- ▶ Reliability of Stochastic Systems

References

Ushakov, I. A. (Ed.). (1994). *Handbook of reliability engineering*. New York: Wiley.

Sample Average Approximation

Alexander Shapiro
Georgia Institute of Technology, Atlanta, GA, USA

Introduction

Consider the following optimization problem

$$\operatorname*{Min}_{x\in\chi} f_0(x) \text{ subject to } f_i(x) \le 0,\ i = 1, \ldots, q. \quad (1)$$

In stochastic optimization (stochastic programming), the objective and/or constraint functions are given in the form of expected values:

$$f_i(x) := \mathbb{E}[F_i(x, \xi)],\ i = 0, \ldots, q, \quad (2)$$

where " := " means "equal by definition", the set $\chi \subset \mathbb{R}^n$ is deterministic, $\xi \in \mathbb{R}^d$ is a vector representing uncertain parameters of the problem and $F_i(x, \xi), i = 0, \ldots, q$, are explicitly defined real-valued functions. In this formulation it is assumed that the parameter vector ξ is modeled as random with a specified probability distribution P, and the expectations in (2) are computed with respect to this distribution. (The same notation ξ will be used to denote a random vector and its particular realization; which one of these two meanings will be used in a particular situation will be clear from the context.)

By writing the objective function $f_0(x)$, say a cost of a certain procedure, as an expected value with respect to the probability distribution of involved random parameters, the optimization (minimization) is supposed to be performed on average. In some situations where the same procedure is repeated many

times, this can be justified by the Law of Large Numbers. On the other hand, modeling constraints as expectations could be quite different. It does not make sense trying to maintain on average power supply in a large city. Nevertheless there are some cases where expectation constraints appear naturally. One such example is given by problems with chance (probabilistic) constraints.

Suppose that it is desirable to enforce constraints $G(x, \xi) \leq 0$, where $G(x, \xi)$ is a given function depending on parameter vector ξ, for all possible realization of vector ξ varying in a specified uncertainty set $\Xi \subset \mathbb{R}^d$. However, this could be too costly or even impossible to maintain for all $\xi \in \Xi$, and one settles for a less restrictive constraint

$$\Pr\{G(x, \xi) \leq 0\} \geq 1 - \alpha, \tag{3}$$

where parameter vector ξ is modeled as random, $\Pr\{G(x, \xi) \leq 0\}$ denotes probability of the event "$G(x, \xi) \leq 0$" and $\alpha \in (0, 1)$ is a small specified number. Constraints of the form (3) are called chance (or probabilistic) constraints and $1 - \alpha$ is often referred to as the corresponding confidence level. Chance constraints were introduced by Charnes et al. (1958) and thoroughly discussed in Prékopa (1995). Recall that probability of an event A can be written as the expectation $\mathbb{E}[1_A]$, where 1_A denotes the corresponding indicator function, i.e., chance constraint (3) can be written as the expectation constraint $\mathbb{E}[H(x, \xi)] \geq 1 - \alpha$, where

$$H(x, \xi) := \begin{cases} 1 & \text{if } G(x, \xi) \leq 0, \\ 0 & \text{if } G(x, \xi) > 0. \end{cases} \tag{4}$$

A problem with the above formulation is that the function $H(x, \xi)$ is not everywhere continuous.

Monte Carlo Sampling

The expected value functions $f_i(x)$, $i = 0, \ldots, q$, are given by integrals $\mathbb{E}[F_i(x, \xi)] = \int F_i(x, \xi) dP(\xi)$. If the probability distribution P of ξ is discrete, say ξ can take values $\xi_1, \ldots, \xi_K$ with respective (positive) probabilities $p_1, \ldots, p_K$, then these integrals can be written as sums

$$\mathbb{E}[F_i(x, \xi)] = \sum_{k=1}^{K} p_k F_i(x, \xi_k), \ i = 0, \ldots, q, \tag{5}$$

and for not too large values of K could be computed in a straightforward way. On the other hand, for continuous distributions these expectations become multivariate integrals which could be evaluated in a closed form only in rather specific cases. Numerical computation of these integrals can be approached by discretization, i.e., by using approximations of the form (5). Suppose, for example, that components of the random vector ξ are distributed independently of each other, and r points are used for discretization of marginal distribution of each component of ξ. Then the total number of discretization points is $K = r^d$ and this number quickly becomes astronomically large with increase of the dimension d of ξ even for moderate values of r.

A way of dealing with exponential growth of discretization points is by using randomization based on Monte Carlo sampling techniques. Suppose that it is possible to generate in the computer a random (or rather pseudo-random) sample $\xi^1, \ldots, \xi^N$ of N independent realizations of the random vector ξ (see, e.g., Fishman 1999), i.e., $\xi^1, \ldots, \xi^N$ is an independent identically distributed (iid) sample of the random vector ξ. Then the expected value functions can be approximated by the respective sample averages $\hat{f}_{iN}(x) := N^{-1} \sum_{j=1}^{N} F_i(x, \xi^j)$. The employed sample can be viewed as a randomized discretization with each discretization point ξ^j taken with equal probability $p_j = N^{-1}$. Consequently the "true" problem (1) can be approximated by the optimization problem

$$\operatorname*{Min}_{x \in \chi} \hat{f}_{0N}(x) \text{ subject to } \hat{f}_{iN}(x) \leq 0, \ i = 1, \ldots, q. \tag{6}$$

Once the sample is generated, each $\hat{f}_{iN}(x)$ becomes an explicitly defined function of the decision vector x and (6) becomes a deterministic problem which could be solved by an appropriate deterministic algorithm. Although the sample average functions $\hat{f}_{iN}$ also depend on the generated sample, for the sake of simplicity, this dependence is suppressed with only the sample size appearing in the notation.

It is difficult to point to an exact origin of such Monte Carlo sampling approach to solving stochastic optimization problems. The idea is rather simple and natural, and the method and its variants were discovered and rediscovered by many authors under different names in various contexts and applications. In the stochastic optimization literature, it can be pointed, for example, to Rubinstein and Shapiro (1990)

and Robinson (1996), where this approach was called the stochastic counterpart method and sample-path optimization, respectively; in statistics, this type of approach was used in Geyer and Thompson (1992); in the machine learning literature, some specific forms of this approach are referred to as the empirical mean optimization. In the recent stochastic programming literature, it is often called the sample average approximation (SAA) method, the term coined in Kleywegt et al. (2001). Interestingly, only relatively recently it was realized that the SAA method can be reasonably efficient in solving certain classes of stochastic optimization problems. This was motivated by development of statistical inference of the SAA method and supported by numerical experiments.

Statistical Properties of SAA Estimates

The SAA problem (6) depends on the corresponding random sample and hence its optimal value and optimal solutions can be viewed as random (statistical) estimates of their counterparts of the "true" problem (1). The notation ϑ^* and $\hat{\vartheta}_N$ will be used for the optimal values of the true (1) and SAA (6) problems, respectively, and $\mathcal{S}$ and $\hat{\mathcal{S}}_N$ for respective sets of optimal solutions. Statistical properties of the optimal value $\hat{\vartheta}_N$ and optimal solutions $\hat{x}_N \in \hat{\mathcal{S}}_N$ of the SAA problem are discussed, e.g., in Shapiro et al. (2009). Below is presented a somewhat informal discussion of the main implications of that theory.

By the Law of Large Numbers (LLN), it follows that sample averages $\hat{f}_{iN}(x)$ converge with probability one (w.p.1) to their expected values $f_i(x)$ as the sample size N tends to infinity. The classical LLN ensures such pointwise convergence w.p.1, i.e., it holds for a fixed x provided that the expected value $f_i(x)$ is well defined and finite valued. Under mild additional conditions, it is possible to show that this convergence is uniform on any bounded subset of $\mathbb{R}^n$ (uniform LLN). It follows, under certain regularity conditions, that $\hat{\vartheta}_N$ and $\hat{x}_N$ converge to their true counterparts w.p.1 as $N \to \infty$. In the statistical terminology this means that $\hat{\vartheta}_N$ is a consistent estimator of ϑ^*. For optimal solutions $\hat{x}_N \in \hat{\mathcal{S}}_N$ of the SAA problem the convergence issue is more delicate. Assuming that the true problem has unique optimal solution $\bar{x}$, i.e., $\mathcal{S} = \{\bar{x}\}$, under mild regularity conditions it holds that $\hat{x}_N \to \bar{x}$ w.p.1 as $N \to \infty$. This, however, does not imply that $\hat{x}_N$ is a feasible point of the true problem for any sample size N. If the problem does not have expectation constraints, i.e., is of the form

$$\operatorname*{Min}_{x \in \chi} \{f(x) := \mathbb{E}[F(x, \xi)]\}, \tag{7}$$

then the (deterministic) feasible set χ of the true and the corresponding SAA problems is the same.

Next the rate of convergence is considered. For a given x, the Central Limit Theorem (CLT) implies that $N^{1/2}[\hat{f}_{iN}(x) - f_i(x)]$ converges in distribution to a normal distribution with zero mean and variance $\sigma_i^2(x) = \operatorname{Var}[F_i(x, \xi)]$. In particular, this implies that the error $\hat{f}_{iN}(x) - f_i(x)$ of the sample average estimator is of stochastic order $O_p(N^{-1/2})$. In other words in order to improve the accuracy of the sample average estimator by one digit (i.e., 10 times) the sample size should be increased by 100 times. There are some variance reduction techniques which are aimed at reducing variance of the corresponding estimators (e.g., Fishman 1999), but the rate $O_p(N^{-1/2})$ of convergence of Monte Carlo sampling estimates cannot be changed. It also could be mentioned that Quasi-Monte Carlo methods have theoretically better rates of convergence and in some cases, especially when the dimension of the parameter vector ξ is relatively small, can outperform the straightforward Monte Carlo methods (e.g., Niederreiter 1992). However, in principle it is not possible to evaluate multidimensional integrals with a high precision.

There are CLT-type results for the optimal value and optimal solutions of the SAA problems. Consider problem (7) and let $\hat{f}_N(x)$ be the sample average estimate of $f(x)$ and $\hat{\vartheta}_N$ be the optimal value of the corresponding SAA problem. Under mild regularity conditions, in particular if $F(\cdot, \xi)$ is Lipschitz continuous, then

$$\hat{\vartheta}_N = \inf_{x \in \mathcal{S}} \hat{f}_N(x) + o_p(N^{-1/2}). \tag{8}$$

Moreover, if the true problem (7) has unique optimal solution $\bar{x}$, then

$$\hat{\vartheta}_N - \vartheta^* = \hat{f}_N(\bar{x}) - f(\bar{x}) + o_p(N^{-1/2}), \tag{9}$$

and hence $N^{1/2}(\hat{\vartheta}_N - \vartheta^*)$ converges in distribution to a normal distribution with zero mean and variance $\sigma^2(\bar{x}) = \operatorname{Var}[F(\bar{x}, \xi)]$ (Shapiro 1991). For the

problems of the form (7) (i.e., without expectation constraints), it also holds that $\mathbb{E}[\hat{\vartheta}_N] \leq \vartheta^*$ and the bias $\vartheta^* - \mathbb{E}[\hat{\vartheta}_N]$ is monotonically decreasing to zero with increase of the sample size N (Norkin et al. 1998). By (9) it follows that if problem (7) has unique optimal solution $\bar{x}$, then this bias is of order $o(N^{-1/2})$ and $\hat{\vartheta}_N$ converges to ϑ^* more or less at the same rate as $\hat{f}_N(\bar{x})$ converges to $f(\bar{x})$. On the other hand, if the problem (7) has a large set $\mathcal{S}$ of optimal solutions, then the bias tends to become bigger and is of order $O(N^{-1/2})$.

Consider now problem (1) with expectation constraints. Suppose that the true problem (1) has unique optimal solution $\bar{x}$ and unique corresponding Lagrange multipliers $\bar{\lambda}_i \geq 0, \quad i = 1, .., q$. Then under certain regularity conditions, in particular if the set χ is convex and functions $F_i(\cdot, \xi)$, $i = 0, \ldots, q$, are either convex or continuously differentiable, it follows that

$$N^{1/2}(\hat{\vartheta}_N - \vartheta^*) \xrightarrow{d} \mathcal{N}(0, \sigma^2), \tag{10}$$

where $\sigma^2 = \mathrm{Var}\left[F_0(\bar{x}, \xi) + \sum_{i=1}^{q} \bar{\lambda}_i F_i(\bar{x}, \xi)\right]$ (Shapiro 1991) and $\xrightarrow{d}$ denotes convergence in distribution and $\mathcal{N}(0, \sigma^2)$ denotes normal distribution with mean 0 and variance σ^2.

There is an interesting implication of this result. In the formulation (6) of the corresponding SAA problem, the same sample is used in estimation of the objective and constraint functions. An alternative will be to employ different, independent of each other, samples for estimation of the involved functions. In the first case, the asymptotic variance of $\hat{\vartheta}_N$ is given by N^{-1} times $\mathrm{Var}\left[F_0(\bar{x}, \xi) + \sum_{i=1}^{q} \bar{\lambda}_i F_i(\bar{x}, \xi)\right]$, and this variance is equal to the sum of variances $\mathrm{Var}[F_0(\bar{x}, \xi)]$ and $\mathrm{Var}\left[\bar{\lambda}_i F_i(\bar{x}, \xi)\right]$, $i = 1, \ldots, q$, and the corresponding covariance terms. On the other hand, in the second case of independent samples, a similar formula holds but without the covariance terms. It could be expected for the covariance terms in the first case to be positive. In such situation it would be preferable to use the independent samples strategy in order to reduce variability of the SAA estimators.

Evaluation of the Sample Size and Validation of Optimality

Consider problem (7) (without expectation constraints). For an $\varepsilon > 0$ it is said that a feasible point $\bar{x} \in \chi$ is an ε -optimal solution of problem (7) if $f(\bar{x}) \leq \vartheta^* + \varepsilon$. A natural question is how large should be the sample size N to ensure that an ε' -optimal solution of the corresponding SAA problem is an ε-optimal solution of the true problem. Recall that the SAA method is not an algorithm; the constructed SAA problem still has to be solved numerically. Clearly the computational effort in solving the SAA problems grows with increase of the sample size N. (For convex problems coupled with good algorithms, this computational effort is more or less proportional to N.) Therefore, this question is directly related to computational complexity of stochastic programming problems.

Suppose for the moment that the feasible set χ is finite (i.e., the true problem is discrete), although its cardinality $|\chi|$ can be very large. The following estimate of the required sample size is given in Kleywegt et al. (2001): for $\varepsilon > 0$, $\varepsilon' \in [0, \varepsilon)$, $\alpha \in (0, 1)$ and sample size N satisfying

$$N \geq \frac{2\sigma^2}{(\varepsilon - \varepsilon')^2} \ln\left(\frac{|\chi|}{\alpha}\right), \tag{11}$$

it follows with probability at least $1 - \alpha$ that any ε'-optimal solution of the SAA problem is an ε-optimal solution of the true problem, i.e., for any sample size satisfying (11), there is a guarantee with confidence $1 - \alpha$ that by solving the SAA problem with accuracy $\varepsilon' < \varepsilon$, an ε-optimal solution of the true problem is recovered.

The constant σ^2 in (11) measures, in a sense, variability of the objective function $F(x, \xi)$. An important feature of the estimate (11) is that the cardinality of the set χ and significance level α are under the logarithm sign. This indicates that the required sample size is not very sensitive to increase of the cardinality of the considered combinatorial problem and a desirable confidence level. On the other hand, for say $\varepsilon' = \varepsilon/2$, the sample size N is of order $O(\varepsilon^{-2})$. Such dependence of the sample size on the required accuracy is unavoidable for Monte Carlo sampling estimates. This type of sample size estimates can be extended to general (bounded) sets $\chi \subset \mathbb{R}^n$ with similar conclusions (Shapiro (2001); see also Shapiro et al. (2009) for a discussion of such estimates).

Although important from the theoretical point of view, sample size estimates of the type (11) are far

too conservative for practical applications. The following procedure for validation of optimality of a candidate feasible point $\bar{x} \in \chi$, and hence for controlling the corresponding sample size, was suggested in Norkin et al. (1998) and developed in Mak et al. (1999). Since $\bar{x} \in \chi$ is a feasible point of the problem (7), it follows that $f(\bar{x}) \geq \vartheta^*$. Value $f(\bar{x})$ can be estimated by a direct Monte Carlo sampling.

An iid random sample ξ^j, $j = 1, \ldots, N'$, is generated, and $f(\bar{x})$ is estimated by the average $\hat{f}_{N'}(\bar{x})$. Note that the employed sample should be independent of sample(s) used in construction of the candidate solution $\bar{x}$. Also since it does not require solving large optimization problems, the sample size N' can be relatively large here. At the same time, the sample variance

$$\hat{\sigma}^2(\bar{x}) := \frac{1}{N'-1} \sum_{j=1}^{N'} \left(F(\bar{x}, \xi^j) - \hat{f}_{N'}(\bar{x})\right)^2$$

is computed. Consequently the upper bound $\hat{f}_{N'}(\bar{x}) + z_\alpha \hat{\sigma}(\bar{x})/\sqrt{N'}$ of the corresponding confidence interval gives an upper bound, with confidence of $1 - \alpha$, for the value $f(\bar{x})$, and hence for the optimal value ϑ^*, where z_α is the $(1-\alpha)$-quantile of the standard normal distribution.

Construction of a lower bound for the optimal value is based on the inequality $\mathbb{E}[\hat{\vartheta}_N] \leq \vartheta^*$, where the expectation $\mathbb{E}[\hat{\vartheta}_N]$ can be estimated by averaging optimal values of several SAA problems. Let $\hat{\vartheta}_N^1, \ldots, \hat{\vartheta}_N^M$ be optimal values of SAA problems based on independent samples each of size N. Then $\hat{v}_{N,M} := M^{-1} \sum_{m=1}^{M} \hat{\vartheta}_N^m$ is an unbiased estimate of $\mathbb{E}[\hat{\vartheta}_N]$, and $\hat{v}_{N,M} - t_{\alpha,M-1}\, \hat{\sigma}_{N,M}/\sqrt{M}$ can be used as a lower bound for ϑ^*, where $\hat{\sigma}^2_{N,M}$ is the sample variance of $\hat{\vartheta}_N^1, \ldots, \hat{\vartheta}_N^M$, and $t_{\alpha,M-1}$ is the $(1-\alpha)$-critical value of t-distribution with $M-1$ degrees of freedom (since M is typically not large, say in the range of 5–10, critical values of t-distribution, rather than standard normal, are used here). This procedure requires solving SAA problems M times, which involves considerable additional computational effort. For some ideas of reducing the computational burden of solving several SAA problems, see Bayraksan and Morton (2006).

Consider now problem (1) (with expectation constraints), and the corresponding Lagrangian $L(x, \lambda) := f_0(x) + \sum_{i=1}^{q} \lambda_i f_i(x)$. Of course,

$$\vartheta^* = \inf_{x \in \chi} \sup_{\lambda \geq 0} L(x, \lambda),$$

and hence $\vartheta^* \geq \inf_{x \in \chi} L(x, \bar{\lambda})$ for any $\bar{\lambda} \geq 0$. The equality $\vartheta^* = \inf_{x \in \chi} L(x, \bar{\lambda})$ holds, under mild regularity conditions, if the problem is convex and $\bar{\lambda}$ is a Lagrange multipliers vector, given by an optimal solution of the dual problem. By fixing $\bar{\lambda} \geq 0$ and solving SAA problems associated with the minimization of $L(x, \bar{\lambda})$ over $x \in \chi$, it is possible in a way described above to construct a lower bound for the optimal value of the true problem (1). In order to construct an upper bound for ϑ^*, it is needed to find, say with confidence $1 - \alpha$, a feasible point $\bar{x}$ of the true problem and then to estimate value $f(\bar{x})$ of the objective function.

Concluding Remarks

It is possible to apply the SAA method to chance-constrained problems. For a generated sample $\xi^1, \ldots, \xi^N$, the probability $p(x) := \Pr\{G(x, \xi) \leq 0\}$ can be estimated by the average $\hat{p}_N(x) := N^{-1} \sum_{j=1}^{N} H(x, \xi^j)$, where function $H(x, \xi)$ is defined in (4), i.e., $\hat{p}_N(x)$ is given by frequency of the event "$G(x, \xi^j) \leq 0$", $j = 1, \ldots, N$. Consequently the chance constraint (3) can be approximated by the constraint $\hat{p}_N(x) \geq 1 - \gamma$. The confidence level $1 - \gamma$ of the SAA problem does not need to be the same as for the true problem, i.e., γ does not need to be equal to α. The constructed SAA problem could be a difficult combinatorial problem. Recently some progress was made in solving such type of problems (Luedtke and Ahmed 2008). It is also possible to construct statistical upper and lower bounds for optimal values of chance constrained problems (Nemirovski and Shapiro 2006).

The SAA method can also be applied in a dynamic setting to multistage stochastic programming problems. However, the complexity of constructed SAA problems, in terms of the number generated scenarios, grows exponentially with increase of the number of stages (Shapiro and Nemirovski 2005; Shapiro 2006).

This poses a question whether multistage stochastic programming problems could be solved with a reasonable accuracy by randomization techniques.

See

- ▶ Monte Carlo Methods
- ▶ Monte Carlo Simulation
- ▶ Simulation Optimization
- ▶ Stochastic Programming
- ▶ Variance Reduction Techniques in Monte Carlo Methods

References

Bayraksan, G., & Morton, D. P. (2006). Assessing solution quality in stochastic programs. *Mathematical Programming, 108*, 495–514.

Charnes, A., Cooper, W. W., & Symonds, G. H. (1958). Cost horizons and certainty equivalents: An approach to stochastic programming of heating oil. *Management Science, 4*, 235–263.

Fishman, G. S. (1999). *Monte Carlo, concepts, algorithms and applications*. New York: Springer.

Geyer, C. J., & Thompson, E. A. (1992). Constrained Monte Carlo maximum likelihood for dependent data. *Journal of the Royal Statistical Society: Series B, 54*, 657–699 (with discussion).

Kleywegt, A. J., Shapiro, A., & Homem-de-Mello, T. (2001). The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization, 12*, 479–502.

Luedtke, J., & Ahmed, S. (2008). A sample approximation approach for optimization with probabilistic constraints. *SIAM Journal on Optimization, 19*, 674–699.

Mak, W. K., Morton, D. P., & Wood, R. K. (1999). Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters, 24*, 47–56.

Nemirovski, A., & Shapiro, A. (2006). Convex approximations of chance constrained programs. *SIAM Journal on Optimization, 17*, 969–996.

Niederreiter, H. (1992). *Random number generation and quasi-Monte Carlo methods*. Philadelphia: SIAM.

Norkin, V. I., Pflug, G. C., & Ruszczyński, A. (1998). A branch and bound method for stochastic global optimization. *Mathematical Programming, 83*, 425–450.

Prékopa, A. (1995). *Stochastic programming*. Boston: Kluwer Academic Publishers.

Robinson, S. M. (1996). Analysis of sample-path optimization. *Mathematics of Operations Research, 21*, 513–528.

Rubinstein, R. Y., & Shapiro, A. (1990). Optimization of static simulation models by the score function method. *Mathematics and Computers in Simulation, 32*, 373–392.

Shapiro, A. (1991). Asymptotic analysis of stochastic programs. *Annals of Operations Research, 30*, 169–186.

Shapiro, A. (2001). Monte Carlo approach to stochastic programming. In B. A. Peters, J. S. Smith, D. J. Medeiros, & M. W. Rohrer (Eds.), *Proceedings of the 2001 Winter Simulation Conference*, 428–431.

Shapiro, A. (2006). On complexity of multistage stochastic programs. *Operations Research Letters, 34*, 1–8.

Shapiro, A., & Nemirovski, A. (2005). On complexity of stochastic programming problems. In V. Jeyakumar & A. M. Rubinov (Eds.), *Continuous optimization: Current trends and applications* (pp. 111–144). New York: Springer.

Shapiro, A., Dentcheva, D., & Ruszczyński, A. (2009). *Lectures on stochastic programming: Modeling and theory*. Philadelphia: SIAM.

Sand Table Battle Model

A model or game with a physical representation of the geography and units. The classical sand table used sand because it could be molded into a model of the terrain's relief. The tin soldiers were used to represent the troops.

See

- ▶ Battle Modeling

Satisficing

In a decision problem, the selection by the decision maker (DM) of a satisfactory alternative as opposed to the selection of an "optimal" alternative. Here, the DM sets aspiration levels or acceptable levels on the outcomes and chooses the (first) alternative that satisfies these levels. This compromise selection is due to the DM's inability to encompass all the complexities of the decision problem and/or lack of a method that can determine an optimal solution. The concept is due to Herb Simon (1955, 1957).

See

- ▶ Bounded Rationality
- ▶ Choice Theory
- ▶ Decision Analysis
- ▶ Decision Maker (DM)

- ▶ Decision Problem
- ▶ Goal Programming
- ▶ Multiple Criteria Decision Making

References

Simon, H. A. (1955). A behavioral model of rational choice. *Quarterly Journal of Economics, 69*, 99–118.
Simon, H. A. (1957). *Models of man: Social and rational*. New York: John Wiley & Sons.

Scaling

The pre-solution transformation of the data of a problem that attempts to make the magnitudes of all the data as close as possible. Such scaling is important for mathematical-and linear-programming problems as it helps to reduce roundoff error. Most mathematical-programming systems have a SCALE command that automatically adjusts the magnitudes of the data in the rows and columns. This can be done by multiplying the technological coefficient matrix $\boldsymbol{A}$ by suitable row and column transformation matrices. A frequently used scaling algorithm is to divide each row by the largest absolute element in it, and then divide each resulting column by the largest absolute element in it. This ensures that the largest absolute value in the matrix is 1.0 and that each column and row has at least one element equal to 1.0.

Scenario

The set of conditions and characteristics that define the situation or environment under which a system or policy has to perform. There is often a baseline scenario (what will happen if trends continue) and an ideal scenario (what future one would like to have). In stochastic programming, a scenario represents a possible future uncertain outcome (or sample path).

See

- ▶ Battle Modeling
- ▶ Forecasting
- ▶ Sensitivity Analysis
- ▶ Stochastic Programming

Scenario Analysis

- ▶ Stochastic Programming

SCERT

Synergistic, Contingency Evaluation and Response Technique, which uses a systematic approach for the identification and articulation of the risks to which a project is subject and the uncertainties and contingencies which might significantly affect the outcome of the project.

See

- ▶ Network Planning
- ▶ PERT

Schedule Recovery

- ▶ Airline Industry Operations Research

Scheduling and Sequencing

Nicholas G. Hall[1] and Michael Magazine[2]
[1]The Ohio State University, Columbus, OH, USA
[2]University of Cincinnati, Cincinnati, OH, USA

Introduction

Scheduling is the allocation of limited resources over time to perform a given set of jobs or activities. The focus here is on scheduling models with applications to factory and computer systems. Other common uses of the term scheduling include:

1. Project scheduling – the determination of activity times and project duration for complex projects composed of multiple activities with precedence relations;
2. Workforce scheduling – the determination of the number of workers and their duty cycles to meet certain labor restrictions; and

3. Timetabling – the determination of the matching of participants with each other and with resources, such as sports scheduling or student/room exam assignments.

Scheduling problems have been studied informally for centuries. The Gantt Chart, developed in World War I for logistics purposes, is a graphical representation of tasks and resources over time, and was the first formal model used for scheduling purposes. Critical path methods were developed after World War II for project management and are still widely used. The 1950s saw the first analyses of machine scheduling problems using mathematical models. Further research interest was provided by the computational complexity paradigm of the early 1970s. More recently, modern manufacturing environments and supply chain coordination issues have attracted the attention of scheduling researchers.

The reason for this continuing activity is the amount of time and value that manufacturing organizations necessarily invest in the scheduling function. This function typically takes place at the operational level, after completion of the planning phases concerned with which tasks are to be performed and which resources are to be made available. Examples of such planning phases include forecasting, aggregate planning, inventory control and materials requirements planning.

In spite of all the research in this area, there is little unifying theory (McKay et al. 1988; Pinedo 1995). This is apparently because of the very difficult and diverse mathematical structure of these problems, due for example to the wide variety of special constraints that arise in applications. Nevertheless, there are some results that have had a major impact. These, along with some of the basic tools of scheduling will be described, following some useful definitions.

Textbooks and surveys reviewing the scheduling literature include Conway et al. (1967); French (1982); Lawler et al. (1993); Morton and Pentico (1993); Baker (1995); Pinedo (1995); Pinedo and Chao (1998); and Potts and Strusevich (2009).

Preliminaries

Scheduling involves the determination of two types of decisions:

- Allocation decisions – which resources (here called machines) will be assigned to perform each of a given set of jobs; and
- Sequencing decisions – in what order and when are each of these jobs performed.

Although many applications involve the determination of a schedule that is simply feasible with respect to scarce resources, most mathematical models use some economic objective that requires comparison of different schedules.

These objectives typically represent some measure of throughput, customer satisfaction or costs. In general, the problems focus on a single objective at a time, although there are some results on multicriteria problems (Hoogeveen 2005).

The constraints or considerations that are relevant in these models include precedence relations among jobs, job priorities, setup times on machines, and preemption capabilities. McKay et al. (1988) identified over 600 types of constraints present in manufacturing environments. The machine environment also affects the ability to solve these problems. Considered here are single-machine problems, several machines in parallel, flow shops that involve several machines in series, and job shops that involve several machines through which jobs follow various routings.

Also influencing the difficulty of solving these problems and the techniques that can be used to solve them is the precision in the problem data. Most models and results assume that the data is deterministic, although there exist some results for stochastic or probabilistic environments (Righter 1994), as well as some models that protect against uncertainty in the data (Daniels and Kouvelis 1995).

Solution Techniques

Since most scheduling problems have economic objectives and constraints, it is natural to use optimization methods to solve these problems. However, the number of feasible schedules grows quickly with problem size, making almost all such problems extremely difficult to solve. Consider the simplest problem with n jobs on a single machine, with no additional constraints. There are $n!$ sequences that must be considered, and even for relatively small problems, for example of size $n = 100$, the enumeration task is virtually impossible. In the mid 1970s, many scheduling problems were classified by complexity theory as intractable. More formally, as optimization problems, they are

NP-hard (Garey and Johnson 1979), and thus optimal solution procedures that are substantially more efficient than enumeration are unlikely to exist.

Traditional optimization techniques such as mathematical programming, enumeration methods, and dynamic programming have had limited success. Quite often, it is necessary to resort to heuristic techniques, which do not guarantee an optimal solution. These techniques have been of two types: limit the space of search by only considering schedules that meet some specified criteria, or search in a limited neighborhood of some known feasible schedule. These heuristics have shown some success, but unfortunately much of this work considers only their worst-case performance, which is not necessarily a true measure of the average performance of a heuristic. Recent advances in heuristics, including constraint satisfaction (Fox and Smith 1984), simulated annealing (Matsuo et al. 1989), tabu search (Widmer and Hertz 1989) and genetic algorithms (Storer et al. 1992) have extended the ability to find good solutions. Anderson et al. (1997) provided a comprehensive discussion of local search methods for scheduling problems.

Surprisingly, it is a simple principle that has given some of the most significant scheduling results. This principle is pairwise interchange (Baker 1995). This neighborhood search technique starts with a feasible schedule for the problem and interchanges the sequence of two already scheduled jobs according to some rule. For some classes of simple problems, this procedure always results in an optimal solution. For other classes, it routinely results in a computationally good, although not necessarily optimal, solution.

In problems with stochastic data, some of these same combinatorial techniques can be used. In some simple problems, it is possible to take advantage of queueing theory results or of stochastic analogs of the pairwise interchange rules discussed above. However, in most cases, simulation becomes the technique of choice. Often, the complicated environment or simply the highly combinatorial nature of the problem is enough to make simulation the only viable method of analysis.

Scheduling Results

Next some results are described, which are key building blocks to scheduling theory and applications. These arise from problems with a single, continuously available machine; with independent, one-operation, and deterministic jobs; and with machine setup times that are negligible or are not sequence-dependent.

Single-Machine Models – it is assumed that there are n jobs with the following data:

p_j = the processing time required by job j;

r_j = the ready time of job j, i.e., the earliest time at which the job can begin processing;

d_j = the time job j is due to have processing completed.

Once a schedule has been determined, the following variables can be evaluated:

C_j = the completion time of job j;

F_j = the flowtime of job $j = C_j - r_j$;

L_j = the lateness of job $j = C_j - d_j$;

T_j = the tardiness of job $j = \max\{0, L_j\}$.

The simplest models assume criteria that are nondecreasing in the completion time of jobs, i.e., in all the above variables. These are called regular measures, and include:

$$\text{Total Flowtime} = \sum_{j=1}^{n} Fj;$$

$$\text{Total Lateness} = \sum_{j=1}^{n} Lj;$$

$$\text{Maximum Lateness} = L_{\max};$$

$$\text{Total Tardiness} = \sum_{j=1}^{n} T_j; \text{ and}$$

$$\text{Maximum Tardiness} = T_{\max}.$$

In these problems, optimal schedules exist in which job preemption and machine idle time do not occur. Thus, a solution is completely characterized by a sequence of the jobs.

The following is an important result in scheduling theory (Smith 1956): *Shortest Processing Time (SPT) sequence*, i.e., *sequencing the jobs from shortest to longest processing time, minimizes total flowtime when* $r_j = 0$ *for all jobs.*

This is proved by interchange arguments and can be extended to include jobs of different importance, or weights, by using weight-to-processing-time ratio.

For customer service measures, due date information is included. Perhaps counterintuitively, SPT also minimizes Total Lateness. The most important result using due date measures is:

Earliest Due Date (EDD), i.e., sequencing the jobs from earliest to latest due date, minimizes both $L_{\max}$ and $T_{\max}$ (Jackson 1955).

In many other problems with due-date measures, EDD provides useful guidance towards an optimal, or close to optimal, solution.

There are several other single machine results. The presence of nonzero ready times creates problems with due date measures unless jobs can be preempted without penalty. If this is not the case, it is necessary to look ahead to jobs not yet in the system, causing simple dispatching rules that do not consider the current partial sequence, such as SPT or EDD, to yield suboptimal solutions.

Another condition imposed on jobs may be precedence relations. They represent a partial ordering of the jobs, which is imposed for technological reasons. These relationships (which can be represented by graphs where job i precedes job j if there is an arc from i to j), may under certain conditions still yield relatively simple algorithms. When the graph is composed of chains or is series–parallel, many problems can be solved optimally. However, general precedence relationships between jobs typically prevent the finding of optimal solutions quickly (Monma 1981).

As noted, these simple models ignore setup times. When setup times are present and sequence-dependent, many simple problems become very difficult to solve. For example, minimizing makespan, i.e., $C_{\max}$ = completion time of the last job, is equivalent to solving a traveling salesman problem.

Thus far only regular measures of performance were considered. One important non-regular measure that represents another attribute of customer service is a job earliness penalty, where a job's earliness is $E_j = \max\{0, d_j - C_j\}$. When there are many jobs and a common due date, minimizing a combination of earliness and tardiness penalties is still possible (Baker and Scudder 1990; Hall and Posner 1991). When the due dates are distinct for each job, these problems are difficult to solve.

Multiple Parallel-Machine Models – Fast optimal algorithms for scheduling problems with several machines are scarce. The existence of several machines requires not only sequencing decisions, but also allocation decisions. The simplest environment assumes there are n jobs available at time zero and m identical machines for processing them.

The simplest model in this environment attempts to minimize makespan, i.e., the time to complete all n jobs. Since changing the sequence of jobs allocated to a particular machine does not affect the makespan, the only decision is allocating jobs to machines. If preemption is permitted, McNaughton (1959) showed that there is a simple algorithm to perform this allocation. When preemption is not permitted, the problem is *NP*-hard even for $m = 2$. However, a computationally reasonable (pseudo-polynomial time) algorithm provides optimal solutions.

Here, a reasonable heuristic seems to be to list the jobs in some prespecified order, placing the next job in the list onto the first machine that becomes available. By cleverly ordering the jobs, list-scheduling heuristics provide acceptable performance guarantees. For example, a simple rule such as longest processing time first (LPT) guarantees a solution within 33% of the optimal makespan (Graham 1969). Graham also found several interesting anomalies in list scheduling, such as that increasing the number of machines or reducing the processing time of the jobs can increase makespan.

If the objective is to minimize total flow time on m identical machines, an SPT list-scheduling algorithm gives an optimal solution. Unfortunately almost all other problems that can be generalized from the single machine case fail to yield optimal solutions in polynomial time.

The next important class of problems is flow shop models. The simplest instance assumes there are m machines and each job requires processing on each machine, i.e., each job has m operations. In addition, processing moves from machine to machine in a prespecified order that is the same for each job. These problems are much more difficult to solve than single-machine problems. One additional consideration is that inserted idle time may be desirable. This was not the case for single machines with regular performance measures. In addition, if there are more than three machines, different permutations of the jobs must be considered for each machine, giving rise to $(n!)^m$ possible sequences.

When $m = 2$, however, Johnson (1954) provided an efficient algorithm for minimizing makespan. When there are more than two machines, the makespan problem becomes *NP*-hard. There are several special cases, however, when $m = 3$ that use variations of Johnson's algorithm to guarantee optimal solutions.

Johnson's theorem is stated using his notation. Here,

A_i = the processing-time (including setup, if any) of the first operation of the ith job;

B_i = the processing-time (including setup, if any) of the second operation of the ith job.

Johnson's algorithm is applied to the following problem: sequence an arbitrary number of jobs in a two-machine flow shop to minimize the makespan. It is assumed that all jobs are simultaneously available.

Johnson's Theorem

An optimal schedule for a two-machine flow shop that minimizes the make-span is obtained when job j precedes job $j + 1$if

$$\min\{A_j, B_{j+1}\} \leq \min\{A_{j+1}, B_j\}.$$

Under certain conditions, the processing of an operation of job j can start before the completion of the previous operation. This overlapping of operations can improve the makespan. Trietsch and Baker (1993) discussed a variety of these lot-streaming problems.

One variation of the flow shop model requires that the processing of a job cannot be interrupted once it has started. A survey of applications, algorithms and complexity results for these no-wait models, and for related models with machine blocking, was given by Hall and Sriskandarajah (1996). Another extension of the flow shop model allows the job to pass through the machines in any order. This is referred to as an open shop and, except for the case of two machines with makespan objective, the problems are essentially all very hard to solve optimally.

Job shop scheduling models cover a significant portion of all factory scheduling problems. In a job shop, the number and order of operations for each job may differ. These problems are virtually intractable because of the large number of possible schedules. The notorious 10-job, 10-machine problem of Fisher and Thompson (1963) provided the benchmark for the computational difficulty of these problems. It took 25 years of research for its optimal solution to be verified (Carlier and Pinson 1988). Many heuristic procedures use dispatching rules. These heuristics choose, according to some rule, the next job from those available to start on a machine when that machine becomes idle. A successful and widely used heuristic for the job shop problem was described by Adams et al. (1988).

Research Directions

Scheduling Families or Groups – Modern manufacturing facilities contain flexible machines that can produce or assemble a variety of products. When products are similar, switching between them requires no setup time. These groups of products are called product families. Switching between different product families is possible, but requires a setup. Here a batch represents a set of items that are produced following a single setup. The difficulty in scheduling these environments arises from the tradeoff between scheduling large batches, which cause delays to jobs in other families, and scheduling small batches, which incur many setups. The issues to be resolved include sequencing items within families, the determination of the batch sizes and the sequencing of the batches from different families (Monma and Potts 1989; Santos and Magazine 1985). A useful survey is provided by Potts and Kovalyov (2002).

Modern Manufacturing Environments – The requirements of modern manufacturing impose new demands on the scheduling function and the theory that supports it. Lee et al. (1992) considered burn-in problems that arise in semiconductor manufacturing, where the processing time required by a batch of jobs is the length of the longest job rather than the total processing time. Automated manufacturing systems, particularly those which require the coordination of computer-controlled material handling devices with production schedules, generate a variety of interesting scheduling problems, as discussed by Crama (1997). For example, Hall et al. (1997) considered robotic cells, in which a robot serves several production machines. Also in accordance with modern manufacturing principles are manufacturing environments with limited storage buffers (Hall et al. 1998). Deterministic scheduling, with an emphasis on modern manufacturing problems, is extensively reviewed by Lee et al. (1997).

Online Scheduling – In various practical situations, information about arriving jobs is not know at the start of the planning horizon. In online scheduling, this information is revealed

over time. A standard performance measure for online algorithms is the competitive ratio, which bounds the performance ratio of online and optimal offline schedules. Online scheduling models have application to purchasing and ordering systems that use the Internet. Pruhs et al. (2004) provide a thorough review of the research in this area. One of the key ideas in online scheduling is to delay the processing of jobs until more information about the future is available. Hall et al. (2009) consider an environment where jobs can only arrive at known future times. This environment interpolates between the classical offline and online environments.

Supply Chain Scheduling – One of the most active research areas within operations management starting in the 1990s has been supply chain management, i.e., the consideration of integration and coordination issues, and incentives, within manufacturing systems. Much of this research has been strategic in nature, but the supply chain scheduling area focuses on operational level decisions. Examples of this research include the coordination of manufacturing and distribution (Hall and Potts 2003), and the coordination of component delivery from multiple suppliers (Chen and Hall 2007). Issues that are discussed include the classical one of computational solvability, evaluation of the cost of conflict if one powerful supply chain member imposes their preferred schedule on others, and evaluation of the benefit of cooperation that results if all supply chain members agree on a common schedule. A comprehensive survey of related results is provided by Chen (2010).

Scheduling with Machine Availability Constraints – Although classical scheduling models typically assume that all processing resources are continuously available from the start of the planning horizon, there are practical situations where this assumption is incorrect. For example, resources may be allocated to outsourcing contracts at particular times, or downtime for maintenance may be planned in advance. Unavailable times may either be known in advance or not. Moreover, jobs that are interrupted by resource unavailability may either be resumable, or may need to be restarted. As the surveys by Schmidt (2000) and Lee (2004) reveal, most scheduling problems in this area are intractable. Consequently, the design of approximation algorithms and approximation schemes (Ng and Kovalyov 2004) is an active research area.

Safe Scheduling – Safe scheduling (Baker and Trietsch 2009) is an approach to stochastic scheduling problems that explicitly considers the role of safety time to meet service levels. These problems are analogous to safety stocks in inventory problems. A key element of safe scheduling is the inclusion of service levels, defined as the probability that a job completes by its due date. Safety time is the difference between the expected completion time and the due date. Safe scheduling problems are formulated with either explicit service level constraints or as an objective function that considers costs associated with not meeting due dates. To completely specify the problem either decisions have to be made as to which jobs (with known release date and due date) to choose or to decide on release dates and due dates and minimizes total cost. Results are often not analogous to the deterministic version of the problem.

See

- ▶ Computational Complexity
- ▶ Critical Path Method (CPM)
- ▶ Flexible Manufacturing Systems
- ▶ Genetic Algorithms
- ▶ Heuristics
- ▶ Integer and Combinatorial Optimization
- ▶ Inventory Modeling
- ▶ Job Shop Scheduling
- ▶ Metaheuristics
- ▶ Operations Management
- ▶ Production Management
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Adams, J., Balas, E., & Zawack, D. (1988). The shifting bottleneck procedure for job shop scheduling. *Management Science, 34*, 391–401.

Anderson, E. J., Glass, C. A., & Potts, C. N. (1997). Machine scheduling. In E. H. L. Aaarts & J. K. Lenstra (Eds.), *361–414 in Local Search in Combinatorial Optimization*. Chichester, UK: Wiley.

Baker, K. (1995). *Elements of sequencing and scheduling* (Rev. ed.). Hanover, NH: Amos Tuck School of Business Administration, Dartmouth College.

Baker, K., & Scudder, G. (1990). Sequencing with earliness and tardiness penalties: A review. *Operations Research, 38*, 22–36.

Baker, K., & Trietsch, D. (2009). *Principles of sequencing and scheduling*. Hoboken, NJ: Wiley.

Bertsekas, D. P. (1987). *Dynamic programming: Deterministic and stochastic models*. Englewood Cliffs, NJ: Prentice Hall.

Blazewicz, J., Cellary, W., Slowinski, R., & Weglarz, J. (1986). Scheduling under resource constraints – Deterministic models. *Annals of Operations Research*, *7*.

Carlier, J., & Pinson, E. (1988). An algorithm for solving the job-shop problem. *Management Science, 35*, 164–176.

Chen, Z.-L. (2010). Integrated production and outbound distribution scheduling: Review and extensions. *Operations Research, 58*, 130–148.

Chen, Z.-L., & Hall, N. G. (2007). Supply chain scheduling: Conflict and cooperation in assembly systems. *Operations Research, 55*, 1072–1089.

Conway, R., Maxwell, W., & Miller, L. (1967). *Theory of scheduling*. Reading, MA: Addison-Wesley.

Crama, Y. (1997). Combinatorial models for production scheduling in automated manufacturing systems. *European Journal of Operational Research, 99*, 136–153.

Daniels, R., & Kouvelis, P. (1995). Robust scheduling to hedge against processing time uncertainty in single stage production. *Management Science, 41*, 363–376.

Fisher, H., & Thompson, G. (1963). Probabilistic learning combinations of local job-shop scheduling rules. In J. Muth & G. Thompson (Eds.), *Industrial scheduling* (pp. 225–251). Englewood Cliffs, NJ: Prentice-Hal.

Fox, M. S., & Smith, S. F. (1984). ISIS: A knowledge-based system for factory scheduling. *Expert Systems, 1*, 25–49.

French, S. (1982). *Sequencing and scheduling: An introduction to the mathematics of the job shop*. Chichester, UK: Harwood.

Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: A guide to the theory of NP-completeness*. San Francisco: W.H. Freeman & Company.

Graham, R. (1969). Bounds on multiprocessor timing anomalies. *SIAM Journal on Applied Mathematics, 17*, 416–425.

Hall, N. G., Kamoun, H., & Sriskandarajah, C. (1997). Scheduling in robotic cells: Classification, two and three machine cells. *Operations Research, 45*, 421–439.

Hall, N. G., & Posner, M. E. (1991). Earliness-tardiness scheduling problems, I: Weighted deviation of completion times about a common due date. *Operations Research, 39*, 847–856.

Hall, N. G., Posner, M. E., & Potts, C. N. (1998). Scheduling with finite capacity output buffers. *Operations Research, 46*, S84–S97.

Hall, N. G., Posner, M. E., & Potts, C. N. (2009). Online scheduling with known arrival times. *Mathematics of Operations Research, 34*, 92–102.

Hall, N. G., & Potts, C. N. (2003). Supply chain scheduling: Batching and delivery. *Operations Research, 51*, 566–584.

Hall, N. G., & Sriskandarajah, C. (1996). A survey of machine scheduling problems with blocking and no-wait in process. *Operations Research, 44*, 510–525.

Hoogeveen, H. (2005). Multicriteria scheduling. *European Journal of Operational Research, 167*, 592–623.

Jackson, J. R. (1955). *Scheduling a production line to minimize maximum tardiness*. Research Report 43, Management Science Research Project, University of California, Los Angeles.

Johnson, S. (1954). Optimal two and three stage production schedules with setup times included. *Naval Research Logistics Quarterly, 1*, 61–68.

Lawler, E., Lenstra, J., Rinnooy Kan, A., & Shmoys, D. (1993). Sequencing and scheduling: Algorithms and complexity. In S. Graves, A. Rinnooy Kan, & P. Zipkin (Eds.), *Handbooks in operations research and management science* (Logistics of production and inventory, Vol. 4). Amsterdam: North-Holland.

Lee, C.-Y. (2004). Machine scheduling with availability constraints. In J. Y.-T. Leung (Ed.), *Handbook of scheduling: Algorithms, models and performance analysis* (pp. 22-1–22-13). Boca Raton: Chapman & Hall/ CRC.

Lee, C.-Y., Li, L., & Pinedo, M. (1997). Current trends in deterministic scheduling. *Annals of Operations Research, 70*, 1–42.

Lee, C.-Y., Uzsoy, R., & Martin-Vega, L. A. (1992). Efficient algorithms for scheduling semiconductor burn-in operations. *Operations Research, 40*, 764–775.

Matsuo, H., Suh, C. J., & Sullivan, R. S. (1989). A controlled search simulated annealing method for the single machine weighted tardiness problem. *Annals of Operations Research, 21*, 85–108.

McKay, K., Safayeni, F., & Buzacott, J. (1988). Job-shop scheduling theory: What is relevant. *Interfaces, 18*(4), 84–90.

McNaughton, R. (1959). Scheduling with deadlines and loss functions. *Management Science, 6*, 1–12.

Monma, C. L. (1981). Sequencing with general precedence constraints. *Mathematics of Operations Research, 4*, 215–224.

Monma, C. L., & Potts, C. N. (1989). On the complexity of scheduling with batch setup times. *Operations Research, 37*, 798–804.

Morton, T., & Pentico, D. (1993). *Heuristic scheduling systems*. New York: Wiley.

Ng, C. T., & Kovalyov, M. Y. (2004). An FPTAS for scheduling a two-machine flowshop with one unavailability interval. *Naval Research Logistics, 51*, 307–315.

Pinedo, M. (1995). *Scheduling: Theory, algorithms, and systems*. Englewood Cliffs, NJ: Prentice Hall.

Pinedo, M., & Chao, X. (1998). *Operations scheduling: Applications in manufacturing and services*. Burr Ridge, IL: McGraw-Hill.

Potts, C. N., & Kovalyov, M. Y. (2002). Scheduling with batching: A review. *European Journal of Operational Research, 120*, 228–249.

Potts, C. N., & Strusevich, V. A. (2009). Fifty years of scheduling: A survey of milestones. *Journal of the Operational Research Society, 60*, S41–S68.

Pruhs, K., Sgall, J., & Torng, E. (2004). Online scheduling. In J. Y.-T. Leung (Ed.), *Handbook of scheduling: Algorithms, models and performance analysis* (pp. 15-1–15-43). Boca Raton: Chapman & Hall/ CRC.

Righter, R. (1994). Stochastic scheduling. In M. Shaked & G. Shanthikumar (Eds.), *Stochastic orders*. San Diego, CA: Academic Press.

Santos, C., & Magazine, M. (1985). Batching in single operation manufacturing systems. *Operations Research Letters, 4*, 99–103.

Schmidt, G. (2000). Scheduling with limited machine availability. *European Journal of Operational Research, 121*, 1–15.

Smith, W. (1956). Various optimizers for single stage production. *Naval Research Logistics Quarterly, 3*, 59–66.
Storer, R. H., Wu, S. D., & Vaccari, R. (1992). New search spaces for sequencing problems with application to job shop scheduling. *Management Science, 38*, 1495–1509.
Trietsch, D., & Baker, K. (1993). Basic techniques for lot streaming. *Operations Research, 41*, 1065–1076.
Widmer, M., & Hertz, A. (1989). A new heuristic method for the flow shop sequencing problem. *European Journal of Operational Research, 41*, 186–193.

Score Functions

Reuven Y. Rubinstein[1], Alexander Shapiro[2] and Stanislav Uryasev[3]
[1]Technion – Israel Institute of Technology, Haifa, Israel
[2]Georgia Institute of Technology, Atlanta, GA, USA
[3]University of Florida, Gainesville, FL, USA

Introduction

Many complex real world systems can be modeled as discrete-event systems (DES). Examples are computer-communication networks, flexible manufacturing systems, probabilistic fracture mechanics models, PERT-project networks and flow networks. In view of the complex interactions within a DES, they are typically studied via stochastic simulation.

In the design and analysis of a DES, the interest is not only in performance evaluation, but also in sensitivity analysis and optimization. Consider for example manufacturing systems. Here (1) the performance measure may be the average waiting time of an item to be processed at several work stations (robots) according to a given schedule and route; (2) the sensitivity and decision parameters may be the average rate at which the work-stations (robots) process the item. In such a system, the goal may be to minimize the average makespan consisting of the processing time and delay time with allowance for some constraints (e.g., cost).

Alternatively, consider failure probability models for mechanical passive components (such as pipes or vessels). Here (1) the performance measure may be cumulative failure or leakage probability of a passive component over some time; (2) the sensitivity and decision parameters may be the geometry of the component (thickness of the walls), fracture toughness of the material and stress intensity factors. In such a system, failure probability is a function of the sizes of defects (cracks) and their time dependent stochastic development.

Methods for sensitivity analysis and optimization of a DES include infinitesimal perturbation analysis (IPA) and the score function (SF) method, also called the likelihood ratio (LR) method (Rubinstein 1976; Reiman and Weiss 1989; Glynn 1990; L'Ecuyer 1990; Ho and Cao 1991; Glasserman 1991; Rubinstein and Shapiro 1993; Asmussen and Glynn 2007). The SF method allows one to evaluate, *simultaneously* from a *single* sample path (simulation experiment) not only the performance and all its sensitivities, (gradient, Hessian, etc.), but to solve an entire optimization problem as well. An alternative approach, called analytic perturbation analysis, for calculating the sensitivities of discrete-event systems using analytical formulas for the expectations of indicator functions, has also been proposed by Uryasev (1995, 1997).

Estimation and Sensitivity Analysis of Discrete-Event Static Systems

Let $l(\boldsymbol{\theta})$ be a real-valued function represented in the form $l(\boldsymbol{\theta}) = E_{\boldsymbol{\theta}}[L(\boldsymbol{Y})]$. Here $\boldsymbol{Y}$ is a random vector whose cumulative distribution function (CDF) $F(\boldsymbol{y}, \boldsymbol{\theta})$ depends on the parameter vector $\boldsymbol{\theta} \in \ominus$ with $\ominus$ being a subset of a finite dimensional vector space, say $\ominus \subset \mathrm{R}^n$. The function $L(\boldsymbol{Y})$ can be viewed as a sample performance driven by the input vector $\boldsymbol{Y}$. The notation $E_{\boldsymbol{\theta}}$ stands for the expectation with respect to the CDF $F(\boldsymbol{y}, \boldsymbol{\theta})$.

In order to estimate the expected value $l(\boldsymbol{\theta})$ by simulation (Monte Carlo) techniques, one can proceed as follows. Generate a sample $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_N$ from N the CDF $F(\boldsymbol{y}, \boldsymbol{\theta})$ and set the sample mean $N^{-1}\sum_{i=1}^{n} L(Y_i)$ as the corresponding estimator. Of course, this procedure requires generation of a new sample every time the expectation $l(\boldsymbol{\theta})$ should be estimated for a different value of the parameter vector $\boldsymbol{\theta} \in \ominus$. In order to overcome this difficulty, consider the following procedure based on a change

of probability measure techniques. Suppose that the random vector $\boldsymbol{Y}$ has a probability density function (PDF) $f(\boldsymbol{y}, \boldsymbol{\theta})$ corresponding to the CDF $F(\boldsymbol{y}, \boldsymbol{\theta})$. For a PDF $g(\boldsymbol{y})$ such that the outcome space of $f(\cdot,\boldsymbol{\theta})$ lies in the outcome space of $g(\cdot)$:

$$l(\boldsymbol{\theta}) = \int L(z)\,[f(z, \boldsymbol{\theta})/g(z)]g(z)dz = E_g[L(Z)W(Z, \boldsymbol{\theta})], \tag{1}$$

where $W(\boldsymbol{z}, \boldsymbol{\theta}) = f(\boldsymbol{z}, \boldsymbol{\theta})/g(\boldsymbol{z})$. Note that the integration variable $\boldsymbol{y}$ was replaced by $\boldsymbol{z}$ when the involved densities were changed from $f(\cdot,\boldsymbol{\theta})$ to $g(\cdot)$, and the integrals are over the outcome spaces of the corresponding density functions. Formula (1) suggests the following way for estimating $l(\boldsymbol{\theta})$. Generate a sample $\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_N$, from the PDF $g(\boldsymbol{z})$ and estimate $l(\boldsymbol{\theta})$ by

$$\hat{l}(\boldsymbol{\theta}) = N^{-1} \sum_{i=1}^{N} L(Zi)W(Zi, \boldsymbol{\theta}) \tag{2}$$

The function $W(z, \boldsymbol{\theta})$ is called the likelihood ratio (LR) function and $g(\boldsymbol{y})$ is the dominating density. Typically one takes $g(\mathrm{z}) = \mathrm{f}(\boldsymbol{z}, \boldsymbol{\theta}_0)$ for a particular value $\boldsymbol{\theta}_0 \in \ominus$ of the parameter vector. The chosen $\boldsymbol{\theta}_0$ is referred to as the reference value of the parameter vector.

The LR function is given explicitly through the corresponding density functions and is typically smooth (differentiable) in $\boldsymbol{\theta}$. As soon as the sample $\boldsymbol{Z}_1,\ldots, \boldsymbol{Z}_N$ is generated, the obtained LR estimator $\hat{l}_N(\boldsymbol{\theta})$ becomes an analytical function of $\boldsymbol{\theta}$ and provides an estimate of the entire function (response surface) $l(\boldsymbol{\theta})$. Moreover, under mild regularity conditions ensuring interchangeability of the integration and differentiation operators, it follows that derivatives of the expected performance $l(\boldsymbol{\theta})$ can be taken inside the expected value representation given in the right-hand side of (1). Consequently, the gradient $\nabla\hat{l}_N(\boldsymbol{\theta})$ of the sample estimate, as defined in (2), provides an unbiased estimator of the corresponding gradient of $l(\boldsymbol{\theta})$. The Hessian matrix can be similarly estimated (Rubinstein and Shapiro 1993).

In particular, for $g(\cdot) = f(\cdot, \boldsymbol{\theta})$, the gradient of the LR function $W(\boldsymbol{y}, \boldsymbol{\theta})$ is called the score function (SF) and can be written in the form $\boldsymbol{S}(\boldsymbol{y}, \boldsymbol{\theta}) = \nabla \log f(\boldsymbol{y}, \boldsymbol{\theta})$. Then, given a random sample $\boldsymbol{Y}_1,\ldots, \boldsymbol{Y}_N$ from $f(y, \boldsymbol{\theta})$, the gradient $\nabla l(\boldsymbol{\theta})$ can be estimated by

$$\bar{\nabla} l_N(\boldsymbol{\theta}) = N^{-1} \sum_{i=1}^{N} L(Y_i)S(Y_{i,}\boldsymbol{\theta}). \tag{3}$$

High-order derivatives can be handled in a similar way. Note that gradient $\nabla W(\boldsymbol{z}, \boldsymbol{\theta})$ of the LR function can be written in the form $W(\boldsymbol{z}, \boldsymbol{\theta})\boldsymbol{S}(\boldsymbol{z}, \boldsymbol{\theta})$ and is called the generalized score function.

A word of caution is due. Although the generalized SF estimators typically are unbiased and consistent, their accuracy is determined by the corresponding variances and can be quite sensitive to the choice of the dominating PDF $g(\boldsymbol{y})$ (reference value $\boldsymbol{\theta}_0$). The problem of an optimal choice of $g(\boldsymbol{y})$ is closely related to the importance sampling method in simulation. A detailed discussion of this problem and relevant variance reduction techniques may be found Rubinstein and Shapiro (1993).

Example 1 (System Reliability). Consider the sample performance function:

$$L(Y) = \max_{l\le k\le p} \min_{j\in\Im} Y_j,$$

where $\Im_1,\ldots, \Im_p$ are the complete paths from a source to a sink and Y_j are durations (lifetimes) of the components in the system. Suppose that the random variables $Y_1,\ldots, Y_m$ are independent and each distributed as a gamma. Given the (vector) random sample $\boldsymbol{Y}_1,\ldots, \boldsymbol{Y}_N$, the gradient of the corresponding expected performance $l(\lambda)$ can be estimated by the SF estimator

$$\bar{\nabla} l_N(\lambda) = N^{-1} \sum_{i=1}^{N} L(\boldsymbol{Y_i})(\beta\lambda^{-1} - \boldsymbol{Y}_i).$$

Estimation and Sensitivity Analysis of Discrete-Event Dynamic Systems

The SF approach presented in the previous section can be extended to dynamic systems as well. Consider a Discrete Event Dynamic System (DEDS) driven by

an input sequence of iid random vectors $\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots$, generated from a PDF $f(\boldsymbol{y}, \boldsymbol{\theta})$ depending on the parameter vector $\boldsymbol{\theta} \in \ominus$. Let $L_1, L_2, \ldots$, be an output process driven by this input sequence. That is, $L_t = L_t(\boldsymbol{Y}_t), t = 1, 2, \ldots$, where vector $\boldsymbol{Y}_t = (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_t)$ represents a history of the input process up to time t, and $\boldsymbol{L}_t(\cdot)$ is a sequence of real valued functions.

Suppose that $\{L_t\}$ is a discrete-time regenerative process with the regenerative cycle of length τ. For example, consider a GI/G/1 queue with FIFO discipline. In that case, the input sequence is represented by the two-dimensional vector $\boldsymbol{Y}_t = (Y_{1t}, Y_{2t})$, with Y_{1t} being the service time of the tth customer, Y_{2t} being the interarrival time between the $(t-1)$stand tth customers, and τ is the number of customers served during the busy period. The output process L_t can be, for example, the system waiting time of the tth customer.

Consider the expected long-run average $l(\boldsymbol{\theta})$ of the process L_t. It is well known in the theory of regenerative processes that $l(\boldsymbol{\theta})$ is equal to the expected steady-state performance of L_t and can be represented as the ratio $l(\boldsymbol{\theta}) = l_1(\boldsymbol{\theta})/l_2(\boldsymbol{\theta})$ of the expectations $l_1(\boldsymbol{\theta}) = E_{\boldsymbol{\theta}}[\sum_{t=1}^{\tau} L_t]$ and $l_2(\boldsymbol{\theta}) = E_{\boldsymbol{\theta}}[\tau]$, respectively. Note that the above expectations and hence the expected performance $l(\boldsymbol{\theta})$ are functions of the parameter vector $\boldsymbol{\theta} \in \ominus$.

The SF approach can again be used to derive an estimate of the expected performance $l(\boldsymbol{\theta})$ and its sensitivities $\nabla l(\boldsymbol{\theta})$ for different values of $\boldsymbol{\theta}$, i.e.,, for a chosen dominating PDF $g(\boldsymbol{y})$, the expected value functions $l_1(\boldsymbol{\theta})$ and $l_2(\boldsymbol{\theta})$ can be written in the form

$$l_1(\boldsymbol{\theta}) = E_g\left[\sum_{t=1}^{t} L_t(Z_t)\tilde{w}_t(\boldsymbol{Z}_t, \boldsymbol{\theta})\right] \text{ and}$$
$$l_2(\boldsymbol{\theta}) = E_g\left[\sum_{t=1}^{t} \tilde{w}(\boldsymbol{Z}_t, \boldsymbol{\theta})\right], \tag{5}$$

where $\tilde{w}(\boldsymbol{Z}_t, \boldsymbol{\theta}) = f_t(\boldsymbol{Z}_t, \boldsymbol{\theta})/g_t(\boldsymbol{Z}_t)$ with $f_t(\mathbf{z_t}, \boldsymbol{\theta}) = \prod_{i=1}^{t} f(\mathbf{z_t}, \boldsymbol{\theta})$ and $g_t(\mathbf{z_t}) = \prod_{i=1}^{t} g(\mathbf{z_i})$. The latter term is the density function of the random vector $\boldsymbol{Z}_t = (\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_t)$, with the random vectors $\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_t$ drawn according to the PDF $g(\cdot)$.

Under standard regularity conditions, the derivatives of $l_1(\boldsymbol{\theta})$ and $l_2(\boldsymbol{\theta})$ can be taken inside the expectation. Consequently, by generating a random sample of N regenerative cycles based on the PDF $g(\cdot)$, one can estimate the above expectations by the corresponding sample means (averages), and hence can estimate $l(\boldsymbol{\theta})$ and $\nabla l(\boldsymbol{\theta})$.

Example 2 (Queueing Delays). Let L_t be the system waiting time of tth customer in a GI/G/1 queue driven by the input sequences of the service times Y_{1t} and the interarrival times Y_{2t} with respective density functions $f_1(y_1, \boldsymbol{\theta}_1)$ and $f_2(y_2, \boldsymbol{\theta}_2)$. By Lindley's equation, it follows:

$$L_t = Y_{1t} + [L_{t-1} - Y_{2t}]+, \quad t = 1, 2, \ldots$$

Let τ be the number of customers served in the first busy period. Then the expected long-run average waiting time of a customer can be written as

$$l(\boldsymbol{\theta}) = \frac{E\left[\sum_{t=1}^{\tau} L_t\right]}{E_{\boldsymbol{\theta}}[\tau]} = \frac{E\left[\sum_{t=1}^{\tau}\sum_{j=1}^{\tau} Y_{1j} - \sum_{t=2}^{\tau}\sum_{j=2}^{\tau} Y_{2j}\right]}{E_{\boldsymbol{\theta}}[\tau]},$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ and $f(\boldsymbol{y}, \boldsymbol{\theta}) = f_1(y_1, \boldsymbol{\theta}_1) f_2(y_2, \boldsymbol{\theta}_2)$. By generating N regenerative cycles (busy periods) of the service and interarrival times according to the PDF $f(z, \boldsymbol{\theta}_0)$ for a chosen value $\boldsymbol{\theta}_0$ of the parameter vector, one can estimate $l(\boldsymbol{\theta})$ and $\nabla l(\boldsymbol{\theta})$ for various values of $\boldsymbol{\theta}$.

Optimization

Consider the following (unconstrained) optimization problem involving the expected performance function of a static or dynamic system:

$$(\mathbf{P}_0) \ \text{minimize}\, l(\boldsymbol{\theta}), \quad \boldsymbol{\theta} \in \ominus$$

Let $\boldsymbol{\theta}^*$ be an optimal solution of the program $(\mathbf{P}_0)$. The optimal solution $\boldsymbol{\theta}^*$ can be estimated from a single simulation using the SF approach, i.e., let $\hat{l}_N(\boldsymbol{\theta})$ be the LR estimator of $l(\boldsymbol{\theta})$ calculated via the corresponding LR function (process). Consider the optimization problem:

$$(\hat{\mathbf{P}}_N) \ \text{minimize}\ \hat{l}_N(\boldsymbol{\theta}), \quad \boldsymbol{\theta} \in \ominus$$

where the program $(\hat{P}_{\mathrm{N}})$ is referred to as the stochastic counterpart of the program $(\mathbf{P}_0)$.

The LR estimator $\hat{l}_N(\boldsymbol{\theta})$ and, hence, the program $\hat{P}_N(\boldsymbol{\theta})$ depend on the generated random sample and in that way are stochastic. However, as soon as the sample is generated by the function $\hat{l}$, its derivatives are given explicitly through the corresponding density functions and can be calculated for various values of $\boldsymbol{\theta}$. Consequently, $(\hat{P}_N)$ becomes a deterministic optimization program and can be solved by standard methods of mathematical programming. Rubinstein and Shapiro (1993) showed that, under mild regularity conditions: (i) The optimal solution $\hat{\boldsymbol{\theta}}_N$ of the program $(\hat{P}_N)$ converges with probability one as $N \to \infty$ to its true counterpart $\boldsymbol{\theta}^*$, i.e., $\hat{\boldsymbol{\theta}}_N$ is a consistent estimator of $\boldsymbol{\theta}^*$; (ii) $N^{1/2}(\hat{\boldsymbol{\theta}}_N - \boldsymbol{\theta}^*)$ converges in distribution to multivariate normal with zero mean vector and covariance matrix $B^{-1}\sum B^{-1}$, where $B = \nabla^2 l(\boldsymbol{\theta}^*)$ and $\sum$ is the asymptotic covariance matrix of $N^{1/2}\nabla \hat{l}_N(\boldsymbol{\theta}^*)$, i.e., $\hat{\boldsymbol{\theta}}_N$ is asymptotically normal $N(\boldsymbol{\theta}^*, N^{-1}B^{-1}\sum B^{-1})$. Extensive simulation studies with the SF approach, as well as extension of the above simulation-based approach to constrained optimization, can be found in Rubinstein and Shapiro (1993).

See

- ▶ PERT
- ▶ Perturbation Analysis
- ▶ Sample Average Approximation
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Simulation Optimization
- ▶ Variance Reduction Techniques in Monte Carlo Methods

References

Asmussen, S., & Glynn, P. W. (2007). *Stochastic simulation*. New York: Springer.

Glasserman, P. (1991). *Gradient estimation via perturbation analysis*. Boston: Kluwer Academic.

Glynn, P. W. (1990). Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM, 33*, 75–84.

Ho, Y. C., & Cao, X. R. (1991). *Perturbation analysis of discrete event dynamic systems*. Boston: Kluwer Academic Publishers.

L'Ecuyer, P. L. (1990). A unified version of the IPA, SF, and LR gradient estimation techniques. *Management Science, 36*, 1364–1383.

Reiman, M. I., & Weiss, A. (1989). Sensitivity analysis for simulations via likelihood rations. *Operations Research, 37*, 830–844.

Rubinstein, R. Y. (1976). *A Monte Carlo method for estimating the gradient in a stochastic network, technical report*. Haifa, Israel: Technion.

Rubinstein, R. Y., & Shapiro, A. (1993). *Discrete event systems: Sensitivity analysis and stochastic optimization via the score function method*. Chichester, UK: Wiley.

Uryasev, S. (1995). Derivatives of probability functions and some applications. *Annals of Operations Research, 56*, 287–311.

Uryasev, S. (1997). Analytic perturbation analysis for DEDS with discontinuous sample-path functions. *Stochastic Models, 13*, 457–490.

Scoring Model

- ▶ Research and Development

Scripted Battle Model

A model or game in which some (or all) major events are predetermined to ensure that particular points are addressed. The list of predetermined events (with time and description) is the script.

See

- ▶ Battle Modeling

Search Theory

Lawrence D. Stone
Metron Inc., Reston, VA, USA

Introduction

Search theory is the study of how to effectively employ limited resources when trying to find an object whose location is not precisely known. The goal is to deploy search assets to maximize the probability of locating the search object with the resources available. Sometimes this goal is stated in terms of minimizing

the time to find the search object. Search theorists seek to find methods, procedures, and algorithms that describe how to achieve these goals. In search theory, the search object is called the target even when the target is neutral or friendly, such as a person lost at sea.

In 1942 work on search theory began in the U. S. Navy's Antisubmarine Warfare Operations Research Group (ASWORG) (1942) in response to the German submarine threat in the Atlantic (Morse 1982). A summary of the work done by this group from 1942 to 1945 is given in Sternhell and Thorndike (1946).

Bernard Koopman joined ASWORG in 1943, and at George Kimball's suggestion, Koopman, James Dobbie, and a few others were given the job of assembling the existing results on search into a coherent theory. Morse (1982) credits Koopman with providing the basic probabilistic foundation of the subject and finding the first results on optimal allocation of search effort, specifically the optimal allocation of a fixed amount of search effort to detect a stationary target with a bivariate normal distribution of possible locations and an exponential detection function.

Koopman defined the elements of the basic problem of optimal search: a prior probability distribution on target location; a function relating search effort and detection probability; a constrained amount of search effort; and an optimization criterion of maximizing probability of detection subject to the constraint on effort. This is called the optimal detection search problem: finding an optimal allocation of a fixed amount of search effort to maximize probability of detection.

The resulting synthesis of search theory by Koopman and his colleagues was published in *Search and Screening* (Koopman 1946), which defines many of the basic search concepts such as lateral range function, sweep width, sweep rate, detection function, and kinematic enhancement.

Search and Screening provided methods for designing barrier searches (bow tie searches) and antisubmarine warfare screens. It presented models for radar and visual search. This report and its updated version, Koopman (1980), are still the classic references on basic search theory. Washburn (1981b) provides an excellent and very readable introduction to search and detection problems.

Types of Search Problems

The work of Koopman and his colleagues in the ASWORG (later the Operations Evaluation Group (OEG)) laid the groundwork for the development of search theory and the applications that followed. It is convenient to categorize this subsequent work according to the type of search problem involved. A detailed bibliography and discussion of the types of search problems can be found in Benkoski, Monticino, and Weisinger (1991).

One-Sided Search Problems

The simplest type of search problems are those in which the searcher can choose his strategy, but the target neither chooses a strategy nor reacts to the search in any way. These are called one-sided search problems. The simplest one-sided problems involve search for a stationary target.

Stationary Targets. A stationary target is one that does not move. The searches for the sunken treasure ship, *SS Central America* (Stone 1992), the missing submarine *USS Scorpion* (Richardson and Stone 1971), and the H-bomb lost off the coast of Spain in 1964 (Richardson 1967) are examples of searches for stationary targets. Other examples include searches for downed aircraft, hidden natural resources (gas, oil, minerals, etc.), searches for archeological sites and artifacts, and even searches for something as mundane as lost car keys.

These are one-sided search problems because the target has not chosen its location and it does not react to the searcher's efforts.

Moving Targets. Search for a life raft adrift in the ocean is an example of a one-sided moving target search problem. The movement of the raft is not (substantially) under the control of the people in the raft, and the people are not able to react to the search effort except perhaps by trying to signal an aircraft or a passing vessel. The U. S. Coast Guard's Search and Rescue Optimal Planning System (SAROPS) employs search theory to plan searches for people and vessels lost at sea (Kratzke et al. 2010). Searches for submarines can be considered one-sided searches when the searching platform or system is covert, i.e., when the target submarine is unaware of the searcher's presence. During the Cold War, the U. S. Navy used a computer system that employed search theory to plan passive sonobuoy searches by Anti-Submarine

Warfare Patrol Aircraft for Soviet Nuclear submarines. Use of the system doubled the success probability of the searches.

Two-Sided Search Problems

In two-sided search problems, both the target and the searcher are allowed to choose their strategies. Two-sided problems can involve either stationary or moving targets. An example of a two-sided stationary target problem occurs when the target chooses a place to hide and stays there. The searcher then has to find the target. Most two-sided problems involve moving targets. Two-sided search problems divide into cooperative and non-cooperative searches.

Cooperative. An example of a two-sided cooperative search is a rendezvous search. In these searches two people are trying to find one another. For example, when two people have become separated and wish to find one another again, one has a cooperative search problem.

Another example is searching for an intelligent person lost in the woods. That person may be trying to move to a place where he can be found more easily or to cooperate in some way by leaving or giving signals to indicate his position.

Non Cooperative. Many two-sided searches are non-cooperative. An example is one submarine searching for another submarine when each is aware of the other's presence. Another example is law enforcement officers searching for drug smugglers.

Optimal Search for Stationary Targets

Koopman (1956a, b, 1957) published three articles that summarized, in an unclassified fashion, the theoretical aspects of the work reported in *Search and Screening* (Koopman 1946),which was classified at that time. In these papers Koopman showed how to find optimal allocations of search effort when the target is stationary and the detection function is exponential. He was able to solve explicitly for the optimal effort allocation for a bivariate normal target location distribution.

Lateral Range Function and Sweep Width. Koopman characterized a sensor's detection capability by the use of a lateral range function defined as follows. Consider a sensor that passes by a stationary target on long a straight path. The range r of the target from the sensor at the point of closest approach is called the lateral range of the target for that path. Let

$$\alpha(r) = \text{probability the sensor detects the target on a path having lateral range } r \text{ to target.} \tag{1}$$

Positive lateral ranges indicate ranges on right hand side of the sensor. Negative ones indicate ranges on the left-hand side.

Sweep Width. The sweep width W of a sensor is defined as

$$W = \int_{-\infty}^{\infty} \alpha(r)dr. \tag{2}$$

If one imagines the sensor moving through space on straight line of length l, then $l\,W$ is the effective (expected) area swept by the sensor.

Exponential Detection Function. Suppose that one is searching with a sensor that has sweep width W and moves at speed v. If the search is uniform in a region of area A with the effectiveness of "random" search, then the probability of detecting the target by time t given it is located in the region is

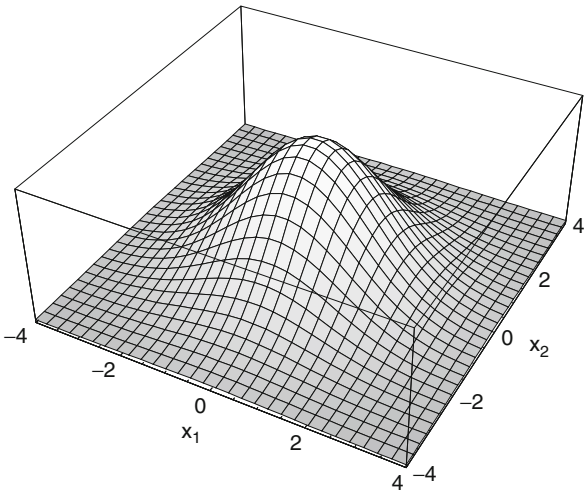
$$P(t) = 1 - \exp\left(-\frac{Wvt}{A}\right). \tag{3}$$

The fraction Wvt/A is the density of search effort in the region. Equation (3) is called the random search formula. This typically gives a lower bound on the effectiveness of a systematic search that tries to spread its effort uniformly over the search region. The term random searchmust not be taken too literally. With completely random searching, one can obtain very non-uniform coverage of the search area, and as a result obtain a lower probability of detection than that given by the random search formula.

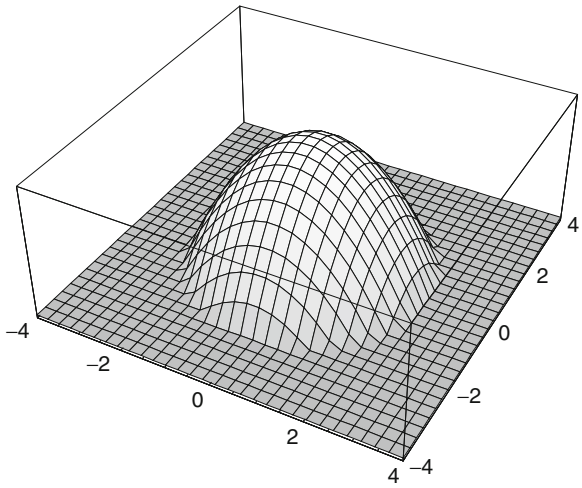
Suppose that $f(x)$ is the density of search effort in the neighborhood of the point x in the search space, X, the plane. Let

$$b(x,f(x)) = \text{probability of detecting the target given it is located at } x \text{ and the search density is } f(x).$$

The function b is called a detection function. If $b(x,f(x)) = 1 - \exp(-f(x))$, then b is an exponential detection function.



Search Theory, Fig. 1 Probability density function for a circular normal distribution



Search Theory, Fig. 2 Optimal search density for a circular normal distribution

Target Location Distribution. Suppose that the target is stationary and located in the plane X. Knowledge of the target's location is given by a bivariate normal probability distribution with its mean at $(0,0)$. This knowledge may have been obtained from a navigational fix with uncertainty where the uncertainty in the fix is modeled by a bivariate normal distribution.

The density function, p, for this distribution is given by

$$p(x_1,x_2)=\frac{1}{2\pi\sigma_1\sigma_2}\exp\left[-\frac{1}{2}\left(\frac{x_1^2}{\sigma_1^2}+\frac{x_2^2}{\sigma_2^2}\right)\right].$$

A graph of this density function is shown in Fig. 1 for the case where $\sigma_1=\sigma_2$. This is called a circular normal distribution. In Fig. 1, the probability density is highest at the center of the distribution (0,0) and decreases as distance from the center increases. In theory, this distribution covers an infinitely large area since the probability density approaches, but never actually reaches, zero. In practice some reasonable cutoff is applied.

Optimal Search Density. Suppose that the search sensor has sweep width W, travels at speed v, and has an exponential detection function. If there are T hours of search time available, then Koopman (1946) showed that the optimal search density f^* is

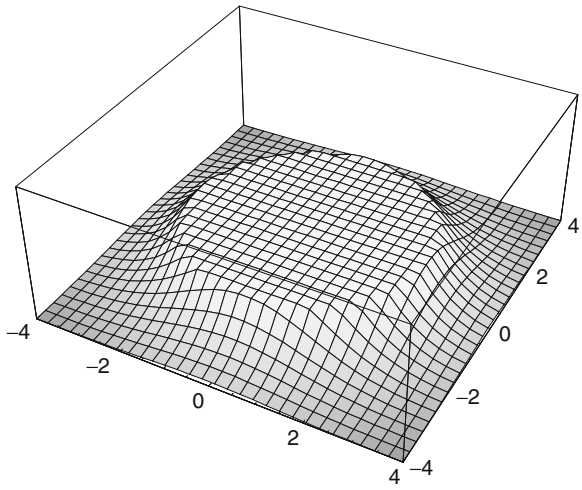
$$f^*(x_1,x_2)=\begin{cases}\left(\frac{WvT}{\pi\sigma_1\sigma_2}\right)^{\frac{1}{2}}-\frac{1}{2}r^2(x_1,x_2) & \text{for } r^2(x_1,x_2)\le 2\left(\frac{WvT}{\pi\sigma_1\sigma_2}\right)^{\frac{1}{2}}\\ 0 & \text{for } r^2(x_1,x_2)>2\left(\frac{WvT}{\pi\sigma_1\sigma_2}\right)^{\frac{1}{2}}\end{cases}$$

where

$$r^2(x_1,x_2)=\frac{x_1^2}{\sigma_1^2}+\frac{x_2^2}{\sigma_2^2}.$$

Figure 2 shows an example of the optimal search effort density for the circular normal location density in Fig. 1 based on a specific amount of search effort. The optimal effort density is highest at the center where the location probability density is also highest. The optimal density decreases with distance from the center until at a certain radius the effort density becomes zero. All the available search effort is expended within a certain radius (depending on the amount of effort available) and none is expended outside that radius even though there is some probability of the target being outside the circle inscribed by this radius.

Posterior Target Location Density. Suppose that the optimal search has been applied as shown in Fig. 2 and failed to detect the target. What is the target location distribution given this unsuccessful search effort? This distribution is computed by employing the form of probabilistic reasoning called



Search Theory, Fig. 3 Posterior Target Location Distribution given Failure to Detect

Bayes' rule. The result is shown in Fig. 3. This is the posterior target location density given the search has been unsuccessful. The posterior density is flat inside the circle where search effort has been applied. As more and more effort is applied (in an optimal fashion), the posterior density becomes flatter and flatter, and the radius of the circle of search increases.

Koopman (1957) extended his optimal allocation results from normal distributions to a more general class of probability distributions

Non-exponential Detection Functions

Koopman's original results have been extended in two important directions to allow one to find optimal allocations of search effort when the detection function is not exponential and when the target location distribution is not bivariate normal. Target location distributions that are not bivariate normal occur often in operational problems. For example, cellular target location distributions commonly arise in both land and maritime search and rescue situations. The use of detection functions that are not exponential is also common in operational search problems. For example the U. S. Coast Guard uses inverse-cube detection models in SAROPS (Kratzke et al. 2010). This detection function was initially postulated by Koopman (1946) as a model for visual search.

The exponential detection function has an important property. It is exhibits a decreasing rate of return. This means that the probability of detection increases more and more slowly as the amount of search effort increases. This effect is seen clearly in Fig. 4.

In mathematical terms, this property is expressed by saying the detection function has a decreasing derivative or rate of return. A decreasing rate of return is a common property in economic situations in which effort may be measured in dollars, time, or manpower and return is in dollars. Most detection functions have the decreasing rate of return property. DeGuenin (1961) extended Koopman's results by finding the optimal allocation of search effort for a stationary target for any detection function with a decreasing rate of return. Let $b'(x,z)$ denote the derivative of detection function $b(x,z)$ with respect to z. If $b'(x,z)$ is decreasing in z, then b has the decreasing rate of return property.

Regular Detection Functions. A detection function is regular if $b(x,0) = 0$ and $b'(x,\cdot)$ is continuous, positive, and strictly decreasing for $x \in X$. Let

$$\rho_x(z) = p(x)b'(x,z) \text{ for } x \in X \text{ and } z \geq 0. \quad (4)$$

If one has applied z effort density at x, then $\rho_x(z)$ is the marginal rate of return (in terms of increase in probability of detection) for applying additional search effort density at x when density z has already been applied.

Optimal Allocations for Continuous Target Location Distributions

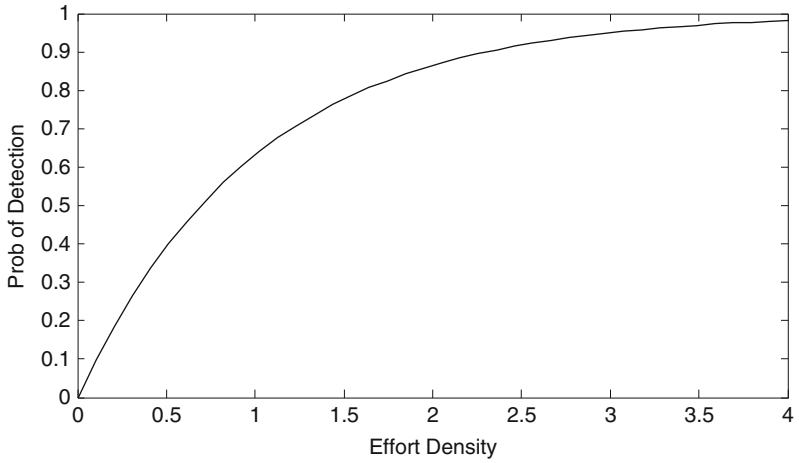
A search plan is a non-negative function f defined on the search space X which is a subset of Euclidean n-space. Typically, $n = 2$ or 3. The probability of detection, $P[f]$ and cost $C[f]$ of the plan f are computed as follows.

$$P[f] = \int_X b(x,f(x))dx \text{ and } C[f] = \int_X f(x)dx. \quad (5)$$

A search plan f^* is optimal for cost K if

$$P[f^*] \geq P[f] \text{ for all search plans } f \text{ such that } C[f] \leq K. \quad (6)$$

Stone (1989, section 2.2) showed that the following is a necessary and sufficient condition for a search plan f^* to be optimal for cost K.



Search Theory,
Fig. 4 Exponential detection function

Optimality Condition for Stationary Target Search. If the detection function b is regular, then a search plan f^* is optimal for cost $C[f^*]$ if and only if there exists $a\ \lambda > 0$ such that for $x \in X$

$$\begin{aligned}\rho_x(f^*(x)) = p(x)b'(x,f^*(x)) = \lambda \text{ if } f^*(x) > 0\\ \leq \lambda \text{ if } f^*(x) = 0.\end{aligned} \tag{7}$$

Observe that the optimal plan searches in a way to level the marginal rate of returns to some value λ in all locations where search effort is allocated and to be less than or equal to λ in all locations where no search is effort is allocated.

In the case where b is an exponential detection function, $\rho_x(f(x)) = p(x)\exp(-f(x))$. Let $\tilde{p}$ denote the density function of the posterior target location distribution given failure of plan f to detect the target. By Bayes' rule,

$$\begin{aligned}\tilde{p}(x) \propto \Pr\{\text{ plan } f \text{ fails to detect target}|\text{target at } x\}p(x) =\\ \exp(-f(x))p(x) = \rho_x(f(x)).\end{aligned}$$

Thus for an exponential detection function, the optimal plan searches the highest probability areas in a way that levels the posterior distribution where search takes place as one can see from Fig. 3.

Computing Optimal Stationary Target Search Plans. The optimality condition provides a method of computing the optimal search plan for a given cost K. The assumption that b is a regular detection function insures that ρ_x has an inverse for any x such that $p(x) > 0$. For $\lambda > 0$ define

$$\begin{aligned}f_\lambda(x) = \begin{cases}\rho_x^{-1}(\lambda) & \text{if } \lambda \leq \rho_x(0)\\ 0 & \text{otherwise}\end{cases} \text{ and}\\ U\ (\lambda) = \int_X f_\lambda(x)dx \text{ for } \lambda > 0.\end{aligned} \tag{8}$$

One can check that U is a function that decreases continuously from ∞ to 0 as λ increases from 0 to ∞. This means that for any $K \geq 0$, one can find a $\lambda^* > 0$ such that $C[f_{\lambda^*}] = K$. By (8), f_{λ^*} satisfies

$$\begin{aligned}\rho_x(f_{\lambda^*}(x)) = \lambda \text{ if } f_{\lambda^*}(x) > 0\\ \leq \lambda \text{ if } f_{\lambda^*}(x) = 0.\end{aligned} \tag{9}$$

By the optimality condition, f_{λ^*} is optimal for cost K.

Optimal Allocations for Cellular Distributions

In the above examples, the probability distributions have density functions that vary smoothly over space. However, there are many situations in which the search space is divided into cells. This occurs in Coast Guard maritime search and rescue problems, land search and rescue problems, and in many others.

Suppose there are J cells with

$$\begin{aligned}p_j &= \text{probability the target is in cell } j\\ A_j &= \text{area of cell } j\\ W_j &= \text{sweep width in cell } j\\ v_j &= \text{search speed in cell } j\\ t_j &= \text{time spent searching in cell } j.\end{aligned} \tag{10}$$

The optimal search problem is to divide the total search time T over the cells to maximize probability of detection.

Charnes and Cooper Algorithm. Suppose the detection function in the jth cell is

$$b(j,t) = 1 - \exp\left(-\frac{W_j v_j t}{A_j}\right). \tag{11}$$

For this case, Charnes and Cooper (1958) presented the following algorithm for computing the optimal distribution of search effort over these cells to maximize probability of detection by search time T.

The problem solved by Charnes and Cooper arises often in land search and rescue. The probability distribution for the location of the target, say a lost boy, is often cellular with cells of varying size. Because of variations in terrain, both the sweep width and the speed at which one can search may vary from cell to cell.

Optimal Allocation f^* of T Search Time for a Cellular Distribution

Set $\alpha_j = \dfrac{W_j v_j}{A_j}$ for $j = 1, \ldots, J$

Index cells so that $\alpha_1 p_1 \geq \alpha_2 p_2 \geq \ldots \geq \alpha_J p_J$

Set $S(0) = 0$ and $y_J = 1$

For $i = 1$ to $J - 1$

$$y_i = \ln \frac{\alpha_i p_i}{\alpha_{i+1} p_{i+1}} \text{ and } S(i) = S(i-1) + y_i \sum_{j=1}^{i} \frac{1}{\alpha_j}$$

End

If $S(j^*) \geq T$ for some $j^* < J$,

then

$$a = \frac{T - S(j^* - 1)}{S(j^*) - S(j^* - 1)}$$

otherwise

$$j^* = J \text{ and } a = \frac{T - S(j^* - 1)}{\sum_{j=1}^{J} \frac{1}{\alpha_j}}$$

End

Set

$$f^*(j) = \begin{cases} \dfrac{1}{\alpha_j}\left(\sum_{i=j}^{j^*-1} y_i + a y_{j^*}\right) & \text{for } 1 \leq j \leq j^* \\ 0 & \text{for } j > j^* \end{cases}$$

Uniformly Optimal Search Plans

The search plans described above maximize the probability of detecting the target with a fixed amount of effort K or time T. They tell the searcher the total effort to put into each cell or region, but they say nothing about how the effort should be put into cells over time. Suppose that one wants a plan that tells the searcher how to allocate search effort in space and time so that at each time t between 0 and T, he has done as well as possible. In fact, one would like the result after time t of search to be *optimal* for time t. A plan with this pleasing property is called uniformly optimal. Uniformly optimal plans also minimize the mean time to detect the target.

Koopman showed that when the detection function is exponential, a uniformly optimal plan exists. In fact, the way to obtain a uniformly optimal plan is to organize the search effort so that by time t one has allocated search effort to be optimal for that time. One then continues on to a plan that is optimal for time $T > t$ by distributing the additional effort that it is required by the time T plan over the time t plan. Stone (1989) showed that uniformly optimal plans exist and may be constructed in a similar fashion whenever the detection function is regular.

Uniformly Optimal Plans for a Continuous Search Space. The family of optimal plans defined in (8) provides a method for finding uniformly optimal plans. Let $M(t)$ be the cumulative amount of search effort available by time t for $0 \leq t \leq T$. Consider the class of search plans φ which are defined over space and time so that

$$\varphi(x,t) \geq 0,\ \varphi(x,\cdot) \text{ is increasing for } x \in X \text{ and } C[\varphi(\cdot,t)] = M(t) \text{ for } 0 \leq t \leq T.$$

Let U^{-1} be the inverse of the function U defined in (8). Set

$$\lambda(t) = U^{-1}(M(t)) \text{ and } \varphi^*(x,t) = f_{\lambda(t)}(x) \text{ for } x \in X \text{ and } 0 \leq t \leq T. \tag{12}$$

Then $\varphi^*(x,t)$ specifies the cumulative effort density to be applied to point x by time t. One can check that λ is a decreasing function and that as a result $\varphi^*(x,\cdot)$ is an increasing function of t for $x \in X$. Furthermore, $C[\varphi^*(\cdot,t)] = M(t)$, and $\varphi^*(\cdot,t)$ satisfies the optimality conditions in (7). Thus, $\varphi^*(\cdot,t)$ is optimal for cost $M(t)$ for $0 \le t \le T$, and φ^* is a uniformly optimal search plan.

Incrementally Optimal Plans. Usually one allocates search effort in discrete increments rather than continuously. So one might first design and implement a plan to be optimal for $M(t_1)$, the amount of effort available by t_1. If this search fails, one might decide to add an additional increment Δ of effort by time t_2 to obtain a total effort of $M(t_2) = M(t_1) + \Delta$. *Question:* How does one allocate this increment Δ optimally and will this allocation produce a plan that is optimal for the total effort $M(t_2) = M(t_1) + \Delta$? The uniformly optimal plan in (12) provides the answer to both of these questions. First, the allocation of the initial increment is specified by $\varphi^*(\cdot,t_1)$. Then the allocation of the increment Δ is given by $\varphi^*(\cdot,t_2) - \varphi^*(\cdot,t_1)$ and the resulting total effort allocation $\varphi^*(\cdot,t_2)$ is optimal for $M(t_2) = M(t_1) + \Delta$.

Minimizing Mean Time to Detection. A common question is, can you design a search plan to minimize the mean time to detect the target? The answer is yes, and the plan that does this is the uniformly optimal plan. First one has to imaging extending to T to ∞ so that the detection probability will reach 1. The mean time to detect, $\mu[\varphi]$, for a plan φ can be calculated by

$$\mu[\varphi] = \int_0^\infty (1 - P[\varphi(\cdot,t)])\,dt. \tag{13}$$

Since φ^* is uniformly optimal, it minimizes the integrand in (13) for all t and therefore produces the minimum mean time to detection. If $T < \infty$ and the probability of detection for the optimal plan is less than 1 by time T, then one can define the mean time to complete the search as follows

$$\mu[\varphi] = \int_0^T (1 - P[\varphi(\cdot,t)])\,dt + (1 - P[\varphi(\cdot,T)])T$$

and see that φ^* minimizes this as well as maximizing the probability of detection by time T.

Uniformly Optimal Plans for Cellular Distributions. Stone (1989, Section 2.2) finds uniformly optimal plans for cellular distributions and regular detection functions.

Suppose there are J cells with probability p_j of the target being cell j and that the detection function b is regular. Let $M(t)$ be the cumulative search effort available by time t. Define

$$\rho_j(z) = p_j b'(j,z) \text{ for } j = 1,\ldots,J \text{ and } z \ge 0$$

$$f_\lambda(j) = \begin{cases} \rho_j^{-1}(\lambda) \text{ if } \lambda \le \rho_x(0) \\ 0 \quad \text{otherwise} \end{cases} \text{ and } U(\lambda) = \sum_{j=1}^J f_\lambda(j)$$

Let U^{-1} be the inverse of the function U, and set $\lambda(t) = U^{-1}(M(t))$. Then

$$f_{\lambda(t)}(j) = \rho_j^{-1}(\lambda(t)) \text{ for } j = 1,\ldots,J \tag{14}$$

is optimal for $M(t)$ effort for $t \ge 0$ and is uniformly optimal.

Optimal Search with Uncertain Sweep Width

In the detection problems discussed so far, such as the cellular search problems, it is assumed that the sweep width of the sensor is known with certainty. It is often the case that the sweep width is uncertain, particularly in cases where state of the target is uncertain. For example, if one is looking for a lost aircraft, one often does not know if the plane has landed in one piece of if there is wreckage scatter over an area. Maybe the planed burned after crashing. Each of these situations will yield a different value for the sweep width of a visual search from the air.

Richardson and Belkin (1972) and Stone (1989, section 2.3) show how to find optimal plans when the sweep width has a probability distribution on its value.

Search in Presence of False Targets

Often searches are performed with a sensor such as a side-looking sonar which is used in underwater searches for lost objects. This type of sensor is excellent for covering large areas with high detection probability. However, these broad search sensors can generate a large number of false targets, i.e., detections on real objects which are not the target. Thus detections have to be investigated, perhaps by a visual sensor, to determine whether the objects detected are targets or not. Thus, a search plan in the presences of false targets must specify not only the

allocation of the broad search effort but a strategy for investigating detections. Chapter 6 of Stone (1989) develops models of search in the presence of false targets, introduces a number of classes of search plans in presence of false targets, and finds plans that minimize expected cost to detect the target.

Optimal Search for Moving Targets

Prior to Brown (1980), optimal allocation results for moving targets were limited to very special cases. Most moving target problems were approached by freezing the target motion over some increment of time, allocating effort as though the target were stationary during that time increment, and then repeating the process for the next time increment. For example, the U. S. Coast Guard's Computer Assisted Search Planning (CASP) System (Richardson and Discenza 1980) used this technique. Brown discovered an efficient algorithm for finding optimal search allocations for moving target problems when the target distribution is cellular, the target motion is Markovian, and the detection function is exponential. Brown's algorithm maximizes detection probability at time T. This algorithm was applied to great effect by the U. S. Navy in searching for Soviet Submarines.

Washburn (1983) generalized Brown's algorithm to the class of forward and backward (FAB) algorithms that apply to a more general class of payoff functions. Algorithms for non-exponential detection functions and non-Markovian motions are given in Stone (1979) and Stromquist and Stone (1981).

Optimal One-Sided Search for a Moving Target

The target's location and motion through X are specified by the stochastic process $X = \{X_t; t \geq 0\}$ where $X_t \in X$ gives the target's position at time t. A time horizon $[0, T]$ is specified and the goal is to maximize the probability of detecting the target by time T. For this discussion, time will be discrete so that $t = 0, 1, \ldots, T$.

A search plan ψ specifies the allocation of search effort in space and time. Specifically $\psi(x, t) =$ effort density placed at point x at time t for $x \in X,\ t = 0, 1, \ldots, T$. Search effort is constrained by the rate at which effort can be applied. Specifically there is a function m such that $m(t) =$ effort available for search at time t for $t = 0, 1, \ldots, T$, and search plans ψ must satisfy

$$\int_Y \psi(x, t)\, dx \leq m(t) \text{ for } t = 0, 1, \ldots, T, \tag{15}$$

$$\psi(x, t) \geq 0 \text{ for } x \in X,\ t = 0, 1, \ldots, T. \tag{16}$$

Let Ψ be the set of search plans satisfying (15) and (16). For each sample path ω of the process X, the probability of detecting the target by time t, given that it follows that path, is a function of the weighted total effort density,

$$\zeta(\psi, \omega, t) = \sum_{s=0}^{t} W(X_s(\omega), s)\psi(X_s(\omega), s),$$

which accumulates by time t on the target over the course of the path. The weight $W(x, s)$ represents the relative detectability or sweep width against the target given it is located at point x at time s. There is a detection function $b : [0, \infty] \to [0, 1]$ such that $b(\zeta(\psi, \omega, t))$ is the probability of detecting the target by time t given that it follows sample path ω and that search plan ψ is executed. Letting E denote expectation over the sample paths of X, it follows that

$$P[\psi] = E[b(\zeta(\psi, \cdot, T))]$$

is the probability of detecting the target by time T with plan ψ. In the remainder of this discussion, the argument ω is suppressed.

The optimal detection problem for a moving target is to find a plan $\psi^* \in \Psi$ such that $P[\psi^*] \geq P[\psi]$ for all $\psi^* \in \Psi$. Such a plan is called *T-optimal*.

Brown's Algorithm for Continuous Space. For an exponential detection function and a target moving in discrete time and space, Brown's algorithm solves the problem of finding a T-optimal allocation by solving a sequence of stationary target problems. The following paragraphs present an extension of Brown's algorithm to continuous search spaces and relate this extension to the original discrete-space algorithm. The continuous space algorithm is based on the following necessary and sufficient condition for a T-optimal search plan proved by Stone (1979).

Define

$$g_t^{\psi}(x) = \Pr\{X_t = x | \text{failure to detect at all times other than } t \text{ using plan } \psi\}$$

$$\text{for } x \in X,\ \psi \in \Psi,\ \text{and } t = 0, 1, \ldots, T.$$

The function g_t^{ψ} is the posterior target location density given failure to detect by the search effort at all times other than t. If the detection function b is exponential, i.e., $b(z) = 1 - \exp(-z)$ for $z \geq 0$, then $g_t^{\psi}(x)$ is proportional to

$$\mathrm{E}\left[\exp\left\{-\sum_{s \neq t} W(\mathrm{X}_s, s)\,\psi(\mathrm{X}_s, s)\right\} \middle| \mathrm{X}_t = x\right]$$

where p_t is the prior probability density function for X_t

Necessary and Sufficient Condition for T-Optimality. Assume that the detection function b is exponential. A necessary and sufficient condition for $\psi^* \in \Psi$ to be T-optimal is that $\psi^*(\cdot, t)$ maximizes the probability of detecting a stationary target with distribution $g_t^{\psi^*}$ using effort $m(t)$ for $t = 0, 1, \ldots, T$.

Description of Algorithm. For time $t = 0$, the algorithm allocates $m(0)$ effort optimally to the target distribution p_0. For $t = 1, \ldots, T$, the algorithm calculates the posterior target distribution at time t given failure to detect by the effort prior to time t and allocates $m(t)$ effort in a manner that is optimal for the stationary target problem with target distribution given by g_t^{ψ}. The plan that results from this first pass is the incrementally optimal or myopic search plan. It maximizes the increment in detection probability at each time t but does not produce a T-optimal plan.

Subsequent passes proceed as follows for $t = 0, 1, \ldots, T$. The algorithm computes g_t^{ψ} based on the allocation ψ obtained up to that point in the iteration, redistributes the effort $m(t)$ at time t to be optimal for g_t^{ψ}, and changes ψ to reflect the reallocation. The algorithm continues in this iterative fashion until a convergence criterion is met.

To use this algorithm, one must be able to calculate g_t^{ψ} and find the optimal allocation of effort for a stationary target problem when the detection function is exponential. The methods described in the section on *Optimal Search for Stationary Targets* provide efficient algorithms for doing this. If one can find an efficient method of computing g_t^{ψ}, then one can implement the above algorithm for the case of a discrete or continuous search space. For the case where X is a discrete time and space Markov process, Brown (1980) devised a very efficient algorithm for computing g_t^{ψ} and used it along with the Charnes-Cooper algorithm described above to produce an algorithm for calculating T-optimal search plans. By making use of the upper bound discovered by Washburn (1981a), one can tell when his solution has come within a specified tolerance of the detection probability of the T-optimal plan. When using Brown's algorithm, the convergence is usually very rapid.

Stone et al. (1978) present a generalization of Brown's algorithm to arbitrary discrete time and space target motions.

Generalized Search Optimization

Stromquist and Stone (1981) found a set of necessary and sufficient conditions for maximizing a class of functionals with search theory applications. Using these conditions, they were able to unify the solutions to a number of previously solved search problems including the following.

Multistate Target Search. A generalization of the detection search described above is the multistate target search. In this case the target's motion is given by $\{(X_t, S_t); t = 0, 1, \ldots, T\}$ where X_t is the target's position at time t and S_t is the target's state at time t. The target may change state as well as location stochastically, and the target's state can affect the target's motion as well as its detectability. As an example, consider a case where there are K states and the sweep width is a function of location, state, and time, so that cumulative effort ζ becomes

$$\zeta(\psi, T) = \sum_{t=0}^{T} W(X_t, S_t, t)\psi(X_t, t)$$

and

$$P[\psi] = E[b(\zeta(\psi, T))]$$

as before. Observe that effort cannot be allocated to states but only to locations. Discenza and Stone (1981) developed algorithms for solving these multistate target search problems.

Optimal Survivor Search is a special case of multistate target search. In the search problems discussed above, the goal is to maximize the probability of detecting the target by some time. In the case of search and rescue problems, a more appropriate goal may be to maximize the probability of detecting the target alive. A search with this goal

may apply the initial effort in some lower probability areas that are particularly hazardous to the target in order to recover a survivor quickly if he or she is located there. This may involve some sacrifice of overall detection probability. As an example, one might want to concentrate initially on search areas where a survivor would be located if he is immersed in the water and delay somewhat searching areas that would be likely only if he is still in his boat.

Defensive Search is another special case of multistate target search. In defensive search, one is trying to detect an attacker before it launches a weapon. In this case the target has two states, weapon launched and not launched. Once the attacker launches a weapon, the sweep width is set to zero and the target remains in the launched state for the remainder of the problem. In this case, maximizing P is maximizing the probability of detecting the attacker before it launches an attack

Surveillance Search. Tierney and Kadane (1983) have developed a technique for solving surveillance problems which builds on the optimal detection search results discussed above. The surveillance problem is to maximize the probability of being in contact (i.e., having a detection on the target) at time T. In contrast to the detection search problem, a detection before time T does not end the problem. It merely helps to obtain a detection at time T. For problems where the target's motion is modeled by a discrete-time-and-space Markov chain, Tierney and Kadane have shown that the optimal surveillance problem can be solved by solving a series of optimal detection search problems. In their method, one starts at time T and works his way backward in time in a fashion similar to dynamic programming. At each time t, one must solve what Tierney and Kadane call a general detection search problem given knowledge of the target's position at time t.

In the general detection problem, the searcher receives a payoff or return $r(j,t)$ if he detects the target in cell j at time t. The search stops the first time the target is detected, and the objective of the general detection problem is to maximize the expected payoff.

Whereabouts Search. Stone and Kadane (1981) solve the problem of optimal whereabouts search for a moving target. In a whereabouts search one can succeed either by detecting the target or guessing its location. When $X = \{1, \ldots, J\}$, Stone and Kadane show that solving a whereabouts problem is equivalent to solving J detection search problems

Constraints on the Searcher

In the search problems considered above, it is assumed that effort can be distributed over the search space any way one chooses. Sometimes this is a reasonable approximation. A visual search by aircraft over a region where the time to travel from one part of the region to another is small is an example. Sometimes the constraints on the movement of the search platforms require that one consider special types of search plans. Usually there are two of types of constraints that are considered – path constraints and simplicity constraints.

Path Constraints. If the search platform is a boat or a person walking on land, then the place where the platform is searching now constrains the places where it can search in the next increment of time. In these cases, one has an optimal searcher path problem. Instead of finding an optimal allocation of search effort, the problem is to find an optimal path for the searcher. The set of paths from which the optimum is chosen is restricted to those that obey the physical constraints on the movement of the search platform. This is a difficult class of problems, especially for moving targets, but there has been some progress in solving them.

Stewart (1979, 1980), Eagle (1984), and Eagle and Yee (1988) have applied integer programming approaches to finding efficient algorithms for solving these problems.

Simplicity Constraints. In executing actual searches, it may be desirable to restrict the search patterns to a class of searches the are simple to execute operationally. A typical example is to restrict search plans to be composed of searches consisting of a set of rectangles each with a uniform search density or coverage. Such plans can be approximated by searches that employ equally spaced, parallel search paths in the rectangles.

Single Rectangle Searches. In the case of search for a stationary target with a bivariate normal location distribution, Richardson and Discenza (1980), show how to find optimal rectangle plans. For an optimal rectangle plan, one chooses a single rectangle and spreads his search effort uniformly over that rectangle. Richardson and Discenza show that it is always possible to pick an optimal rectangle plan that

comes quite close to (within 3% of) the detection probability of the optimal plan.

Multiple Rectangle Searches. Discenza (1980) developed an algorithm for finding optimal multiple rectangle searches for the cellular target location distributions generated by CASP. These multiple rectangle searches consist of non-overlapping rectangles. In each rectangle the search effort is spread uniformly over the rectangle. Furthermore, each search asset (say an aircraft) is assigned to search one and only one rectangle. The solution method proposed by Discenza involves some additional restrictions on the choices of rectangles to allow an efficient solution of this problem.

Kratzke, Stone, and Frost (2010) describe the methodology used in the SAROPS program to find optimal non-overlapping rectangles for Coast Guard Search and rescue problems. The optimization approach is numerical, but it does account for the movement of the search object during the search.

Search and Evasion Problems

A classic two-sided search problem involves a target that is trying to evade a searcher. In one case the target's goal may be simply to avoid detection. In other cases, the target may have additional goals such as reaching a certain area undetected. This would be the goal for a smuggler or an infiltrator. The goal for the search theorist is to solve for the optimal strategy for both the searcher and the evader. These problems tend to have a game theory formulation. Although they are difficult to solve, there has been some progress made by Auger (1991), Dobbie (1975), Eagle and Washburn (1991), Gal (1980), Garnaev (2000), Stewart (1981), and Washburn (1980b).

Alpern and Gal (2003, Book I) present an excellent and highly readable summary of the main results in search games. Below is a sampling of the results presented by them.

Search Games in a Bounded Connected Region Q of Euclidean Space of dimension 2 or more with an Immobile hider.

Search in a Region. Let A be the area or volume of the region Q and R the search rate (area or volume per unit time) of the searcher.

Result. The (minimax) value of this game is $A/2R$. This is true whether the searcher's path is continuous or not, whether he chooses his initial search point or not, and whether or not his strategy is randomized.

Search on a Network: A network is defined to be a finite connected set of arcs that intersect only at their end points. Let the sum of the length of all the arcs be equal to L. The searcher moves through the network at unit speed.

Result: The value of the search game using pure strategies for an immobile hider on the network Q is L if Q is Eulerian. Note: A network Q is Eulerian if there is a tour (a path traversing all the arcs of Q and returning to its starting point) of length L. If mixed strategies are allowed the value of the game drops to $L/2$.

Mobile Hider in Bounded Connected Region:

Result*:* In the case where the searcher's path does not have to be continuous, the value of the game is A/R.

Mobile Hider in a Bounded Convex Region Q in Euclidean 2-Space. For these searches r is the detection radius (which is small compared to the diameter of Q), the searcher moves in continuous paths at a speed of 1, and the mobile hider moves on a continuous path with a speed that is "not too small." The searcher starts at specified point in Q, the hider chooses his starting point.

Result. Both the searcher and the hider can keep the probability of capture before time t close to $1 - \exp(-2rt/A)$ where A is the area of Q.

Note that $1 - \exp(-2rt/A)$ is probability of detection by time t for random search (as defined by Koopman) for a stationary target in a region of area A.

See

- Bayes Rule
- Game Theory
- Markov Chains

References

Alpern, S., & Gal, S. (2003). *The theory of search games and rendezvous*. Boston: Kluwer.

ASWORG. (1942). Preliminary report on the submarine search problem by the ASWORG, 1 May.

Auger, J. M. (1991). An infiltration game on k arcs. *Naval Research Logistics, 38*, 511–529.

Benkoski, S. J., Monticino, M. G., & Weisinger, J. R. (1991). A survey of the search theory literature. *Naval Research Logistics, 38*, 469–494.

Brown, S. S. (1980). Optimal search for a moving target in discrete time and space. *Operations Research, 28*, 1275–1289.

Charnes, A., & Cooper, W. W. (1958). The theory of search: Optimum distribution of search effort. *Management Science, 5*, 44–50.

DeGuenin, J. (1961). Optimum distribution of effort: An extension of the Koopman basic theory. *Operations Research, 9*, 1–7.

Discenza, J. H. (1980). A solution for the optimal multiple rectangle problem. In K. B. Haley & L. D. Stone (Eds.), *Search theory and applications*. New York: Plenum Press.

Discenza, J. H., & Stone, L. D. (1981). Optimal survivor search with multiple states. *Operations Research, 29*, 309–323.

Dobbie, J. M. (1975). Search for an avoiding target. *SIAM Journal of Applied Mathematics, 28*, 72–86.

Eagle, J. N. (1984). Optimal search for a moving target when the search path is constrained. *Operations Research, 32*, 1107–1115.

Eagle, J. N., & Washburn, A. R. (1991). Cumulative search-evasion games. *Naval Research Logistics, 38*, 495–510.

Eagle, J. N., & Yee, J. R. (1990). An optimal branch and bound procedure for the constrained path, moving target search problem. *Operations Research, 38*, 110–114.

Gal, S. (1980). *Search games*. New York: Academic.

Garnaev, A. Y. (2000). *Search games and other applications of game theory*. Berlin: Springer.

Koopman, B. O. (1946). *Search and screening* (Operations Evaluations Group Report No. 56). Alexandria, VA: Center for Naval Analyses.

Koopman, B. O. (1956a). The theory of search, part I: Kinematic bases. *Operations Research, 4*, 324–346.

Koopman, B. O. (1956b). The theory of search, part II: Target detection. *Operations Research, 4*, 503–531.

Koopman, B. O. (1957). The theory of search, part III: The optimum distribution of searching effort. *Operations Research, 5*, 613–626.

Koopman, B. O. (1980). *Search and screening: General principles with historical applications*. New York: Pergamon Press.

Kratzke, T. M., Stone, L. D., & Frost, J. R. (2010). Search and rescue optimal planning system. *Proceedings of the 13th International Conference on Information Fusion*, Edinburgh, Scotland, 26–29 July 2010.

Morse, P. M. (1982). In memoriam: Bernard Osgood Koopman, 1900–1981. *Operations Research, 30*, 417–427.

Richardson, H. R. (1967). Operations analysis. Chapter V of Part 2 in *Aircraft Salvage Operation*, *Mediterranean*; Report to Chief of Naval Operations from the Supervisor of Salvage and the Deep Submergence Systems Project, US Navy.

Richardson, H. R., & Belkin, B. (1972). Optimal search with uncertain sweep width. *Operations Research, 20*, 764–784.

Richardson, H. R., & Discenza, J. H. (1980). The United States Coast guard computer-assisted search planning system (CASP). *Naval Research Logistics, 27*, 659–680.

Richardson, H. R., & Stone, L. D. (1971). Operations analysis during the underwater search for scorpion. *Naval Research Logistics, 18*, 141–157.

Sternhell, C. M., & Thorndike, A. M. (1946). *Antisubmarine warfare in world war II*. Operations (Evaluations Group Report No. 51). Alexandria, VA: Center for Naval Analyses.

Stewart, T. J. (1979). Search for a moving target when searcher motion is restricted. *Computers and Operations Research, 6*, 129–140.

Stewart, T. J. (1980). Experience with a branch and bound algorithm for constrained searcher motion. In K. B. Haley & L. D. Stone (Eds.), *Search theory and applications*. New York: Plenum Press.

Stewart, T. J. (1981). A Two-cell model of search for an evading target. *European Journal of Operations Research, 8*, 369–378.

Stone, L. D. (1979). Necessary and sufficient conditions for optimal search plans for moving targets. *Mathematics of Operations Research, 4*, 431–440.

Stone, L. D. (1989). *Theory of optimal search* (2nd ed.). Linthicom, MD: INFORMS.

Stone, L. D. (1992). Search for the *SS central America*: Mathematical treasure hunting. *Interfaces, 22*, 32–54.

Stone, L. D., Brown, S. S., Buemi, R. P., & Hopkins, C. R. (1978). *Numerical optimization of search for a moving target*. Daniel H. Wagner, Associates Report to Office of Naval Research.

Stone, L. D., & Kadane, J. B. (1981). Optimal whereabouts search for a moving target. *Operations Research, 29*, 1154–1166.

Stromquist, W. R., & Stone, L. D. (1981). Constrained optimization of functionals with search theory applications. *Mathematics of Operations Research, 6*, 518–529.

Tierney, L., & Kadane, J. B. (1983). Surveillance search for a moving target. *Operations Res., 31*, 720–738.

Washburn, A. R. (1980a). On search for a moving target. *Naval Res. Logist. Quart., 27*, 315–322.

Washburn, A. R. (1980b). Search-evasion game in a fixed region. *Operations Research, 28*, 1290–1298.

Washburn, A. R. (1981a). An upper bound useful in optimizing search for a moving target. *Operations Research, 29*, 1227–1230.

Washburn, A. R. (1981b). *Search and detection*. Linthicum, MD: INFORMS.

Washburn, A. R. (1983). Search for a moving target: The FAB algorithm. *Operations Research, 31*, 739–751.

Second-order Conditions

Conditions Involving Second Derivatives.

Self-Dual Parametric Algorithm

A variation of the simplex method and parametric programming in which the given linear-programming problem is adjusted so that the same parameter is added

to each cost coefficient and each right-hand-side element. By using a sequence of primal and dual simplex transformations, the problem will be optimal for some value of the parameter, with the process continuing until a solution with a zero value of the parameter is found.

See

► Simplex Method (Algorithm)

Semi-Markov Process

A stochastic process that evolves via an embedded discrete-time Markov process, where the times spent in a state before making a transition are independent random variables following general distributions. Generalizes the continuous-time Markov process setting where the time spent in a state is exponentially distributed. Used for analyzing queueing and related systems.

See

► Markov Processes

Semi-Strictly Quasi-Concave Function

A function $f(x)$ is semi-strictly quasi-concave over a convex set S if for any two points $x_1 \neq x_2$ in S and for any $0 < \alpha < 1$, $f(x_2) > f(x_1)$ implies that $f(\alpha\, x_1 + (1 - \alpha)\, x_2) > f(x_1)$.

See

► Concave Function
► Convex Function
► Quasi-Concave Function
► Quasi-Convex Function
► Semi-Strictly Quasi-Convex Function
► Strictly Quasi-Concave Function
► Strictly Quasi-Convex Function

Semi-Strictly Quasi-Convex Function

A function $f(x)$ is semi-strictly quasi-convex over a convex set S if for any two points $x_1 \neq x_2$ in S and for any $0 < \alpha < 1$, $-f(x_2) > -f(x_1)$ implies that $-f(\alpha\, x_1 + (1 - \alpha)\, x_2) > -f(x_1)$.

See

► Concave Function
► Convex Function
► Quasi-Concave Function
► Quasi-Convex Function
► Semi-Strictly Quasi-Concave Function
► Strictly Quasi-Concave Function
► Strictly Quasi-Convex Function

Sensitivity Analysis

Andres Redchuk[1,2] and David Ríos Insua[3]
[1]Universidad Rey Juan Carlos, Mostoles, Madrid, Spain
[2]Universidad Autónoma de Chile, Santiago, Chile
[3]Spanish Royal Academy of Sciences, Madrid, Spain

Introduction

In operations research (OR), sensitivity analysis describes the methods and tools used to study how the output of a model varies with changes in the input data. The input data may refer to parameters affecting the objective functions and/or constraints or to the structure of the problem. Depending on the problem and model, the output could refer to:

- the optimal alternative and/or the optimal value, or,
- a set of alternatives with a certain property. Some examples include the non-dominated set in a multi-objective optimization problem; the set of alternatives satisfying certain constraints in a classification problem; or the set of the, say, five best alternatives.

Typical questions addressed within sensitivity analysis are whether a given optimal solution will remain as such if inputs are changed in a certain way, and, if not, which other alternatives may become

optimal. Finding the most critical directions for changes in inputs that may affect the model output are also relevant sensitivity analysis issues; see French and Ríos Insua (2000), Saltelli et al. (2000), and Saltelli et al. (2004) for reviews.

As motivating examples, consider the following problems:

1. Linear programming. One may be interested in checking how the costs (reduced costs) and/or the right hand side terms and/or the technological matrix terms impact over the optimal solution. A typical question would be: does the optimal solution change if one of the costs increases so much?
2. Decision analysis. One may be interested in checking the impact of the beliefs and preferences, modeled, respectively, through a probability distribution and a utility function, over the optimal alternative and its expected utility. For example, one could wonder how much arelevant binomial parameter and a risk tolerance parameter should be changed to make a certain alternative optimal.
3. Multi-objective decision making with a multi-attribute value function. One may want to check which alternatives become optimal when the weights vary within a range around the current weight settings.

An important, and occasionally controversial, issue in sensitivity analysis is the distinction between decision sensitivity and value sensitivity (Kadane and Srinivasan 1996). A variety of situations may hold. For instance, when performing sensitivity analysis, it may happen that value changes considerably with virtually no change in the optimal alternative.

Motivations

There may be many reasons to check the sensitivity of the output of an OR model to its inputs. A first reason may be the almost ubiquitous uncertainty in the inputs. One may not be willing or capable of assessing such uncertainty with a probability distribution. Then, baseline values for the inputs could be assessed and changes in how they affect the output are observed.

Similarly, the assessment of the inputs might be affected by inherent imprecision and output robustness may be checked. In relation with this, it may be interesting to check the robustness of the output under various input scenarios.

Note also that, since some of the inputs to an analysis may encode the subjective judgments of the decision maker (DM), their implications and possible inconsistencies should be explored. The need for sensitivity analysis is further emphasized by the fact that the assessment of such judgments could be a difficult task. For example, it is frequently mentioned that assessing a subjective probability distribution is involved. Consider the simplest case in which it is desired to elicit a prior over a finite set of states $\theta_i, i \in \{1, \ldots, I\}$. A common technique to assess a precise probability distribution $\pi(\theta_i) = p_i$ proceeds as follows, with the aid of a reference experiment: one progressively bounds $\pi(\theta_i)$ above and below until no further discrimination is possible and then takes the midpoint of the resulting interval as the value of p_i. Instead, one could directly operate with the obtained constraints $\alpha_i \leq \pi(\theta_i) \leq \beta_i$, acknowledging cognitive limitations. This is an especially important point, as the DM's judgments will evolve through the analysis until they are requisite. Sensitivity analysis may guide such process.

In relation with the limitations of elicitation, consider also the situation in which there are several decision makers and/or experts involved in the elicitation. Then it is not even necessarily possible theoretically to obtain a single model: one might be left with only classes of each, corresponding to differing expert opinions, and one may need to study the model under those various settings.

Finally, note that sensitivity analysis may be used to perform value of information calculations that allow one to compute how much to pay for information used to reduce uncertainty in an analysis.

To sum up, sensitivity analysis aims at increasing the confidence in an OR model and its output by providing an understanding of the responses of the model to changes in the inputs.

Foundations

A number of results show that imprecision in model inputs may be dealt with through a class of probability distributions and a class of utility functions. These results have two basic implications. First, they provide a qualitative framework for sensitivity analysis, describing under what conditions the standard and natural sensitivity analysis approach of perturbing the

initial input assessments within some reasonable constraints may be undertaken. Second, they point out to the basic solution concept of robust approaches, thus indicating a key computational objective in sensitivity analysis, as long as the interest is in decision analytic problems: that of non-dominated alternatives. An alternative a dominates another alternative b, if its evaluation is better for each potential input to the analysis. Then, an alternative a is non-dominated if no other feasible alternative dominates it. This corresponds to a Pareto ordering of alternatives based on inequalities on their evaluations.

To construct an appropriate framework for general sensitivity analysis, the standard decision-theoretic axiomatic foundations should be reconsidered to account for imprecision in model inputs. Although this approach does not lead to such well-rounded development as in the precise case, in which various axiomatizations essentially lead to the subjective expected utility model, various partial results do exist (e.g., Ríos Insua 1990 or Walley 1991), leading to, essentially, the same conclusion: imprecise beliefs and preferences may be modeled by a class of priors and a class of utility functions, so that preferences among alternatives may be represented by inequalities of the corresponding posterior expected utilities. The basic argument for such results assumes that the underlying preference relation, rather than being a weak order (complete and transitive), is a quasi order (reflexive and transitive).

Key Approaches to Sensitivity Analysis

Clearly, as there is a large variety of OR models, there is a comparatively large number of approaches to sensitivity analysis. Only a few of the approaches that may be applied to various OR models are described, without much dwelling into their numerical details.

Testing Alternative Inputs

One first generic approach refers to changing the inputs to the model and observe variations in the output. This may be done in several ways.

Trying Other Values

The first approach may be termed the informal one, which considers several inputs and compares the quantity of interest (e.g., the posterior mean, the difference in value between two alternatives, the optimal alternative) under them. The approach is very popular because of its simplicity. While this is a healthy practice and a good way to start a sensitivity analysis, in general this will not be sufficient and more formal analysis should be undertaken: the limited number of priors chosen might not include some which are compatible with the prior knowledge and could lead to very different values of the quantity. Sometimes, the alternative inputs considered are randomly generated.

Parametric Analysis

Another way of changing the inputs is through parametric analysis. Using a baseline input assessment, one determines a relevant direction to perturb the inputs and considers a parametric perturbation along such direction observing whether there is a change, or not, in, e.g., the optimal alternative (Gal and Greenberg 1997).

Global Robustness

Another popular approach in SA is called global sensitivity. All inputs compatible with the prior knowledge available are considered and robustness measures are computed as the inputs vary within that class. Computations are not always easy since they require the evaluation of suprema and infima of quantities of interest. The choice of the class of inputs should be driven by the following goals:

1. the class should be related with the elicitation method used;
2. the class should contain only reasonable inputs, avoiding unreasonable inputs which might erroneously lead to lack of robustness;
3. computation of sensitivity measures should be as simple as possible.

The robustness measures provides, in general, a number that should be interpreted in the following way:

- if the measure is small, then robustness is achieved and any input in the class can be chosen without relevant effects on the quantity of interest;
- if the measure is large, then new data should be acquired and/or further elicitation to narrow the class, recomputing the robustness measure and stopping as before;
- otherwise, if the measure is large and the class cannot be modified, then an input can be chosen in

the class but the relevant influence of the choice over the quantity of interest should be considered carefully.

Given a class of inputs, global sensitivity analysis will usually pay attention to the range of variation of a quantity of interest as the input ranges over the class. As an example, in a decision-theoretic problem, suppose a quadratic loss function is used in a problem. The optimal rule is the posterior expectation. If there is imprecision about the prior, the range of the posterior expectation as the prior ranges in the class would be computed.

Behavior of the Optimal Alternative

The other family of sensitivity analysis approaches studies the behavior of the output of interest under small input perturbations, either via differential approaches or convergence arguments.

Local Sensitivity

Local sensitivity analysis studies the rate of change in inferences and decisions, using functional analysis differential techniques, trying to assess how a small change in the input affects the quantity of interest. The two issues involved refer to choosing the derivative and the corresponding norm, over the appropriate class of inputs. For the first choice, Fréchet derivatives, total derivatives and Gateaux differentials have been used, among others. Divergence measures have been used as well. For the second choice, the total variation, Prohorov, Levy, and Kolmogorov metrics have been used among others. The direction providing the supremum norm is used as the most sensitive direction; alternatively, the average sensitivity is sometimes used integrating the norm along all possible relevant directions.

As an example, Ruggeri and Wasserman (1993) measured the local sensitivity of a posterior expectation with respect to the prior by computing the norm of the Fréchet derivative of the posterior with respect to the prior over several different classes of inputs.

Stability

Stability theory provides another unifying, general sensitivity framework, formalizing the idea that imprecision in elicitation of inputs should not affect the optimal decision greatly. When strong stability holds, a careful enough elicitation leads to decisions with optimal value close to the greatest achievable; when weak stability holds, at least one stabilized decision will have such property. However, when neither concept of stability applies, even small elicitation errors may lead to disastrous results in terms of large losses in value.

Stability theory studies the convergence of decisions, nearly optimal for input sequences converging to the baseline inputs, to the corresponding optimal alternative. The arguments involved refer to the continuity of the relevant operator, e.g., the posterior expected utility functional. Note that stability is not always guaranteed, even in standard problems, as shown e.g., in Kadane and Srinivasan (1996). Such counterexamples show a need for conditions which ensure stability. While these conditions simplify the task of verifying stability, it can still be hard to do so in practice.

An Operational Approach to Sensitivity Analysis

An operational approach to sensitivity analysis in OR models may be described as follows. At a given stage of the analysis, information on the DM's inputs is elicited, and the class of all inputs compatible with such information is considered. The set of non-dominated solutions is approximated. If these alternatives do not differ too much in their value, the analysis may be stopped; otherwise, additional information will be gathered. This would further constrain the class: the set of non-dominated alternatives will be smaller. It is hoped that this iterative process would converge until the non-dominated set is small enough to reach a final decision. It is conceivable in this context that at some stage it might not be possible to gather additional information yet there remain several non-dominated alternatives with very different values. In these situations, ad hoc approaches such as maximin solutions may aid as a way of selecting a single robust solution: each alternative is associated with its worst evaluation, given the current input imprecision. The alternative with best worst evaluation is suggested. Alternatively, a prior over the class of inputs could be built and base choice on expectations over evaluations.

The relevant steps are now described.

Non-dominated Alternatives

As mentioned, a key solution concept is the efficient set, i.e., the set of non-dominated alternatives. In most cases, it is not possible to compute the non-dominated set exactly, and thus approximation schemes are necessary. Typically, one would proceed by randomly sampling the set of alternatives, randomly sampling the set of inputs, and checking dominance among pairs of alternatives. Under appropriate conditions, this sampling scheme is such that the sample non-dominated set converges to the non-dominated set, as the sample size grows.

Extracting Additional Information

In some cases, non-dominance is a very powerful concept leading to a unique non-dominated alternative. However, in most cases the non-dominated set will be too large to imply a final decision. It may happen that there are several non-dominated alternatives and differences in expected utilities are non-negligible. If such is the case, additional information should be sought that would help to reduce the classes, and, perhaps, reduce the non-dominated set. Some tools based on functional derivatives to elicit additional information may be seen in Ríos and Ruggeri (2000). Tools based on distance analysis may be seen in Ríos Insua (1990).

Robust Solutions

When no additional information may be extracted from experts to reduce the set of inputs, and the set of alternatives is still too large, robust solution concepts should be sought. One is the maximin approach which may be considered as an automated method that allows the choice of actions that guard against catastrophic consequences. A maximin approach would be suitable after a sensitivity analysis has been unable to significantly narrow the range of variation, under changes in inputs of the quantity of interest.

Hyperpriors

Another approach to dealing with lack of robustness would be to place a hyperprior on the class of inputs. Indeed, if there were no possibility of obtaining additional information to deal with the lack of robustness, this technique would be recommended, with the hyperprior being chosen in some default fashion.

Misconceptions in Sensitivity Analysis

Some basic issues corresponding to a number of sensitivity analysis approaches are described. Their relevance stems from them corresponding to typical misconceptions.

- It is not enough to study changes in output by trying some other inputs.
- Partial sensitivity studies may not be sufficient: a problem may be insensitive to changes in utility and changes in probability, but sensitive to simultaneous changes in utility and probability.
- When performing sensitivity analysis, there are cases in which the optimal value may change a lot, with virtually no change in the optimal action, even if the utility is fixed.
- Alternatively, there are cases in which the optimal alternative varies widely, but the optimal value does not practically change.
- Big changes in optimal value do not necessarily correspond to big changes in consequences of interest.
- Standard global robustness studies, based for example on ranges of expected utilities of actions, may not be sufficient within a decision-theoretic perspective.

Concluding Remarks

Imprecise probability is a generic term used to describe mathematical models that measure uncertainty without precise probabilities. This is certainly the case with robust Bayesian analysis, but there are many other imprecise probability theories, including upper and lower probabilities, belief functions, Choquet capacities, fuzzy logic, and upper and lower previsions (Walley 1991). Some of these theories, such as fuzzy logic and belief functions, are only tangentially related to sensitivity analysis, whereas others are intimately related. For example, some classes of probability distributions that are considered in Bayesian sensitivity analysis, such as distribution band classes, can also be interpreted in terms of upper and lower probabilities (Ríos Insua et al. 2000). Also, classes of probability distributions used in sensitivity analysis robust Bayesian analysis will typically generate upper and lower previsions as their upper and lower envelopes (Berger et al. 1996).

Finally, sensitivity analysis is linked to uncertainty analysis, which aims at quantifying the uncertainty of the output as a function of the uncertainty in the model inputs.

See

- ▶ Bayes Rule
- ▶ Bayesian Decision Theory, Subjective Probability, and Utility
- ▶ Decision Analysis
- ▶ Decision Maker (DM)
- ▶ Hundred Percent Rule
- ▶ Linear Programming
- ▶ Multiple Criteria Decision Making
- ▶ Nonlinear Programming
- ▶ Parametric Linear Programming
- ▶ Pareto-Optimal Solution
- ▶ Ranging
- ▶ Robustness Analysis
- ▶ Tolerance Analysis

References

Berger, J., Betrò, B., Moreno, E., Pericchi, L., Ruggeri, F., Salinetti, G., & Wasserman, L. (eds.) (1996). *Bayesian robustness*. IMS Lecture Notes.

French, S., & Ríos Insua, D. (2000). *Statistical decision theory*. London: Arnold.

Gal, T., & Greenberg, H. (1997). *Advances in sensitivity analysis and parametric programming*. Boston: Kluwer.

Kadane, J., & Srinivasan, C. (1996). Bayesian robustness and stability. In *Bayesian robustness, IMS Lecture Notes* (Vol. 29, pp. 81–96).

Ríos Insua, D. (1990). *Sensitivity analysis in multiobjective decision making*. New York: Springer.

Ríos Insua, D., & Ruggeri, F. (2000). *Robust Bayesian analysis*. New York: Springer.

Ríos Insua, D., Ruggeri, F., & Martin, J. (2000). Bayesian sensitivity analysis: A review. In A. Saltelli et al. (Eds.), *Handbook on sensitivity analysis*. New York: Wiley.

Ruggeri, F., & Wasserman, L. (1993). Infinitesimal sensitivity of posterior distributions. *Canadian Journal of Statistics, 21*, 195–203.

Saltelli, A., Chan, K., & Scott, M. (2000). *Mathematical and statistical methods: Sensitivity analysis*. New York: Wiley.

Saltelli, A., Tarantola, S., Campolongo, F., & Ratto, M. (2004). *Sensitivity analysis in practice: A guide to assessing scientific models*. New York: Wiley.

Walley, P. (1991). *Statistical reasoning with imprecise probabilities*. London: Chapman and Hall.

Separable Function

A function $f(x_1, \ldots, x_n)$ is a separable function if $f(x_1, \ldots, x_n) = f_1(x_1) + \ldots + f_n(x_n)$. Certain nonlinear-programming problems that contain separable functions can be suitably represented by a linear approximation and solved by a variation of the simplex method.

See

- ▶ Separable-Programming Problem
- ▶ Simplex Method (Algorithm)

Separable-Programming Problem

A nonlinear-programming problem in which some or all of the constraints and the objective function are separable functions of one variable. Using linear approximations to the separable functions, the problem can be approximated and solved by a variation of the simplex algorithm that uses a restricted-basis entry rule.

See

- ▶ Separable Function
- ▶ Special-Ordered Sets (SOS)

Separating Hyperplane Theorem

Let C_1 and C_2 by two nonempty disjoint convex sets in n-dimensional space. Then there exists an n-dimensional hyperplane $\boldsymbol{ax} = \boldsymbol{b}$, $\boldsymbol{a} \neq \boldsymbol{0}$, that separates them. That is, for $\boldsymbol{x}$ in C_1, $\boldsymbol{ax} \leq \boldsymbol{b}$, and for $\boldsymbol{x}$ in C_2, $\boldsymbol{ax} \geq \boldsymbol{b}$.

See

- ▶ Hyperplane

Series Queues

A network of queues with serial routing as found in traditional assembly lines; also called tandem queues.

See

► Networks of Queues

Service Science

James C. Spohrer[1] and Wendy M. Murphy[2]
[1]Almaden Research Center, San Jose, CA, USA
[2]IBM Corporation, Armonk, NY, USA

Introduction

Service science is the interdisciplinary study of service systems and value-cocreation phenomena in the human-made world. However, because the service science community brings together participants from many academic disciplines, industry sectors, and national governments, the community is still working to answer basic questions, such as: (1) what is service? What is a service system? What is value-cocreation? (2) What are the causes of the tremendous observed growth in service activities in both developed and emerging economies around the world? How are product-systems and service-systems alike and different? Is there any longer a meaningful distinction to be made? (3) What should the research agenda associated with service science be? (4) Where is the science in service science? (5) How is service science different from <name your favorite academic discipline that studies complex human-made systems>? (6) If service science is successful, how will the world be different/better?

Service science is an emerging area of research and practice for the interdisciplinary study and improvement of service system structures and value-cocreation mechanisms. Value-cocreation or non-zero-sum games have been growing in quantity and quality, and between more complex entities, throughout human history (Wright 2000). Service systems are human-made systems to improve customer-provider interactions. Service systems include the complex business and societal systems, in which customers, providers, and other stakeholders interact directly and indirectly to create mutual benefits (value-cocreation). Service systems include government, education, health, finance, retail, buildings, communications, energy, food, water, and transportation.

Many of the grand challenges facing the world's growing population, such as hunger, poverty, and discrimination can be framed as lack of access to resources and capabilities in a world of increasing abundance. For the last 50 years the economies of most developed nations and nearly all cities have been dominated by what traditional economists refer to as the service sector, and yet service has been understudied in academia relative to its economic importance (Chesbrough and Spohrer 2006). Economics, marketing and operations (including operations research) were three of the first disciplines to begin scientific study of service and service systems, and more recently management, engineering, computing, design, law, social and behavioral sciences, and other disciplines have applied their unique methods and also established service-oriented specializations. As the world's population shifts from rural to urban areas and as national economies become dominated by the what economist call the service sector, or the knowledge economy, interest in service science has grown.

Also, the service science community has become increasingly focused on the study of holistic service systems, such as cities, universities, luxury resort hotels, and cruise ships that are parts of a complex system of systems. As service science matures operations research is playing a central role and provides the mathematical models and optimization tools for the study of holistic service systems, in which local optimization does not necessarily lead to global optimization and in which small changes in one component system can lead to large consequences in other systems. These are familiar optimization challenges for operation researchers, especially in queueing theory, supply chain optimization, and total quality improvement.

Service Growth

The growing interest in the study of service and service systems can be traced to five factors:

1. Service Sector Growth (OECD 2005): Traditional economic measures of the overall relative percentage of and growth of economic output from service sector and knowledge economy in nations, often referred to as the intangible economy and contrasted with agriculture and manufacturing which create tangible output. This also includes the growth of manufacturing companies growing their service revenue (servitization). Servitization often results because of the growing complexity of manufactured goods, and the need for more customer service to help maintain the product and for users to get the full benefit from the products they purchase.
2. Urban and Knowledge Economy Growth (North 2005: 87–102): The increase in the percentage of the world's population that lives in urban areas as compared with rural areas. The growth of cities and the dependence of cities on universities to ensure skills needed to compete in a global knowledge economy, as well as to provide sustainable innovations that improve quality of life from one generation to the next.
3. IT-Enabled Service Growth (UK Royal Society 2009): The increase in IT enabled service, exemplified by both the rise of internet and web-based service delivery, but also what IBM calls "Smarter Planet" in which natural and human-made systems are becoming instrumented, interconnected, and intelligent. The potential for new types of service that IT-enables creates the possibility of continuous improvement by tapping into increasingly powerful IT, has resulted in the growth of areas such as service computing. This is also tied to the view that "nature's service" that access to resource and capability of the planet are increasingly in jeopardy if greener and more sustainable technology-enabled approaches are not implemented.
4. Grand Challenges Framing (Sen 2001): The increasing realization that grand challenges such as hunger, poverty, discrimination, etc. result from lack of access to resources and capabilities in a world of increasing abundance. Using traditional economist definitions of service, the challenge is not only about increasing the number of farms, factories, and law-enforcement/court-houses, but ensuring access to these resources. Other examples are the large percentage high school drop out rate in the US and other developed nations, especially in urban areas; or joblessness during economic cycles. Recovering from natural disasters can be seen as getting service systems back up and running. Even warfare these days may be viewed as nation-building and building up service systems.
5. Skills for Twenty-first Century (Hefly and Murphy 2008; Donofrio et al. 2010): In a knowledge economy, the challenge is there is too much to teach. Even with specialization, it is important that specialists be able to communicate with other specialists and work on teams together. Specialists or I-shaped people, who have good deep knowledge and problem-solving skills, become better team members when they have broad communications skills, and become what is known as a T-shaped person. T-shaped people are better at team-oriented projects, and in general are faster learners and more adaptable because they already have advanced organizers and knowledge of the main concepts in many domains.

For more on the growth of service and national responses to the need for scientific approaches to service innovation see (US Congress 2007) and (UK Royal Society 2009).

Academic Response: Discipline Growth

While many have argued convincingly that academic institutions have been slow to respond to service growth, nevertheless an ever increasing number of existing disciplines have established service-oriented sub-disciplines. Five of the major areas are:

1. Service marketing (Vargo and Lusch 2004; Zeithaml et al. 2006): Marketing was one of the first disciplines to establish a service-oriented sub-discipline, with service quality a primary measure of concern, largely from the customer perspective. Concepts include service-dominant-logic, the gaps model, linkage research relating employee and customer perceptions of quality, the service profit chain, customer equity, customer co-development of service innovations, and relationship marketing

2. Service engineering (Tien and Berg 2003; Chang 2010; Karwowski and Salvendy 2010): More recently, systems engineers have begun to study service systems, and textbooks have begun to appear. Service engineering is working to identify common building blocks or architectural components of service systems.
3. Service design (Glushko 2010): This is one of the fastest growing areas of service research. Service design has an architectural component like service engineering, but much more emphasis on customer and employee experience during service interactions.
4. Service computing (Demirkan and Goul 2006; Zysman 2006; Zhang 2007; Katzan 2008): Another fast growing area, the rise of service-oriented architectures, web services and smart phones have given a huge boost to this emerging service-oriented sub-discipline of computer science. IEEE and ACM have been collaborating on establishing a standard curriculum in this area.
5. Service operations (Fitzsimmons and Fitzsimmons 2007; Chase 2010; Sampson 2010; Daskin 2010): Operations, which includes operations management, service management, and operations research, was also one of the first discipline areas to establish a service-oriented sub-discipline, with service delivery productivity a primary measure of concern, largely from the provider perspective. Concepts include customer contact theory, unified service theory based on customer input to processes, modeling and optimization and much more.

Some Examples of Major Contributions of Operations Research in Services

- Supply chain; personnel staffing; scheduling
- Pricing optimization
- Layout (queues/waiting – Disney; banks)
- (Help desk) call flow optimization, forecasting
- Projects – critical path, resource constraints (could be inventory management for people)
- Models to assist in forecasting demand for services, regression, econometric, time series
- Managing demand
  - Incentives to affect demand (shifting demand, off peak demand)
  - Overbooking of airline seats
  - Sharing capacity
- Yield management
- Simulations
- Capacity planning
- Queueing models
- Vehicle routing (bus, train service)

Operations researchers are working to address the grand challenges mentioned above. Local optimization does not always lead to global optimization; and in highly interconnected system of systems, a small problem in one place can lead to large problems elsewhere. There are issues of stability and resilience in service networks to consider. Operations research benefits from the interdisciplinary framework of service science, and therefore contributions from operations research to service science, can also impact service marketing, service engineering, service design, and service computing as well. For example, consider some of the human-aspects of the following service system challenges:

1. Staffing. People aren't widgets and perhaps one doesn't *want* to do the assignment the tool "optimally" matched –them with according to existing criteria (skills, availability), or perhaps the customer doesn't like them and so the employee might better be assigned another place or another time.
2. Price optimization. Perhaps perception of service value is resistant to price incentives; maybe models take this into account, but can they be adjusted for each day's mass mental state (a la why the stock market rises and falls – emotion!)
3. Call routing. Maybe the person next on the call queue is tired of answering the same questions and wants a new challenge; is there a way to accommodate the human need to achieve and contribute?
4. Demand management. Overbooking of airline seats is today getting a great deal of customer unhappiness.
5. Queueing and routing. Tarmac waiting time has not been deemed illegal after certain limits.
6. Yield management. One of these days passengers are going to revolt against paying premium for business travel over leisure travel. Companies are eschewing travel for alternative means for people to meet face to face.

As additional disciplines, from management of information systems to areas of the social sciences,

add service-oriented sub-disciplines, there is a growing interest working together to establish a science of service (Ostrom et al. 2010).

Industry Response: Product-Service-Systems Growth

The servitization of traditional manufacturing businesses and the growth of self-service technologies in traditional service businesses have given rise to the concept of product-service-systems (IfM and IBM 2008; Spohrer et al. 2010). Industry practitioners and academic researchers are collaborating on public-private partnerships to engineer smarter product-service-systems in three main areas.

1. Systems associated with moving physical things: Transportation, supply chain, water, air, waste, food, products, energy, electricity, and information and communications technology deal with systems engineered to move physical things. Productivity is a primary concern.
2. Systems that support human activities: Buildings, construction, retail, hospitality, media, entertainment, tourism, sports, financial, business consulting, health, family life, education, and professional life deal with systems engineered to support human activities. Quality is a primary concern.
3. Systems that govern: Cities, security, states, economic development, nations, and the law deal with systems engineered to govern. Compliance and competitiveness are primary concerns.

As product-service-systems become smarter, or more instrumented, interconnected, and intelligent (using analytics to support decision making), there is an ever growing need to model and optimize networks or product-service-system value chains.

Service Science: An Emerging Framework

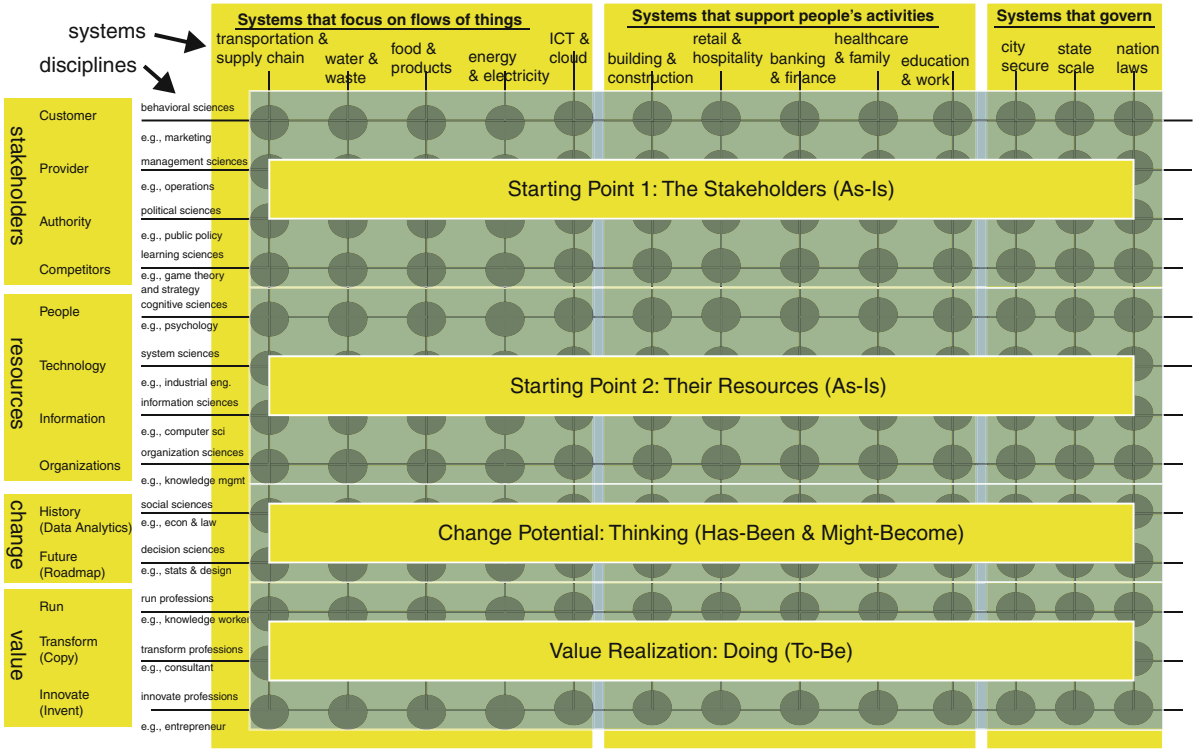
Service science is the interdisciplinary study of service systems and value-cocreation phenomena (Spohrer et al. 2007; Spohrer and Maglio 2008). A service system is a human-made system to improve customer-provider interactions, or value-cocreation. Service science is also an emerging community of academic researchers, industry practitioners, and government policy makers working together to study the entities, interactions, and outcomes associated with the growth of the service system ecology (Spohrer and Maglio 2009, 2010). The "systems and disciplines matrix" is used to visualize the areas of study that is service science, and is shown in Fig. 1 below:

Spohrer and Maglio (2009) describe the foundational concepts of service science (see Fig. 2). Spohrer et al. (2010) provides a glossary of the key terms.

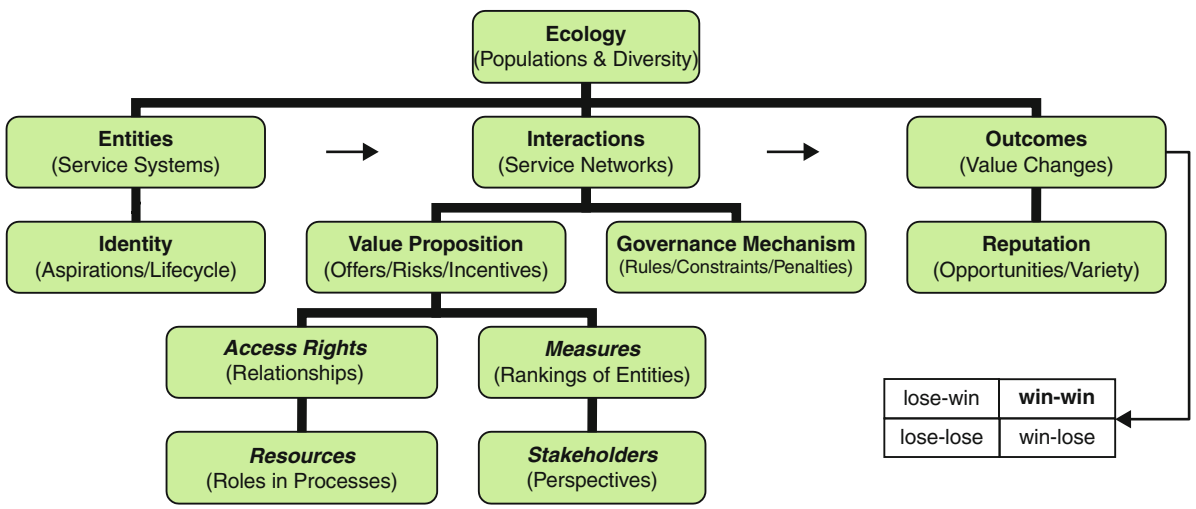
Service science identifies three premises summarized in Fig. 3. Service system entities dynamically configure (transform) four types of resources: people, technology, organizations, and shared information. Service system entities calculate value from multiple stakeholder perspectives, including: customer, provider, authority, and competitor. Service system entities reconfigure access rights to resources by mutually agreed to value propositions, and include: owned-outright, leased-contracted, shared-access, and privileged-access. These concepts and premises allow service scientists to bridge across multiple disciplines that may use different vocabulary or take different perspectives when analyzing service system entity structure or interaction mechanisms.

Concluding Remarks

While the growth of the service science community has been accelerating, nothing is settled and much work remains to be done (Chesbrough and Spohrer 2006; Maglio et al. 2010). Members of the community from marketing refer to service systems entities as resource-integrators, consistent with Service-Dominant Logic of (Vargo and Lusch 2004; Lusch et al. 2008). Members of the community from operations and manufacturing companies may refer to service system entities as product-service-systems. What is clear is that the old distinction between product businesses and service businesses is gradually disappearing as result of the efforts of the service science community, and instead the focus is shifting to entities, interactions, and outcomes. Practitioners are concerned with practical tools and methods that apply service science to guide investments to improve quality, productivity, compliance, and sustainable innovation.

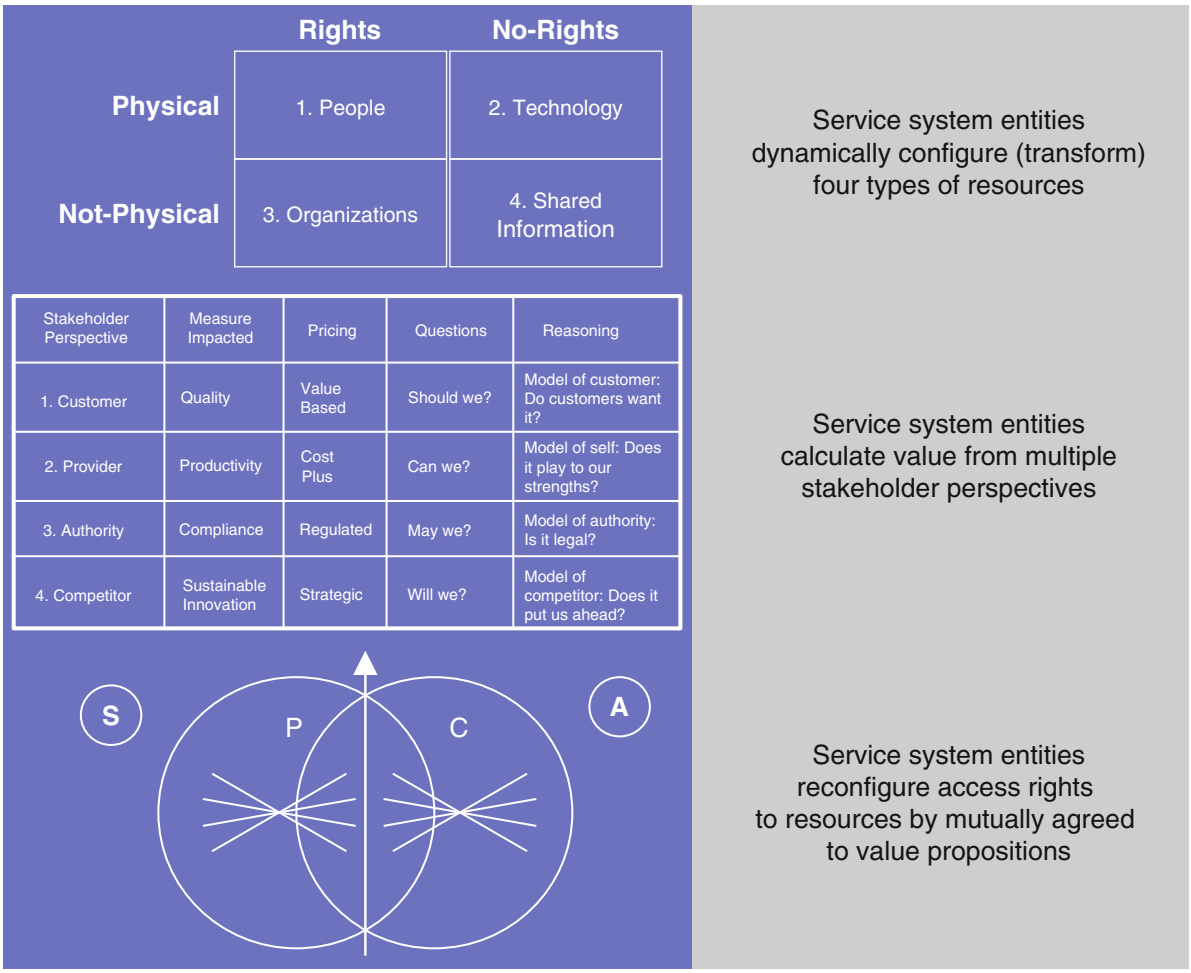


Service Science, Fig. 1 The "systems and disciplines matrix" used to visualize service science, from Spohrer and Maglio



- *Resources:* People, Technology, Information, Organizations
- *Stakeholders:* Customers, Providers, Authorities, Competitors
- *Measures:* Quality, Productivity, Compliance, Sustainable Innovation
- *Access Rights:* Own, Lease, Shared, Privileged

Service Science, Fig. 2 Foundational concepts of service science, from Maglio and Spohrer



Service Science, Fig. 3 Summary of service science based on Spohrer and Maglio

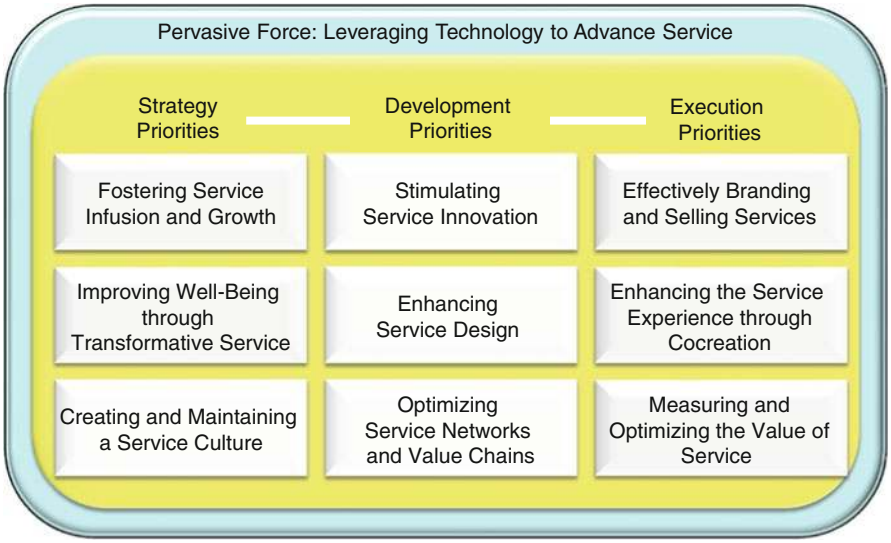
Operations research challenge: Analytic framework needed.

- What type of mathematical models can be constructed?
- What analytic tools exist to study and refine mathematical models?
- Analytic tools – mathematical tools and techniques

No single discipline provides the comprehensive view but by using an interdisciplinary approach, the service science community is making progress. Service science is not just the union of separate disciplines but a deeper integration that may lead to the formation of a service science transdiscipline someday. Mathematics and computer science are existing examples of transdisciplines because they can be used to model aspects of many types of systems. As service scientists develop new tools and methods to model a wide range of service systems and diverse types of value-cocreation phenomena, then service science will likely emerge as a transdiscipline.

The results of an Arizona State University (Ostrom et al. 2010) survey of the global service research community on research priorities for those working to develop a science of service are summarized in Fig. 4.

In sum, service science is an area of research and practice for the interdisciplinary study and improvement of service system structures and value-cocreation mechanisms. Service systems are complex business and societal systems the create benefits for customers, providers, and other



Service Science, Fig. 4 Research priorities for a science of service (Ostrom et al. 2010)

stakeholders, and include all human-made systems that enable and/or grant diverse entities access to resources and capabilities such as transportation, water, food, energy, communications, buildings, retail, finance, health, education, and governance. Many of the grand challenges facing the world's growing population, such as hunger, poverty, and discrimination can be framed as lack of access to resources and capabilities in a world of increasing abundance. For the last 50 years the economies of most developed nations and nearly all cities have been dominated by what traditional economists refer to as the service sector, and yet service has been understudied in academia relative to its economic importance. Economics, marketing and operations (including operations research) were three of the first disciplines to begin scientific study of service and service systems, and more recently management, engineering, computing, design, law, social and behavioral sciences, and other disciplines have applied their unique methods and also established service-oriented specializations. As the world's population shifts from rural to urban areas and as national economies become dominated by what economist call the service sector, or the knowledge economy, interest in service science has grown. Also, the service science community has become increasingly focused on the study of holistic service systems, such as cities, universities, luxury resort hotels, and cruise ships that are parts of a complex system of systems. As service science matures operations research is playing a central role and provides the mathematical models and optimization tools for the study of holistic service systems, in which local optimization does not necessarily lead to global optimization and in which small changes in one component system can lead to large consequences in other systems. These are familiar optimization challenges in for operation researchers, especially in queueing theory, supply chain optimization, and total quality improvement (Daskin 2010).

See

- ▶ Operations Management
- ▶ Queueing Theory
- ▶ Simulation of Stochastic Discrete-Event System
- ▶ Supply Chain Management

References

Chang, C. M. (2010). *Service systems management and engineering: Creating strategic differentiation and operational excellence*. Hoboken: Wiley.

Chase, R. B. (2010). Revisiting "where does the customer fit in a service operation?" Background and future development of contact theory. In P. P. Maglio, C. A. Kieliszewski, & J. C. Spohrer (Eds.), *Handbook of service science* (pp. 11–18). New York: Spring.

Chesbrough, H., & Spohrer, J. (2006). A research manifesto for services science. *Communications of the ACM, 49*(7), 35–40.
Daskin, M. S. (2010). *Service science*. Hoboken: Wiley.
Demirkan, H., & Goul, M. (2006). Towards the service-oriented enterprise vision: Bridging industry and academics. *Communications of the AIS, 18*(26), 546–556.
Donofrio, N., Sanchez, C., & Spohrer, J. (2010). Collaborative innovation and service systems: Implications for institutions and disciplines. In D. Grasso & M. Burkins (Eds.), *Holistic engineering education: Beyond technology* (pp. 243–270). New York: Springer.
Fitzsimmons, J. A., & Fitzsimmons, M. J. (2007). *Service management: Operations, strategy, information technology* (6th ed.). New York: McGraw-Hill Irwin.
Glushko, R. J. (2010). Seven contexts for service system design. In P. P. Maglio, J. A. Kieliszewski, & J. C. Spohrer (Eds.), *Handbook of service science* (pp. 219–250). New York: Spring.
Hefly, B., & Murphy, W. (2008). *Service science, management, and engineering: Education for the 21st century*. New York: Springer.
IfM, & IBM. (2008). *Succeeding through service innovation: A service perspective for education, research, business and government*. Cambridge, UK: University of Cambridge Institute for Manufacturing (IfM).
Karwowski, W., & Salvendy, G. (2010). *Introduction to service engineering*. Hoboken: Wiley.
Katzan, H. (2008). *Service science: Concepts, technology, management*. Bloomington: iUniverse.
Lusch, R. F., Vargo, S. L., & Wessels, G. (2008). Toward a conceptual foundation for service science: Contributions from service-dominant logic. *IBM Systems Journal, 47*(1), 5–14.
Maglio, P. P., Kieliszewski, C. A., & Spohrer, J. C. (2010). *Handbook of service science*. New York: Springer.
North, D. C. (2005). *Understanding the process of economic change*. Princeton: Princeton University Press.
OECD. (2005). *Growth in services: Fostering employment, productivity, and innovation*. Meeting of the OECD Council at Ministerial Level, 2005. No. 83117. http://www.oecd.org/dataoecd/58/52/34749412.pdf
Ostrom, A. L., Bitner, M. J., Brown, S. W., Burkhard, K. A., Goul, M., Smith-Daniels, V., Demirkan, H., & Rabinovich, E. (2010). Moving forward and making a difference: Research priorities for the science of service. *Journal of Service Research, 13*(1), 4–36.
Sampson, S. E. (2010). The unified service theory: A paradigm for service science. In P. P. Maglio, C. A. Kieliszewski, & J. C. Spohrer (Eds.), *Handbook of service science* (pp. 107–132). New York: Spring.
Sen, A. (2001). *Development as freedom*. New York: Knopf.
Spohrer, J., & Maglio, P. P. (2008). The emergence of service science: Toward systematic service innovations to accelerate co-creation of value. *Production and Operations Management, 17*, 1–9.
Spohrer, J. C., & Maglio, P. P. (2009). Service science: Toward a smarter planet. In W. Karwowski & G. Salvendy (Eds.), *Introduction to service engineering*. Hoboken: Wiley.
Spohrer, J. C., & Maglio, P. P. (2010). Toward a science of service systems: Value and symbols. In P. P. Maglio, C. A. Kieliszewski, & J. C. Spohrer (Eds.), *Handbook of service science* (pp. 157–195). New York: Spring.
Spohrer, J., Maglio, P. P., Bailey, J., & Gruhl, D. (2007). Steps toward a science of service systems. *Computer, 40*, 71–77.
Spohrer, J., Gregory, M., & Ren, G. J. (2010). The Cambridge-IBM SSME white paper revisited. In P. P. Maglio, C. A. Kieliszewski, & J. C. Spohrer (Eds.), *Handbook of service science* (pp. 677–706). New York: Spring.
Tien, J. M., & Berg, D. (2003). A case for service systems engineering. *Journal of Systems Science and Systems Engineering, 1*(12), 13–38.
UK Royal Society. (2009). *Hidden wealth: The contribution of science to service sector innovation* (RS Policy document 09/09. RS1496).
US Congress. (2007). *America COMPETES Act. H.R. 2272. Sec. 1005. Study of service science. Became a law on August 9, 2007 by President George W. Bush.* http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=110_cong_bills&docid=f:h2272enr.txt.pdf
Vargo, S. L., & Lusch, R. F. (2004). Evolving to a new dominant logic for marketing. *Journal of Marketing, 68*(1), 1–17.
Wright, R. (2000). *Nonzero: The logic of human destiny*. New York: Vintage/Random House.
Zeithaml, V. A., Bitner, M. J., & Gremler, D. D. (2006). *Services marketing: Integrating customer focus across the firm* (4th ed.). New York: McGraw-Hill Irwin.
Zhang, L. J. (2007). *Modern technologies in web services research*. Hershey: IGI Publishing.
Zysman, J. (2006). The 4th service transformation: The algorithmic revolution. *Communications of the ACM, 49*(7), 48.

Service Systems

Systems in which the workers provide a service to customers, as opposed to manufacturing systems where workers produce or assemble products or goods. Examples of service systems include financial services (such as old-fashioned banks), healthcare systems, and call centers. Both manufacturing and service systems are modeled in operations research using queueing models.

See

▶ Call and Contact Centers
▶ Networks of Queues
▶ Operations Management
▶ Queueing Theory
▶ Service Science
▶ Supply Chain Management

Set-covering Problem

The set-covering problem is an integer-programming problem defined as follows:
- Minimze $\boldsymbol{cx}$
- subject to $\boldsymbol{Ex} \geq \boldsymbol{e}$

where the components of $\boldsymbol{E}$ are either 1 or 0, the components of the column vector $\boldsymbol{e}$ are all 1's, and the variables are restricted to be either 1 or 0. The idea of the problem is to find the minimum cost set of column from $\boldsymbol{E}$ such that the 1's in vector $\boldsymbol{e}$ are covered by at least one of the 1's in the selected set of columns. Note that multiple coverage is allowed.

See

- ▶ Bin-Packing
- ▶ Packing Problem
- ▶ Set-partitioning Problem

Set-partitioning Problem

The set-partitioning problem is an integer-programming problem defined as follows:
- Minimize $\boldsymbol{cx}$
- subject to $\boldsymbol{Ex} = \boldsymbol{e}$

where the components of $\boldsymbol{E}$ are either 1 or 0, the components of the column vector $\boldsymbol{e}$ are all 1 s, and the variables are restricted to be either 1 or 0. It is similar to a set-covering problem except multiple coverage is allowed in that more general setting.

See

- ▶ Packing Problem
- ▶ Set-covering Problem

SEU

Subjective expected utility.

See

- ▶ Decision Analysis

Shadow Prices

The optimal dual variables (marginal values) to a linear-programming problem. For an activity-analysis and similar problems, the shadow price associated with a constraint can be interpreted as the change in the value of the objective function per unit increase of the constraint's right-hand-side (resource).

See

- ▶ Complementarity Applications
- ▶ Lagrange Multipliers
- ▶ Marginal Value

Shapley Value

- ▶ Game Theory
- ▶ Group Decision Making

Shell

An expert system development tool providing a pre-fabricated inference engine.

See

- ▶ Expert Systems
- ▶ Inference Engine

Shewhart Chart

- ▶ Quality Control

Shortest Path Problem

- ▶ Shortest-route Problem

Shortest-route Problem

A network problem where the goal is to find the shortest route from a home (source) node to a destination node, or the shortest routes from a home node to all other nodes. This can be formulated as a linear-programming problem and solved by the simplex method, but special shortest route algorithms exist that are computationally more efficient.

See

- ▶ Dijkstra's Algorithm
- ▶ Simplex Method (Algorithm)

Signomial Programming

- ▶ Geometric Programming

SIMD

Single Instruction, Multiple Data. A class of parallel computer architectures in which a single stream of instructions controls multiple processing elements. Processors synchronously perform the same computations on differing data.

See

- ▶ Parallel Computing

Simple Upper-bounded Problem (SUB)

A linear-programming problem in which some or all of the variables x_j are constrained by upper-bound conditions of the form $x_j \leq u_j$, where u_j is a given finite bound. It can be solved by a special adaptation of the simplex method in which the upper-bounded constraints are considered implicitly.

See

- ▶ Linear Programming
- ▶ Simplex Method (Algorithm)

Simplex

A polyhedron of the form $x_1 + \ldots + x_n \leq 1$, $x_j \geq 0$. Also, a simplex is the convex hull of $n + 1$ points in general position in Euclidean n-space.

Simplex Method (Algorithm)

A computational procedure for solving linear-programming problems of the form: Minimize (maximize) $\boldsymbol{cx}$, subject to $\boldsymbol{Ax} = \boldsymbol{b}$, $\boldsymbol{x} > \boldsymbol{0}$, where $\boldsymbol{A}$ is an $m \times n$ matrix, $\boldsymbol{c}$ is an n-dimensional row vector, $\boldsymbol{b}$ is an m-dimensional column vector, and $\boldsymbol{x}$ is an n-dimensional variable vector. The simplex method was developed by George B. Dantzig in the late 1940s. The method starts with a known basic feasible solution or an artificial basic solution, and, given that the problem is feasible, finds a sequence of basic feasible solutions (extreme-point solutions) such that the value of the objective function improves or does not degrade. Under a nondegeneracy assumption, the simplex algorithm will converge in a finite number of steps, as there are only a finite number of extreme points and extreme directions of the underlying convex set of solutions. At most, m variables can be in the solution at a positive level. In each step (iteration) of the simplex method, a new basis is found and developed by applying Gaussian elimination in a manner that preserves the nonnegativity of the solution. The elimination step replaces a variable in the current solution with a new one. The inverse of the basis (revised-simplex method) is used to develop a pricing vector for "pricing out" the variables not in the current basic solution and to select one to enter the solution if the current solution is not optimal. The optimal solution to the corresponding dual problem is also generated by the simplex method as part of the solution to the original, primal problem. The simplex method has been implemented on just about all major computer systems and a wide

range of simplex-based software is available for personal computers and in spreadsheets.

See

- ▶ Artificial Variables
- ▶ Dual Linear-Programming Problem
- ▶ Linear Programming
- ▶ Prices
- ▶ Primal Problem
- ▶ Revised Simplex Method

Simplex Multipliers

- ▶ Multiplier Vector

Simplex Tableau

A schematic, numerical representation that displays the transformed data set associated with a basic solution to a linear-programming problem. For the problem: Minimize $\boldsymbol{cx}$, subject to $\boldsymbol{Ax} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$, if the $m \times m$ matrix $\boldsymbol{B}$ is a feasible basis and the $m \times 1$ row vector c_0 the ordered cost coefficients for the variables in the basis, then the simplex tableau displays the following information in an $(m + 1) \times (n + 1)$ rectangular matrix

$$\begin{bmatrix} \boldsymbol{B}^{-1}\boldsymbol{A} & \boldsymbol{B}^{-1}\boldsymbol{b} \\ \boldsymbol{\pi A} - \boldsymbol{c} & \boldsymbol{\pi b} \end{bmatrix}$$

where $\boldsymbol{\pi}$, the pricing vector, is equal to $\boldsymbol{c}_0 \boldsymbol{B}^{-1}$. The last or $(m + 1)$st, row of the tableau contains the reduced costs and the current value of the objective function, respectively. If the simplex method is in Phase I, then an additional row is added to the tableau, which contains the reduced costs associated with the artificial basis. In some arrangements of the tableau, the rows associated with the reduced costs are given at the top of the tableau; also, a reduced tableau is obtained by leaving out the columns that correspond to the columns in the basis as they transform to unit columns with reduced costs of zero. The simplex tableau is useful when solving small problems by hand and as an instructional tool. Computer-based simplex method software do not use the tableau as it is computationally inefficient, instead using some form of the revised simplex method.

See

- ▶ Basis
- ▶ Phase I Procedure
- ▶ Phase II Procedure
- ▶ Revised Simplex Method
- ▶ Simplex Method (Algorithm)

Simulated Annealing

Balram Suman
Energy Technology Company, Chevron Corporation, Houston, TX, USA

Introduction

Simulated annealing (SA) is a compact and robust technique to solve single and multiple objective optimization problems with a substantial reduction in the computation time. The method is based on an analogy with the way metals cool and anneal. When a liquid metal is cooled slowly, its atoms form a pure crystal corresponding to the state of minimum energy for the metal. In contrast, when cooled quickly, the metal reaches a state with higher energy (imperfect crystal). Kirkpatrick et al. (1983) and Cerny (1985) showed that a model for simulating the annealing of solids, proposed by Metropolis et al. (1953), could be used for optimization where the objective function to be minimized corresponds to the energy of states of the metal. Since the late 1980s, SA has received significant attention to solve optimization problems where a desired global optimum is hidden among many poor local optima. Thus, SA has become one of the many heuristic approaches designed to give a good, not necessarily optimal solution.

Key Advantages of this Approach

(i) It is very simple to formulate and it can handle mixed discrete and continuous problem with ease.

(ii) The method is also efficient and has low memory requirement. (iii) It takes less CPU time than using a genetic algorithm because it finds the solution using a point-by-point iteration rather than a search over a population of individuals.

With the initiation of SA, the method has been used to solve combinatorial optimization problems. SA can be considered as one type of randomized heuristic approaches for combinatorial optimization problems. Many combinatorial problems belong to a class known as NP-hard problems, whose computation time increases with N as exp(constant $\times$ N). A well-known traveling salesman problem belongs to this class. The salesman visits N cities (with given positions) only once and returns to his city of origin. The objective is to make the route as short as possible. Later, SA was extended to solve single and multi-objective optimization problems with continuous N-dimensional control space. Summary of these approaches is presented in Van Laahoven and Aarts (1987). Surveys on single objective SA have been well documented (Collins et al. 1988; Rutenbar 1989; Aarts and Korst 1989; Eglese 1990, and Reeves 1993). While much of the work on SA to focused on combinatorial optimization (and integer programming) problems, there are also articles on the use of SA for continuous variables (Dekkers and Aarts 1991). A recent review on SA for single and multi-objective optimization problems was presented by Suman and Kumar (2006).

Applications of SA

SA has been greatly in use in operational research problems. Chen et al. (1988) reported a new approach to setup planning of prismatic parts using Hopfield neural net coupled with SA. Sridhar and Rajendran (1993) described three perturbation schemes to generate new sequences for solving the scheduling problem in cellular manufacturing system. Suresh and Sahu (1994) used SA for assembly line balancing. They only considered single objective problems. They found that SA performed at least as well as the other approaches. Meller and Bozer (1996) applied SA to facility layout problems with single and multiple floors. The facility layout problem is highly combinatorial in nature and generally exhibits many local minima. SA achieves low-cost solutions that are much less dependent on the initial layout than other approaches. Mukhopadhyay et al. (1998) used SA to solve the problem of Flexible Manufacturing system (FMS) machine loading with the objective of minimizing the system imbalance. Kim et al. (2002) considered a multi-period multi-stop transportation planning problem in a one-warehouse multi-retailer distribution system to determine the routes of vehicles and delivery quantities for each retailer. They suggested a two-stage heuristic algorithm based on SA as an alternative for large problems that can not be solved by the column-generation algorithm in a reasonable computation time to minimize the total transportation distance for product delivery over the planning horizon while satisfying demands of the retailers. Golenko-Ginzburg and Sims (1992) defined a priority list to be any permutation of a set of symbols where the symbol for each job appears the same number of times as its operations. Every priority list can be associated in a natural way with a feasible schedule and every feasible schedule arises in the same way. Therefore, priority lists are a representation of feasible schedules that avoid the problems normally associated with schedule infeasibility. Shutler (2003) presented priority list based Monte Carlo implementation of SA, which was competitive with the current leading schedule based SA and tabu search heuristics. New job sequences were generated with a proposed perturbation scheme called the modified insertion scheme (MIS), which has been used in the proposed SA algorithm to arrive at a near global optimum solution. The SA algorithm using the proposed MIS gave substantial improvement in system imbalance. Its other applications were presented by machine loading problem of FMS (Swarnkar and Tiwari 2004), part classification (Tiwari and Roy 2003), resource-constrained project scheduling (Cho and Kim 1997). McCormick and Powell (2004) described a two-stage SA algorithm to derive pump schedules for water distribution in a time short enough for routine operational use. They built the model based on automatic interaction with a hydraulic simulator, which deals with non-linear effects from reservoir-level variations.

Adaptive simulated annealing (ASA) offers a viable optimization tool for tackling these difficult nonlinear optimization problems (Chen and Luk 1999). Optimization of batch distillation processes, widely used in chemical industry can be solved using SA.

Hanke and Li (2000) showed the potential of SA for developing optimal operation strategies for batch chemical processes. SA was applied in antenna array synthesis (Girard et al. 2001), multimedia data placement (Terzi et al. 2004) molecular physics. Suman (2002, 2004, 2005) used SA-based multi-objective algorithms to optimize the profit and its sensitivity of a refinery model problem. Suman (2003) applied five simulated annealing based multi-objective algorithms to find a Pareto set of solutions of a system reliability multi-objective optimization problem in a short time. Kumral (2003) applied chance-constrained programming based on multi-objective SA to optimize blending of different available ores in a way that expected value and standard deviation of the cost of buying ores is minimized while satisfying the quality specifications.

Application of SA is not restricted to optimization of nonlinear objective function, it was also applied for many other purposes. Bell et al. (1987) used it to cluster tuples in databases. They attempted to use SA in circuit board layout design and it suggests that it would be advantageously applied to clustering tuples in database in order to enhance responsiveness to queries. SA has not only been applied for optimization but also for recognition of patterns and object classification (Liu and Huang (1998), Yip and Pao (1995), Starink and Barker (1995)). Liu and Huang (1998) proposed hybrid pattern recognition based on the evolutionary algorithms with fast SA that can recognize patterns deformed by transformation caused by rotation, scaling or translation, singly or in combination. Object recognition problem as matching of a global model graph with an input scene graph representing either a single object or several overlapping objects has been formulated. Chu et al. (1996) used SA to analyze the network of interacting genes that control embryonic development and other biological processes.

SA for Optimization

SA for Single Objective Optimization

A solution space (S) is a finite set of all solutions and the objective function (f) is a real-valued function defined for the members of S. The minimization problem can be formulated to find a solution or state, $i \in S$, which minimizes f over S.

A simple form of a local search algorithm, say a descent method, starts with an initial solution. In the neighborhood of this solution a new solution is generated using suitable algorithms and the objective function is calculated. If a reduction in the objective function is observed, the current solution is updated. Otherwise, the current solution is retained and the process is repeated until no further reduction in the objective function is obtained. Thus, the search terminates with a local minimum, which may or may not be the true global minimum. Due to this disadvantage, this algorithm is not relied on, though this is simple and easy to execute. In SA, instead of this strategy, the algorithm attempts to avoid being trapped in a local minimum by sometimes accepting even a worse move. The acceptance and rejection of the worse move is controlled by a probability function. The probability of accepting a move, which causes an increase δ in f, is called the acceptance function. It is normally set to $\exp(-\delta/T)$ where T is a control parameter, corresponding to the temperature in analogy with the physical annealing. This acceptance function implies that a small increase in f is more likely to be accepted than a large increase in f. With high T, most uphill moves are accepted, but at low T, fewer uphill moves get accepted. Therefore, SA starts with a high temperature to avoid being trapped in local minimum. The algorithm proceeds by attempting a certain number of moves at each temperature and decreasing the temperature slowly. The SA-based algorithm for a single objective optimization problem is illustrated in Table 1. Similar to other heuristic optimization techniques, there is a chance of revisiting a solution multiple times in SA, leading to extra computational time without any improvement in the optimal solution. A strategy of avoiding such moves would improve SA efficiency.

SA for Multi-objective Optimization

Real life problems are multi-objective in nature. Researchers have developed many multi-objective optimization procedures, which has a number of disadvantages and pitfalls. Increasing acceptance of SA and other heuristic algorithms is due to their ability to: (1) find multiple solutions in a single run, (2) work without derivatives, (3) converge speedily to Pareto-optimal solutions with a high degree of accuracy, (4) handle both continuous function and combinatorial

Simulated Annealing, Table 1 SA based Algorithm for single objective

1. Initialize the temperature.
2. Start with a randomly generated initial solution vector, X and generate the objective function.
3. Give a random perturbation and generate a new solution vector, Y, in the neighborhood of current solution vector, X, revaluate the objective function and apply a penalty function approach to the objective function, if necessary.
4. If the generated solution vector is archived, make it the current solution vector by putting $X = Y$. Update the existing optimal solution and go to step 6.
5. Else accept Y with probability $P = \exp(-\Delta s/T)$, (1) where $\Delta s = Z(Y) - Z(X)$. If the solution is accepted, replace X with Y.
6. Decrease the temperature periodically.
7. Repeat step 2 through 6 until stopping criterion is met.

optimization problems with ease, (5) be less susceptible to the shape or continuity of the Pareto front.

Acceptance of SA in a multi-objective framework is due to its simplicity and capability of producing a Pareto set of solutions in a single run with little computational cost. In addition, the method is not susceptible to the shape of the Pareto set, whereas these two issues are real concerns for mathematical programming techniques. The first multi-objective version of SA has been proposed by Serafini (1985, 1992). The algorithm is similar to the SA-based algorithm for single objective problems. The method uses a modification of the acceptance criterion of solutions in the original algorithm. Various alternative criteria have been investigated in order to increase the probability of accepting non-dominated solutions. A special rule given by the combination of several criteria has been proposed in order to concentrate the search almost exclusively on the non-dominated solutions. Thereafter, this method was applied by Ulungu and Teghem (1994). They only used the notion of the probability in the multi-objective framework. Serafini (1994) applied a simulated annealing algorithm on the multi-objective framework. A target-vector approach to solve a bi-objective optimization problem was used. Ulungu et al. (1999) proposed a complete MOSA algorithm which they tested on a multi-objective combinatorial optimization problem. A weighted aggregating function to evaluate the fitness of solutions was used. The algorithm worked with only one current solution but maintained a population with the non-dominated solutions found during the search. This method was further improved and extensively tested by Ulungu et al. (1998) and an interactive version of MOSA was used to solve an industrial problem (UMOSA method). Tuyttens et al. (2000) used the MOSA (multi-objective optimization simulated annealing) method for the bicriteria assignment problem.

Suppapitnarm and Parks (1999) proposed a different SA-based approach to tackle multi-objective problems (SMOSA method). The algorithm uses only one solution and the annealing process adjusts each temperature independently according to the performance of the solution in each criterion during the search. An archive set stores all the non-dominated solutions between each of the multiple objectives. An acceptance probability formulation based on an annealing schedule with multiple temperatures (one for each objective) was proposed. The acceptance probability of a new solution depends on whether or not it is added to the set of potentially Pareto- optimal solutions. If it is added to this set, it is accepted to be the current solution with probability equal to one. Otherwise, a multi-objective acceptance rule is used.

Czyżak and Jaszkiewicz (1998) proposed another way to adopt simulated annealing to a multi-objective framework, (PSA method). Czyżak et al. (1994) combined unicriterion simulated annealing and a genetic algorithm to provide efficient solutions of multicriteria shortest path problem. A population-based extension of simulated annealing proposed for multi-objective combinatorial optimization problems were used. The population of solutions explored their neighborhood similarly to the classical simulated annealing, but weights for each objective tuned in each iteration. The weights for each solution were adjusted in order to increase the probability of moving away from its closest neighborhood in a similar way as in the multi-objective tabu search.

Suman (2002 and 2003) proposed two different SA-based approaches (WMOSA and PDMOSA) to tackle the multi-objective optimization of constrained problems. Suman (2003) also tested five simulated annealing algorithms for the system reliability optimization problem. The goal of these methods

Simulated Annealing, Table 2 SA based Algorithm for Multi-objective Optimization

The basic steps involved in the SMOSA algorithm for a problem having N objective functions and n decision variables are as follows:
1. Start with a randomly generated initial solution vector, X (an n x 1 vector whose elements are decision variables) and evaluate all objective functions and put it into a Pareto set of solutions.
2. Give a random perturbation and generate a new solution vector, Y,in the neighborhood of current solution vector, X, reevaluate the objective functions and apply a penalty function approach to the corresponding objective functions, if necessary.
3. Compare the generated solution vector with all the solutions in the Pareto set and update the Pareto set, if necessary.
4. If the generated solution vector is archived, make it the current solution vector by putting $X = Y$ and go to step 7.
5. If the generated solution vector is not archived, accept it with the probability: $P = \min\left(1, \prod_{i=1}^{N} \exp\left\{\frac{-\Delta s_i}{T_i}\right\}\right)$, (2) where $\Delta s_i = f(z_i(Y) - z_i(X))$ with the function, f, depends on the choice of a SA based algorithm. If the generated solution is accepted, make it the current solution vector by putting $X = Y$ and go to step 7.
6. If the generated solution vector is not accepted, retain the earlier solution vector as the current solution vector and go to step 7.
7. Periodically, restart with a randomly selected solution from the Pareto set. While periodically restarting with the archived solutions, Suppapitnarm et al. (2000) have recommended biasing towards the extreme ends of the trade-off surface.
8. Reduce the temperature periodically using a problem dependent annealing schedule.
9. Repeat steps 2 to 8, until a predefined number of iterations is carried out.

was to generate a set of solutions, which are a good approximation to the whole set of efficient (non-dominated or Pareto-optimal) solutions in a relatively short time.

Suman (2005) further improved the SA based multi-objective algorithm so that the user does not need to give a predefined number of maximum iterations. All simulated annealing multi-objective algorithms have the advantage that they allow the full exploration of the solution space: because the starting temperature is high, any move is accepted. The move becomes selective as temperature decreases with an increase in the iteration number and by the end it accepts only the improving moves. Suman et al. (2010) proposed an OSA (orthogonal simulated annealing) algorithm incorporates an orthogonal based on experiment design (OED) with a simulated annealing based multi-objective algorithm aiming to provide an efficient multi-objective algorithm. A typical multi-objective algorithm based on SA is presented in Table 2.

Annealing Schedule

Parameters of an annealing schedule for the SA-based algorithm determine performance of the SA-based algorithm. A high cooling rate leads to poor results due to lack of representative states, while a low cooling rate requires a very high computation time to get the solution. The following choices must be made for any implementation of SA and they constitute the annealing schedule: initial value of temperature (T), cooling schedule, number of iterations to be performed at each temperature and stopping criterion to terminate the algorithm.

Initial value of temperature (T)

The initial temperature is chosen such that it can capture the entire solution space. One choice is a very high initial temperature to increase the solution space. But, at a high initial temperature, SA performs a large number of iterations, which may be even without generating better solutions. Therefore, the initial temperature is chosen by experimentation depending upon the nature of the problem. The range of change, Δf_0 in the value of the objective function with different moves is determined. The initial value of temperature should be considerably larger than the largest Δf_0 encountered. van Laarhoven and Aarts (1987) proposed a method to select the initial temperature based on the initial acceptance ratio χ_0, and the average increase in the objective function, Δf_0:

$$T = -\frac{\Delta f_0}{\ln(\chi_0)}, \tag{3}$$

where χ_0 is defined as the number of accepted bad moves divided by the number of attempted bad moves. A similar formula has been proposed by Sait and Youssef (1999) with the only difference being in the definition of χ_0. They defined χ_0 as the number of accepted moves divided by the number of attempted moves. A simple way of selecting initial temperature has been proposed by Kouvelis and Chiang (1992). They proposed to select the initial temperature by the formula:

$$P = \exp\left(\frac{-\Delta s}{T}\right) \tag{4}$$

where P is the initial average probability of acceptance and taken in the range of 0.50 to 0.95.

Cooling Schedule

The cooling schedule determines the functional form of the change in temperature required in SA. The earliest annealing schedules were based on the analogy with physical annealing. Therefore, they set the initial temperature high enough to accept all transitions, which means heating up substances until all the molecules are randomly arranged in liquid. A proportional temperature can be used i.e., $T\,(i+1) = \alpha T\,(i)$ where α is a constant known as the cooling factor and it can vary from 0.80 to 0.99. Finally, the temperature becomes very small and it will not search any smaller energy level. It is called the frozen state.

Three important cooling schedules are logarithmic, Cauchy and exponential (Azencott 1992). SA converges to the global minimum of the cost function if the temperature change is governed by a logarithmic schedule in which the temperature $T\,(i)$ at step i is given by $T\,(i) = T_o/\log i$ (Geman and Geman 1984). This schedule requires the move to be drawn from a Gaussian distribution. A faster schedule is the Cauchy schedule in which $T\,(i) = T_o/i$ converges to the global minimum when moves are drawn from a Cauchy distribution (Szu and Hartley 1987). It is sometimes called "fast simulated annealing". The fastest is exponential or geometric schedule in which $T\,(i) = T_o\exp(-Ci)$ where C is a constant. There is no rigorous proof of the convergence of this schedule to the global optimum although good heuristic arguments for its convergence have been made for a system in which annealing state variables are bounded (Ingber 1989).

A proportional temperature cooling schedule does not lead to equilibrium at low temperature. Therefore, there is a need for a small number of transitions to be sufficient to reach the thermal equilibrium. However, a serious attempt was made with adaptive simulated annealing (Gong et al. 2001). Annealing schedules use information about the cost function obtained during the annealing run itself. Such a schedule is called an adaptive cooling schedule (Ingber 1989; Azizi and Zolfaghari 2004). An adaptive cooling schedule tries to keep the annealing temperature close to the equilibrium as well as reducing the number of transitions to reach equilibrium. It adjusts the rate of temperature decrease based on the past history of the run. Otten and van Ginneken (1984) proposed the following cooling schedule:

$$T_{i+1} = T_i - \frac{1}{M_k}\frac{T_k^3}{\sigma^2(T_i)} \tag{5}$$

where σ^2 is the variance of the objective function at equilibrium and M_k is given by

$$M_k = \frac{f_{\max} + T_i \ln(1+\delta)}{\sigma^2(T_i)\ln(1+\delta)} T_i \tag{6}$$

where $f_{\max}$ is an estimated maximum value of the objective function.

Similar to equation (3.3), Van Laarhoven and Aarts (1987) proposed the following cooling schedule:

$$T_{i+1} = \frac{T_i}{1 + \frac{\ln(1+\delta)T_i}{3\sigma T_i}} \tag{7}$$

where δ is a small real number.

Another adaptive cooling schedules is the adaptive schedule of Lam. The Lam schedule (Lam and Delosme 1988a, 1988b) has been derived by optimizing the rate at which temperature can be decreased subject to the constraint of maintaining quasi-equilibrium. It is given as:

$$S_{k+1} = S_k + \lambda\left(\frac{1}{\sigma(S_k)}\right)\left(\frac{1}{S_k^2\,\sigma^2(S_k)}\right)\left(\frac{4\rho_o(S_k)(1-\rho_o(S_k))^2}{(2-\rho_o(S_k))^2}\right) \tag{8}$$

where $S_i = 1/T_i$ and T_i is the temperature at ith iteration of the cost function E. The quantity $\sigma(S_k)$ is the standard deviation of E at this step and $\rho_o(S_k)$ is the acceptance ratio; that is, the ratio of accepted to attempted moves. The following four factors play important roles:

(a) λ is a quality factor. Smaller λ improves the quality of the solution, but it also increases the computation time.

(b) $\frac{1}{\sigma(S_k)}$ measures the distance of the system from quasi-equilibrium.

(c) $\left(\frac{1}{S_k^2\sigma^2(S_k)}\right)$ is the inverse of the statistical specific heat which depends on the variance.

(d) $\left(\frac{4\rho_o(S_k)(1-\rho_o(S_k))^2}{(2-\rho_o(S_k))^2}\right)$ is equal to $\rho_2/2$ where ρ_2 is the variance of the average energy change during a move. This is a measure of how effectively the state space is sampled and was found to be at a maximum value when $\rho_o \approx 0.44$

Azizi and Zolfahgari (2004) used an adaptive annealing schedule that adjusts the temperature dynamically based on the profile of the search path. Such adjustments could be in any direction including the possibility of reheating. In their proposed method, an adaptive temperature control scheme was used to change temperature based on the number of consecutive improving moves. In the second method, a tabulist was added to the adaptive simulated annealing algorithm in order to avoid revisits to the solutions.

Triki et al. (2004) studied annealing schedules. They performed experiments to construct an optimum annealing schedule that showed that there was no clearly better annealing schedule than the logarithmic schedule to ensure convergence towards the set of optima with probability one. They developed software to calculate a practical and optimum annealing schedule for a given objective function. They also conducted experiments on adaptive annealing schedules to compare classical annealing schedules. They proposed the following cooling schedules:

$$T_{i+1} = T_i \exp\left(-\frac{\lambda T_i}{\sigma(T_i)}\right) \tag{9}$$

$$T_{i+1} = T_i\left(1 - T_i\frac{\Delta(T_i)}{\sigma^2(T_i)}\right) \tag{10}$$

They showed that several classical adaptive temperature decrement rules proposed in the literature, which have different theoretical foundations and different mathematical equations, were in fact the same practical rule. They calculated a new adaptive decrement rule for controlling and tuning the SA algorithm.

Other cooling schedules make a more direct appeal to the theoretical results on asymptotic convergence. Lundy and Mees (1986) proposed an annealing schedule where there is only a single iteration at each temperature. They used heuristic arguments to derive a temperature function of the form

$$T_{i+1} = \frac{T_i}{1 + BT_i} \tag{11}$$

where B is a constant. Equation (11) is equivalent to

$$T_i = \frac{C_1}{1 + iC_2} \tag{12}$$

where C_1 and C_2 are constants. SA proposed by Connolly (1987, 1988) suggested that the majority of the iterations should be conducted at a suitably fixed temperature.

The choice of decreasing the temperature is an important issue as there has been a conflict, since the early days of SA, between theory and practice. There is no universally valid conclusion in the literature. However, a general choice is to cool the system slowly at the stage where the objective function is rapidly improving.

Number of Iterations

The number of iterations at each temperature is chosen so that the system is sufficiently close to the stationary distribution at that temperature. Aarts and Korst (1989) and van Laarhoven and Aarts (1987) refered this as quasi-equilibrium. An enough number of iterations at each temperature should be performed if the temperature is decreased periodically. If a smaller number of iterations is performed, all represented states will not be searched and the solution will not be able to reach the global optimum. The value of the number of iterations depends on the nature of the problem and its complexity.

Stopping Criterion

Various stopping criteria have been employed with SA-based algorithms. First, prefix a total number of iterations and number of iterations to move at each temperature. This criterion leads to a higher computation time without much update in f sometimes leading to a local optimal due to a less number of iterations. The biggest issue with this method is the prerequisite of the setting the number of iteration, which may not be known beforehand. Second, set a minimum value of the temperature and the number of iterations at each temperature. This idea is generated by the fact that the chance of improvement in a solution is rare once the temperature is close to zero. At a very low temperature, moves will be trapped in the neighborhood of the current solution. Third, set a number of iterations to move at each temperature and a predefined number of iterations to get a better solution. SA-based algorithms are capable of solving single objective and multi-objective optimization problems where a desired global optimal is hidden among many local optima. These methods have attractive and unique features when compared with other optimization techniques. First, a solution does not get trapped in a local minimum or maximum by sometimes accepting even the worse move. Second, configuration decisions proceed in a logical manner in SA. As a result, SA-based algorithms have been popular and their applications have been expanded. However, the search for an efficient algorithm based on SA is still continuing.

See

- ▶ Artificial Intelligence
- ▶ Evolutionary Algorithms
- ▶ Genetic Algorithms
- ▶ Global Optimization
- ▶ Heuristics
- ▶ Integer and Combinatorial Optimization
- ▶ Optimization
- ▶ Pareto-optimal Solution

References

Aarts, E. H. L., & Korst, J. H. M. (1989). *Simulated annealing and Boltzmann machines*. Chichester, UK: Wiley.

Aarts, E. H. L., & van Laarhoven, P. J. M. (1985). Statistical cooling: A general approach to combinatorial optimization problems. *Philips Journal of Research, 40*, 193–226.

Azencott, R. (1992), "Sequential simulated annealing: Speed of convergence and acceleration techniques" in: simulated annealing: Penalization techniques, Wiley, New York, 1992, 1.

Azizi, N., & Zolfaghari, S. (2004). Adaptive temperature control for simulated annealing: A comparative study. *Computers and Operations Research, 31*, 2439–2451.

Bell, D. A., McErlean, F. J., Stewart, P. M., & Mcclean, S. (1987). Application of simulated annealing to clustering tuples in database. *Journal of the American Society for Information Science, 41*, 98–110.

Cerny, V. (1985). Thermodynamics approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications, 45*, 41–51.

Chattopadhyay, A., & Seeley, C. E. (1994). A simulated annealing technique for multi-objective optimization of intelligent structures. *Smart Materials and Structures, 3*, 98–106.

Chen, S., & Luk, B. L. (1999). Adaptive simulated annealing for optimization in signal processing applications. *Signal Processing, 79*, 117–128.

Chen, J., Zhang, Y. F., & Nee, A. Y. C. (1988). Setup planning using Hopfield net and simulated annealing. *International Journal of Production Research, 36*, 981–1000.

Cho, J.-H., & Kim, Y.-D. (1997). A simulated annealing algorithm for resource constrained project scheduling problems. *Journal of the European Research Society, 48*, 736–744.

Chu, K. W., Deng, Y., & Reinitz, J. (1996). Parallel simulated annealing by mixing of states. *Journal of Computational Physics, 148*(2), 646–662.

Collins, N. E., Eglese, R. W., & Golden, B. L. (1988). Simulated annealing- an annotated bibliography. *American Journal of Mathematical and Management Science, 8*, 209–307.

Connolly, D. T. (1987). "Combinatorial optimization using simulated annealing", report, London School of Economics, London, WC2A 2AE, presented at the Martin Beale Memorial Symposium, London, July, 1987.

Connolly, D. T. (1988). An improved annealing scheme for the QAP. *European Journal of Operational Research, 46*, 93–100.

Czyżak, P., & Jaszkiewicz, A. (1998). Pareto simulated annealing – a metaheuristic technique for multiple-objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis, 7*, 34–47.

Czyżak, P., Hapke, M., & Jaszkiewicz, A. (1994). "Application of the Pareto-simulated annealing to the multiple criteria shortest path problem". Technical Report, Politechnika Poznanska Instytut Informatyki, Poland.

Dekkers, A., & Aarts, E. (1991). Global optimization and simulated annealing. *Mathematical Programming, 50*, 367–393.

Eglese, R. W. (1990). Simulated annealing: A tool for operational research. *European Journal of Operational Research, 46*, 271–281.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 6*, 721.

Girard, T., Staraj, R., Cambiaggio, E., & Muller, F. (2001). A simulated annealing algorithm for planner or conformal antenna array synthesis with optimized polarization. *Microwave and Optical Technology Letters, 28*, 86–89.

Golenko-Ginzburg, D., & Sims, J. A. (1992). Using permutation spaces in job-shop scheduling. *Asia Pacific Journal of Operation Research, 9*, 183–193.

Gong, G., Liu, Y., & Qian, M. (2001). An adaptive simulated annealing algorithm. *Stochastic Processes and their Applications, 94*(1), 95–103.

Hanke, M., & Li, P. (2000). Simulated annealing for the optimization of batch distillation process. *Computers and Chemical Engineering, 24*, 1–8.

Ingber, L. (1989). Very fast simulated annealing. *Mathematical Computing Modeling, 12*, 967.

Kim, J. U., Kim, Y. D., & Shim, S. O. (2002). Heuristic algorithms for a multi-period multi-stop transportation planning problem. *Journal of the Operational Research Society, 53*, 1027–1037.

Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science, 220*, 671–680.

Kouvelis, P., & Chiang, W. (1992). A simulated annealing procedure for single row layout problems in flexible manufacturing systems. *International Journal of Production Research, 30*, 717–732.

Kumral, M. (2003). Application of chance-constrained programming based on multi-objective simulated annealing to solve a mineral blending problem. *Engineering Optimization, 35*, 661–673.

Lam, J., & Delosme, J. M. (1988a). "An efficient simulated annealing schedule: Derivation", Technical Report 8816, Electrical Engineering Department, Yale, New Havan, CT, September.

Lam, J., & Delosme, J. M. (1988b). "An efficient simulated annealing schedule: Implementation and Evaluation", Technical Report 8817, Electrical Engineering Department, Yale, New Havan, CT, September.

Liu, H. C., & Huang, J. S. (1998). Pattern recognition using evolution algorithms with fast simulated annealing. *Pattern Recognition Letters, 19*, 403–413.

Lundy, M., & Mees, A. (1986). Convergence of an annealing algorithm. *Mathematical Programming, 34*, 111–124.

Maffioli, F. (1987). Randomized heuristic for NP-hard problem. In G. Andreatta, F. Mason, & P. Serafini (Eds.), *Advanced school on stochastic in combinatorial optimization* (pp. 760–793). Singapore: World Scientific.

McCormick, G., & Powell, R. S. (2004). Derivation of near-optimal pump schedules for water distribution by simulated annealing. *Journal of the Operational Research Society, 55*, 728–736.

Meller, R. D., & Bozer, Y. A. (1996). A new simulated annealing algorithm facility layout problem. *International Journal of Production Research, 34*, 1675.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., & Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics, 21*, 1087–1092.

Mukhopadhyay, S. K., Singh, M. K., & Srivastava, R. (1998). FMS machine loading: A simulated annealing approach. *International Journal of Production Research, 36*, 1529.

Otten, R. H. J. M., & van Ginneken, L. P. P. P. (1984). "Floorpan design using simulated annealing. In: Proceedings of the IEEE International Conference in Computer-Aided Design, Santa Clara. 96–98.

Reeves, C. R. (1993). *Modern heuristic techniques for combinatorial problems*. New York: John Wiley.

Rutenbar, R. A. (1989). Simulated annealing algorithms: An overview. *IEEE circuits and Devices Magazine, 5*, 1989.

Sait, S. M., & Youssef, H. (1999). "Iterative computer algorithms with applications in engineering". Press of IEEE Computer Society.

Serafini, P. (1985). *Mathematics of multi-objective optimization, CISM courses and lectures* (Vol. 289). Berlin: Springer, Verlag.

Serafini, P. (1992). "Simulated annealing for multiple objective optimization problems", In: Proceedings of the Tenth International Conference on Multiple Criteria Decision Making, Taipei 19–24.07, 1, 87–96.

Serafini, P. (1994). "Simulated annealing for multiple objective optimization problems" In: G. H. Tzeng, H. F. Wang, V.P. Wen, & P. L. Yu (Eds.), Multiple criteria decision making. Expand and enrich the domains of thinking and application, (pp. 283–292). Springer Verlag.

Shutler, P. M. E. (2003). A priority list based heuristic for the job shop problem. *Journal of the Operational Research Society, 54*, 571–584.

Sridhar, J., & Rajendran, C. (1993). Scheduling in a cellular manufacturing system: A simulated annealing approach. *International Journal of Production Research, 31*, 2927.

Starink, J. P. P., & Barker, E. (1995). Finding point correspondences using simulated annealing. *Pattern Recognition, 28*, 231–240.

Suman, B. (2002). Multi-objective simulated annealing–a metaheuristic technique for multi-objective optimization of a constrained problem. *Foundations of Computing and Decision Sciences, 27*, 171.

Suman, B. (2003). Simulated annealing based multi-objective algorithm and their application for system reliability. *Engineering Optimization, 35*, 391.

Suman, B. (2004). Study of simulated annealing based multi-objective algorithm for multi-objective optimization of a constrained problem. *Computers and Chemical Engineering, 28*, 1849.

Suman, B. (2005). Self-stopping PDMOSA and performance measure in simulated annealing based multi-objective optimization algorithms. *Computers and Chemical Engineering, 29*, 1131–1147.

Suman, B., Hoda, N., & Jha, S. (2010). Novel simulated annealing for multi objective optimization. *Computers & Chemical Engineering, 34*, 1618–1631.

Suman, B., & Kumar, P. (2006). A survey of simulated annealing as a tool for single and multiobjective optimization. *Journal of the Operational Research Society, 57*, 1143–1160.

Suppapitnarm, A. & Parks, T. (1999). "Simulated annealing: An alternative approach to true multi-objective optimization" In: Genetic and Evolutionary Computation Conference, Orlando, Florida.

Suresh, G., & Sahu, S. (1994). Stochastic assembly line balancing using simulated annealing. *International Journal of Production Research, 32*, 1801.

Swarnkar, R., & Tiwari, M. K. (2004). Modeling machine loading problem of FMSs and its solution methodology

using a hybrid tabu search and simulated annealing-based heuristic approach. *Robotics and Computer-Integrated Manufacturing, 20*, 199–209.
Szu, H., & Hartley, R. (1987). Fast simulated annealing. *Physics Letter A, 122*, 157.
Terzi, E., Vikiali, A., & Angelis, L. (2004). A simulated annealing approach for multimedia data placement. *Journal of Systems and Software, 73*(3), 467–480, to appear.
Tiwari, M. K., & Roy, D. (2003). Solving a part classification problem using simulated annealing-like hybrid algorithm. *Robotics and Computer-Integrated manufacturing, 19*, 415–424.
Triki, E., Collette, Y., & Siarry, P. (2004). A theoretical study on the behavior of simulated annealing leading to a new cooling schedule. *European Journal of Operational Research, 166*, 77–92.
Tuyttens, D., Teghem, J., Fortemps, P. H., & Nieuwenhuyze, K. V. (2000). Performance of the MOSA method for the bicriteria assignment problem. *Journal of Heuristics, 6*, 295.
Ulungu, L. E., & Teghem, J. (1994). Multi-objective combinatorial optimization problems: A survey. *Journal of Multicriteria Decision Analysis, 3*, 83–104.
Ulungu, L. E., Teghem, J., & Ost, C. (1998). Interactive simulated annealing in a multi-objective framework: Application to an industrial problem. *Journal of the Operational Research Society, 49*, 1044–1050.
Ulungu, L. E., Teghem, J., Fortemps, P. H., & Tuyttens, D. (1999). MOSA method: A tool for solving multi-objective combinatorial optimization problems. *Journal of Multicriteria Decision Analysis, 8*, 221–236.
Van Laahoven, P. J. M., & Aarts, E. H. L. (1987). *Simulated annealing: Theory and practice*. Dordrecht: Kluwer Academic Publishers.
van Laarhoven P. J. M., & Aarts, E. H. L. (1987). "Simulated annealing: Theory and application", Dordrecht: Reidel.
Yip, P. P. C., & Pao, Y. H. (1995). Combinatorial optimization with use of guided evolutionary simulated annealing. *IEEE Transactions, 6*, 290–295.

Simulation Metamodeling

Linda Weiser Friedman
Baruch College, City University of New York, New York, NY, USA

Introduction

Simulation experiments may be conducted for many reasons, for example, optimization. Additionally, and not incidentally, an objective of any system simulation must be to achieve a certain measure of understanding of the nature of the relationships between the input variables and the output variables of the real system under study. The simulation model, although simpler than the real-world system, is still a very complex way of relating input to output. Sometimes, a simpler model may be used as an auxiliary to the simulation model in order to better understand the more complex model and to provide a framework for testing hypotheses about it. This auxiliary model is frequently referred to as a metamodel (Santos 2009; Cheng 2008; Santos and Santos 2007; Friedman 1996).

One simple metamodel favored by some simulation researchers, e.g., Kleijnen (1979), Kleijnen and Sargent (2000), Santos and Santos (2009), is the general linear model. For a univariate response experiment, this is

$$\mu = \beta_0 + \sum_{j=1}^{k} \beta_j x_j + \varepsilon.$$

When simulation-generated data are used to estimate the parameters of this first-order linear additive model, the resulting estimated metamodel is

$$y_i = b_0 + \sum_{j=1}^{k} b_j x_{ij} + e_i \, (i = 1, \ldots, n).$$

This general linear metamodel can provide additional information regarding the relative contribution of each input factor to a response variable of interest.

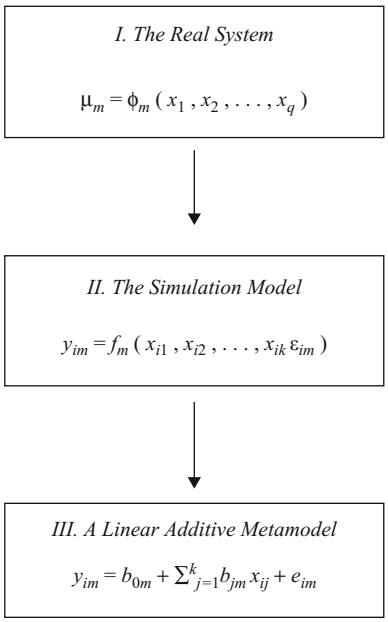
Most simulation experiments study more than one response variable, and so a multivariate metamodel (Friedman 1989) must necessarily be proposed. The multivariate general linear model

$$\mu_m = b_{0m} + \sum_{j=1}^{k} \beta_{jm} x_j + \varepsilon_m \, (m = 1, \ldots, p)$$

is estimated by

$$y_{im} = b_{0m} + \sum_{j=1}^{k} b_{jm} x_{ij} + e_{im} \, (i = 1, \ldots, n;\ m = 1, .., p).$$

Thus, the multivariate response metamodel is actually a series of regression-type equations, each representing the contributions of the criterion variables to the value of a response. This metamodel may be tested for significance via the multivariate



Simulation Metamodeling, Fig. 1 The simulation metamodel in context

general linear hypothesis (see, e.g., Hair et al. 2010), which was automated in Friedman and Friedman (1985a).

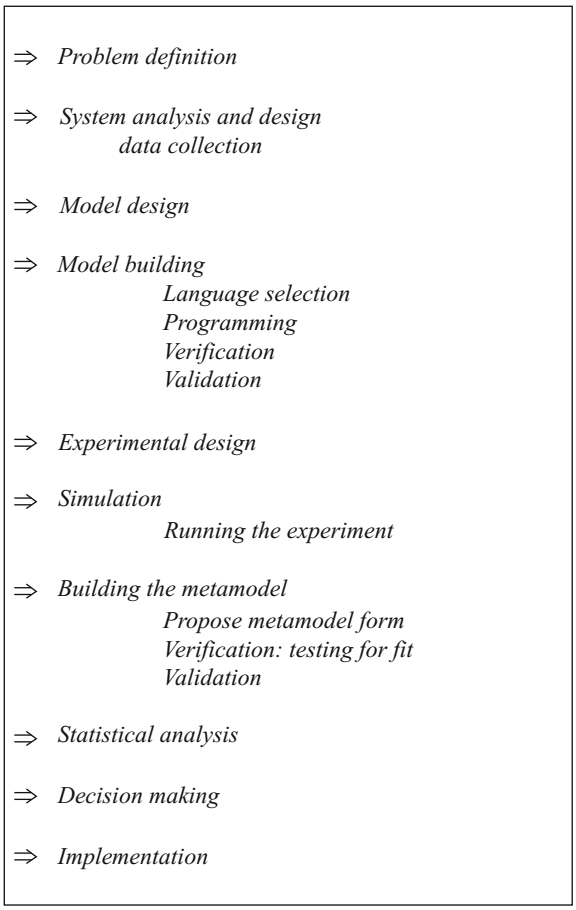
It can be shown that many multivariate statistical techniques as well as the univariate techniques of experimental design are specific cases of this general multivariate linear model. Thus, depending on the experimental layout, whether the factors are quantitative or qualitative, and the aim of the study, the general linear metamodel may be applied to regression analysis, analysis of variance, *t*-test, paired *t*-test, etc. In fact, whether a researcher explicitly says so or not, designing simulation experiments that will be analyzed via one of these statistical tests implies the use of a linear metamodel in one of its forms. The explicit use of a general linear metamodel enables one to interpret the simulated system more easily and more fully supporting model simplification; the enhanced exploration, optimization, and interpretation of the model; generalization to models of other systems of the same type; sensitivity analysis; etc.

Figure 1 is a pictorial representation of the three levels of explanation of the dynamics of a system simulation study. At the first level, the real system itself is unapproachable by the researcher, who can never hope to understand it completely. The system analysis and data collection functions take place here. At the second level, although the simulation model is leaner and more streamlined than the real system, it does attempt to replicate the real system at least with regard to the variables that are important to the goals of the researcher. The simulation model building, verification, and validation functions take place here. Finally, the analytic metamodel is at the leanest and most streamlined level. It attempts to approximate and aid in the interpretation of the simulation model and ultimately, of the real system itself. The experimental design and analysis function takes place here, and the multivariate general linear metamodel can often be used as a generalization of the various types of analyses performed on simulation output data.

Figure 2 displays the steps in a typical simulation study, with metamodeling included. During the metamodel construction phase, information uncovered during system analysis is used to propose one or more possible metamodel forms; use the simulation-generated data to fit the model, providing estimates of the parameters of the proposed metamodel; verify this metamodel by applying a statistical test for fit; and, validate the metamodel in the same manner that the simulation model was validated, for example, by comparing it to actual data from the system under study or a similar system. Overviews on the use of simulation metamodeling may be found in Madu and Kuei (1994), Barton (1994), Friedman (1996) and, more recently, Kuei et al. (2008), Iooss (2009), Santos and Santos (2009), and Zobel and Keeling (2008).

A Metamodeling Example

As an illustration, a multivariate general linear metamodel for the *M*/*M*/*s* queueing system is built and validated against known theoretical results (Friedman 1989). For this example, a simulation program was developed using SIMSCRIPT II.5 of the *M*/*M*/*s* queuing system with a single service facility and a single waiting line. The parameters λ, μ, and the number of identical service channels, *s*, were input as data to the simulation program, i.e., these were the factors in the simulation experiment. Twenty-five independent replications were generated for each of the six system variants displayed in Table 1. These system variants were selected judgmentally



Simulation Metamodeling, Fig. 2 Progress of a simulation study, metamodeling included

as being fairly congested, i.e., utilization factors ($\rho = \lambda/s\mu$) between 0.90 and 0.95. Three performance characteristics were output from each 15-week simulation run: average number of demands in the system, L; average system waiting time per demand, W; and average utilization per server, U.

The simulation model was validated by comparing the values of performance characteristics generated by the simulation program at the end of a 15-week run with the steady-state values expected using queueing theory. This comparison was done via the one-sample multivariate Hotelling T^2 test for the three measures of effectiveness. Validation results indicated that the simulation-generated estimates of the measures of performance were what one would expect as output from the *M*/*M*/*s* queuing system.

In developing the metamodel, the first impulse might be to fit a linear additive model with three

Simulation Metamodeling, Table 1 Experimental design

System	Arrival rate(λ)	Service rate (μ)	# Servers (s)
1	15	8	2
2	15	16	1
3	18	10	2
4	18	20	1
5	19	10	2
6	19	20	1

main effects (λ, μ, s) and some interaction effects. However, attempts to fit such a model resulted in a lack-of-fit test that showed that the model fit the data poorly. Poor fit of a linear regression model is a common problem in metamodels developed from queueing system simulation data since λ, μ, and s are actually intricately related in a nonlinear fashion.

The functional relationship next hypothesized for the *M*/*M*/*s* simulation metamodel was designed to take advantage of the fact that an important determinant in the behavior of queuing systems is the utilization factor, $\rho = \lambda/s\,\mu$. Thus, the proposed metamodel was

$$MOE_{im} = \alpha_m \frac{\lambda_i^{\beta_{1m}}}{\mu_i^{\beta_{2m}} s_i^{\beta_{3m}}} \eta_{im}$$
$$(i = 1, \ldots, n;\ m = 1, \ \ldots, p),$$

where i is the index for observations; m is the index for measurements and functions; MOE_m is the mth measure of effectiveness; α_m is a constant multiplier for the mth equation; and η_{im} represents an error factor in the hypothesized function. While this proposed metamodel is neither linear nor additive, it is a form of the intrinsically linear multiplicative model, namely,

$$MOE_{im} = \alpha_m \lambda_I^{\beta_{1m}}\ \mu_i^{\beta_{2m}}\ s_i^{\beta_{3m}}\ \eta_{im}$$

which may be transformed to a linear model by a logarithmic transformation

$$\ln MOE_{im} = \ln \alpha_m + \beta_{1m} \ln \lambda_i - \beta_{2m} \ln \mu_i - \beta_{3m} \ln s_i + \eta_{im}.$$

That this is the familiar multivariate general linear (regression) model

$$Y_{im} = \beta_{0m} + \beta_{1m} X_{1i} + \beta_{2m} X_{2i} + \beta_{3m} X_{3i} + \varepsilon_{im}$$

New Variable	←	Old Variable
Y_1	=	$\ln L$
Y_2	=	$\ln W$
Y_3	=	$\ln U$
β_{0m}	=	$\ln \alpha_m$
X_1	=	$\ln \lambda$
X_2	=	$\ln \mu$
X_3	=	$\ln s$
ε_m	=	$\ln \eta_m$

Simulation Metamodeling, Fig. 3 Change of variables for the regression metamodel

Simulation Metamodeling, Table 2 Multivariate regression metamodel estimates

	Y_1 (ln L)	Y_2 (ln W)	Y_1 (ln U)
Estimated regression coefficients			
$\boldsymbol{b}_0$	3.279	3.286	0.008*
$\boldsymbol{b}_1$	12.979	11.973	1.000
$\boldsymbol{b}_2$	−12.877	−12.874	−1.003
$\boldsymbol{b}_3$	−12.826	−12.823	−1.003
Standard errors of the coefficients			
$\boldsymbol{b}_0$	0.277	0.270	0.015
$\boldsymbol{b}_1$	0.445	0.433	0.024
$\boldsymbol{b}_2$	0.428	0.417	0.023
$\boldsymbol{b}_3$	0.429	0.418	0.023

*not significantly different from zero

is made clear by the change of variables listed in Fig. 3. The least-squares estimates for the vectors β_0, β_1, β_2, β_3 are, respectively, the vectors $\mathbf{b}_0$, $\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_3$. Applying this multivariate regression equation to the *M/M/s* simulation data produced the regression estimates presented in Table 2 along with their standard errors.

Metamodel Verification

As can be seen from Table 3, the multivariate linear regression model specified is indeed a significant effect in explaining the responses, as are the individual treatment effects. Further model exploration was not necessary, as the test for overall lack of fit was not significant, and the metamodel was accepted. (For further discussion of the Wilks' Λ statistic, and the

Simulation Metamodeling, Table 3 Tests of multivariate hypotheses

Source	Wilks'Λ	F	d.f.	p
Model	0.0001	9885.35	4,294	<0.001
X_1 (ln λ)	0.0519	876.26	3,144	<0.001
X_2 (ln μ)	0.0356	1299.22	3,144	<0.001
X_3 (ln s)	0.0358	1292.60	3,144	<0.001
Lack-of-fit	0.9951	0.12	6,284	>0.990

F-test derived from it, see Hair et al. 2010). Table 4 displays the results of the regression analysis on each response variable individually, laid out in the familiar analysis of variance table.

In this case, estimation of the metamodel parameters by (say) linear regression analysis is not an end in itself. Inserting the regression estimates of the metamodel coefficients into the general linear model, and then taking antilogarithms of both sides of the set of regression equations, leads back to the original (predictive) functional relationships, i.e., the simulation metamodel:

$$L = e^{3.279} \frac{\lambda^{12.979}}{\mu^{12.877} s^{12.826}}$$

$$W = e^{3.286} \frac{\lambda^{11.973}}{\mu^{12.871} s^{12.823}}$$

$$U = \frac{\lambda}{\mu^{1.003} s^{1.003}}$$

Metamodel Validation

Once a metamodel is developed, there is naturally a great temptation to simply use it as is, but it is at least as important to validate the simulation metamodel as validating the simulation model. After all, the metamodel is two steps removed from the real-world system under study. Once built and verified to determine that the metamodel fits the data with which it was developed, the metamodel should then be tested for two types of validity: Internal validity reflects the degree to which the metamodel accurately approximates the simulation model; external validity reflects the degree to which the metamodel accurately approximates the real-world system.

Simulation Metamodeling, Table 4 Regression analysis table by response variable

Source ***Response***:	d.f. Y_1	Sum of squares (ln L)	Mean square	F	p	R^2
Model	3	13.173	4.391	303.12	<0.001	0.862
Residual	146	2.115	0.014			
Lack of fit	2	0.004	0.002	0.14	>0.860	
Pure Error	144	2.111	0.015			
Total	149	15.288				
Response:	Y_2	(ln W)				
Model	3	14.992	4.997	364.11	<0.001	0.882
Residual	146	2.004	0.014			
Lack of fit	2	0.004	0.002	0.15	>0.850	
Pure error	144	2.000	0.014			
Total	149	16.996				
Response:	Y_3	(ln U)				
Model	3	0.080	0.027	613.66	<0.001	0.927
Residual	146	0.006	0.000			
Lack of fit	2	0.000	0.000	0.00	>0.999	
Pure error	144	0.006	0.000			
Total	149	0.086				

Internal Validity — Regression analysis, used to develop the general linear metamodel, is very much a data-based technique in that it finds the model with the best possible fit to the data. Frequently, models built in this manner fail to perform as well on new data. Several appropriate, practical statistical validation techniques for the general linear simulation metamodel have been examined (see, e.g., Friedman and Friedman 1985b; Panis et al. 1994; Santos and Porta Nova 2007; Iooss 2009; Kleijnen and Sargent 2000).

In the cross-validation technique, the regression metamodel is developed using only a portion, say, two-thirds of the observations, selected randomly. The regression metamodel is then tested against the remaining third, the holdout group, to see how well this equation, developed on one set of data, explains the responses in the new data. In this procedure, the held-out data is used to predict a whole new set of values of the response variable, which is then used together with the "true" responses (the values in the simulation-generated data of the holdout sample) to find the coefficient of determination, R^2. When compared to the original R^2 in the first set of data, one can see how much deterioration there was from the original data used to develop the model to the new, fresh data of the holdout sample, for example, a very low R^2 for the unselected cases would indicate that the model lacks predictive validity since it does not sufficiently explain the variation in the new, held out data. In validating the *M*/*M*/*s* metamodel, the value of the R^2 statistic for the original set of data was .70 and the R^2 for the holdout sample was .80. This indicated that the metamodel developed does indeed have predictive validity.

Further insight into how well the model predicts may be obtained by examining the residuals — computed by taking the actual observations minus the values predicted by the metamodel equation — of the holdout sample. The mean absolute percentage error, whereabsolute percentage error is calculated as $100\% \times |Residual|/y$ was found to be 5.6%, indicating good predictive validity on the part of the simulation metamodel with respect to the simulation model.

External Validity: Methods used in testing a simulation metamodel for external validity are equivalent, and sometimes identical, to those used to validate the simulation model, for example, face validity or expert judgment. Just as simulation responses have been compared with historical data from the real (or similar) system, so metamodel responses may also be compared with historical observations from the real (or similar) system. For this simulation metamodel to be valid, it should be a useful approximation not merely to the simulation model, from which the data used in building the metamodel was drawn, but also to the real-world system, to which any inferences and conclusions will

Simulation Metamodeling, Table 5 Metamodel validation to the theoretical system

			Analytic			Metamodel		
λ	μ	*s*	*L*	*W*	*U*	*L*	*W*	*U*
16	8.5	2	16.485	1.030	0.941	16.718	1.044	0.933
16	17	1	16.000	1.000	0.941	16.137	1.007	0.933
16	9	2	8.471	0.529	0.889	8.008	0.500	0.881
16	18	1	8.000	0.500	0.889	7.730	0.483	0.881
17	9	2	17.486	1.029	0.944	17.590	1.033	0.936
17	18	1	17.000	1.000	0.944	16.979	0.997	0.936
18	9.5	2	18.486	1.027	0.947	18.411	1.021	0.939
18	19	1	18.000	1.000	0.947	17.771	0.986	0.939
				Avg. error:		1.69%	1.70%	0.81%

be applied. Since the *M*/*M*/*s* system has been widely studied, it can be used to test the metamodel developed here for validity to the real system which the simulation models. This would be equivalent to taking actual (historical) data from the real system, or a similar equivalent one, and validating the simulation model and the metamodel with portions of that data.

Towards this end, several additional *M*/*M*/*s* system variants were selected which were different from the six used to develop the metamodel yet still within the experimental space. These new system configurations also had fairly high utilization factors. Values computed for *L*, *W*, and *U* using the multivariate metamodel were compared with the actual steady-state values for these system measures of effectiveness. The average absolute error, used to measure metamodel validity, is obtained by means of the ratio $100\% \times |Metamodel - Analytic|/Analytic$. Given the results of Table 5, the multivariate metamodel is taken to have performed well in representing the *M*/*M*/*s* queueing system in the range of system configurations studied.

Using the Simulation Metamodel

It is obvious from the patterns evident in the estimated coefficients in Table 2 that the multivariate metamodel may serve as more than a predictive functional model relating dependent variables with the independent variables. When such patterns appear, they urge the researcher to examine the metamodel further for relationships that are not immediately obvious. In simplified form, the multivariate metamodel may be (after suitable testing of estimated coefficients) represented as:

$$L = e^{3.28}\left(\frac{\lambda}{\mu s}\right)^{12.85}$$

$$W = \frac{e^{3.26}}{\lambda}\left(\frac{\lambda}{\mu s}\right)^{12.85}$$

$$U = \frac{\lambda}{\mu s}$$

It turns out that the formula for *U* is, of course, simply a restatement of the analytic formula for utilization, ρ. Additionally, from the metamodel formulas for *L* and *W*, it can be concluded that $L = \lambda W$, which is (not coincidentally) the well-known relationship first demonstrated by Little. An example of another sort of simplified-form relationship derived from this simplified metamodel that might prove useful in a study of this nature is $L = e^{3.28}\, U^{12.85}$.

Thus, in addition to providing a vehicle for prediction, the multivariate metamodel may also be expected to provide a means of exploring relationships inherent in the real system and in the simulation model of the real system, but otherwise masked by the complexity of the system studied.

See

- Response Surface Methodology
- Simulation of Stochastic Discrete-Event Systems
- Simulation Optimization
- Verification, Validation, and Testing of Models

References

Barton, R. R. (1994). Metamodeling: A state of the art review. *Proceedings of the Winter Simulation Conference*, 237–244.

Cheng, R. (2008). Selecting the best linear simulation metamodel. In S. J. Mason, R. R. Hill, L. Monch, O. Rose, T. Jefferson, & J. W. Fowler (Eds.), *Proceedings of the 2008 Winter Simulation Conference*, *Austin*, *TX*, 371–378.

dos Santos, M. I. R. (2009). Nonlinear regression metamodels: A systematic approach. *International Journal of Simulation and Process Modelling, 5*(3), 241–255.

dos Santos, M. I. R., & Porta Nova, A. M. O. (2007). Estimating and validating nonlinear regression metamodels in simulation. *Communications in Statistics - Simulation and Computation, 36*(1), 123–137.

dos Santos, P. M. R., & dos Santos, M. I. R. (2009). Using subsystem linear regression metamodels in stochastic simulation. *European Journal of Operational Research, 196*, 1031–1040.
Friedman, L. W. (1989). The multivariate metamodel in queuing system simulation. *Computers and Industrial Engineering, 16*, 329–337.
Friedman, L. W. (1996). *The simulation metamodel*. Norwell, MA: Kluwer Academic Press.
Friedman, L. W., & Friedman, H. H. (1985a). MULTIVREG: A SAS program. *Journal of Marketing Research (Computer Abstracts), 22*, 216–217.
Friedman, L. W., & Friedman, H. H. (1985b). Validating the simulation metamodel: Some practical approaches. *Simulation, 44*(September), 144–146.
Hair, J. F., Babin, B., & Anderson, R. (2010). *Multivariate data analysis*. NJ: Prentice Hall.
Iooss, B. (2009). Numerical study of the metamodel validation process. *First International Conference on Advances in System Simulation*, *SIMUL 2009*, September 20–25, 2009, Porto, Portugal, 100–105.
Kleijnen, J. P. C. (1979). Regression metamodels for generalizing simulation result. *IEEE Transactionas on Systems, Man, & Cybernetics, SMC, 9*(2), 93–96.
Kleijnen, J. P. C., & Sargent, R. G. (2000). A methodology for fitting and validating metamodels in simulation. *European Journal of Operational Research, 120*, 14–29.
Kuei, C.-H., Madu, C. N., & Winch, J. K. (2008). Supply chain quality management: A simulation study. *Information and Management Sciences, 19*(1), 131–151.
Madu, C. N., & Kuei, C. H. (1994). Regression metamodeling in computer simulation — The state of the art. *Simulation Practice and Theory, 2*, 27–41.
Panis, R. P., Myers, R. H., & Houck, E. C. (1994). Combining regression diagnostics with simulation metamodels. *European Journal of Operational Research, 73*, 85–94.
Santos, I. R., & Santos, P. R. (2007). Simulation metamodels for modeling output distribution parameters. In S. G. Henderson, G. Biller, M. H. Hsieh, J. Shortle, J. D. Tew, & R. R. Barton (Eds.), *Proceedings of the 2007 Winter Simulation Conference*, 910–918.
Zobel, C. W., & Keeling, K. B. (2008). Neural network-based simulation metamodels for predicting probability distributions. *Computers and Industrial Engineering, 54*(4), 879–888.

Simulation of Stochastic Discrete-Event Systems

Michael C. Fu[1] and Donald Gross[2]
[1]University of Maryland, College Park, MD, USA
[2]George Mason University, Fairfax, VA, USA

Introduction

One of the most powerful modeling tools in the operations research analyst's toolbox is stochastic (or Monte Carlo) simulation, which provides the ability to study complex stochastic systems in great detail using a computer program. Simulation models complement analytical models that require many simplifying assumptions, and in many situations, simulation provides the only way to analyze a system. Stochastic discrete-event systems are systems whose state changes upon the occurrence of discrete events, usually at stochastic times (Cassandras and Lafortune 2010). For example, in a queueing system, the state of the system includes the queue lengths, which change at discrete points in time when arrivals or departures occur. Discrete-event systems can be contrasted with continuous-time, continuous-state systems whose state changes continuously over time, with dynamics usually driven by differential equations, e.g., the motion of particles in a fluid. Discrete-event systems are ubiquitous in the man-made world, and thus simulation is widely used for modeling, analysis, and decision making in manufacturing, logistics and transportation, telecommunications, military operations, computer networks, health care, emergency response, finance, business processes, and many other service sectors. The following presents an overview of the key points in discrete-event stochastic simulation. For more details, refer to textbooks such as Banks et al. (2010), Law and Kelton (2000), and Fishman (1978), and the handbook on simulation edited by Henderson and Nelson (2006).

Elements of a Simulation Model

A simulation model contains three major components: input generation, process/event/sample path construction, and statistical output analysis. The input to the simulation model requires generation of the appropriate input processes. For example, a manufacturing system is generally modeled as a network of queues with a variety of different interarrival time and service time distributions. Random variates from these different distributions must be generated so that realizations or sample paths of the system "in action" can be constructed. Once these distributions are chosen, input random variates are generated, from which the next-event mechanism performs the simulation by advancing time upon occurrences of scheduled events, updating the system

state (e.g., queue lengths, idle/busy status of servers), and keeping track of various statistics through counters (e.g., number of customers served, waiting times, cumulative queue lengths) in order to calculate appropriate output performance measures. Output analysis employs the appropriate statistical techniques required to make valid statements concerning system performance based on the output performance measures obtained from the simulation runs.

The following simple hypothetical example illustrates the three basic elements described above.

> A small manufacturer of specialty items has signed a contract with a very prestigious customer for twenty orders of its premiere product. Management is concerned with current capacity and wishes to analyze the situation using discrete-event simulation. The customer will place orders at random times and would like them filled as soon as possible. Orders are placed only at the beginning of a month and could come as frequently as two months apart or as infrequently as seven months apart, or anything in between, all with equal probability (assumed independent). Currently, the production capability for this product is such that orders are shipped only at the end of a month and order filling time is equally likely between one and six months, inclusive (again, assumed independent). Only one order at a time can be processed, so that if a second order comes in while one is being prepared, it must wait until the order ahead of it is completed. For this capability, management would like to get an idea of the average number of orders in the system, the average time an order spends in the system, the maximum time an order spends in the system and the percentage of time the system is idle. The date of the first order is known and the production line will be set up just in time to receive the first order. The production line will be taken down after the last (20th) order is completed.

For this simple example, an easy way to generate the random input data would be to use independent rolls of a fair die. To generate the interarrival times, simply roll the die 19 times and add one to each value to get the times between successive orders after the first one. For the service times, the value of the roll itself suffices, requiring a roll of the die 20 more times. Table 1 below gives a sample of using a fair die to generate the input data, and Table 2 presents the (abbreviated) simulation table constructed from Table 1's input data. This would be called one simulation replication.

Table 2 was constructed from the interarrival and service-time input data as follows. At clock time 0, the first order (transaction) comes into the system, has a service time of 1 (month) and is due to depart at clock time 1 (month). At clock time 1, the next arrival, order 2, is due in at clock $0 + 7 = 7$, and since no order is in the system, will depart at its arrival time plus service time, i.e., $7 + 3 = 10$. The clock is advanced to time 7, and the next arrival (order 3) scheduled at $7 + 2 = 9$. Since order 3 arrives before order 2 leaves, the clock is advanced to time 9, the arriving order 3 enters the queue and order 2 is still in service, but due to depart at time 10. Order 4 is due in at $9 + 6 = 15$. The clock is then advanced to time 10, when order 2 leaves the system, order 3 enters service and is scheduled to depart at $10 + 2 = 12$. Order 4 is next to arrive and it is due in at 15, so the clock advances to 12. The simulated next-event mechanism continues in this fashion until the 20th order is processed.

The data in Table 2 can be used obtain the queue wait time and total time in system for each order. For example, order 1 entered the system at time 0, went right into service and left at time 1, spending zero time in queue and one month in the system. Order 2 arrived at time 7, also went directly into processing and left at time 10, spending 3 months in the system. Order 3, however, arriving at time 9, had to enter the queue since order 2 was still in process when it arrived. It left the queue for processing at time 10 and exited

Simulation of Stochastic Discrete-Event Systems, Table 1 Input data

Time between orders:	–,7,2,6,7,6,7,2,5,4,5,3,2,6,2,4,2,6,5,5
Service times:	1,3,2,3,6,5,4,5,1,1,3,1,3,2,2,6,5,1,3,5

Simulation of Stochastic Discrete-Event Systems, Table 2 Key: [n],t = [order number], time of occurrence

Master clock time	Next events		Transaction in queue	Transaction in service
	Arrival	Departure		
0	[2],7	[1],1		→ [1]
1	[2],7	[2],10		[1] →
7	[3],9	[2],10		→ [2]
9	[4],15	[2],10	→ [3]	[2]
10	[4],15	[3],12	[3] → →[3]	[2] →
●			●	●
●			●	●
●			●	●
81	[20],86	[19],84		→[19]
84	[20],86			[19] →
86		[20],91		→[20]
91				[20] →

the system at time 12 (not shown in the abbreviated table), spending 1 month in queue waiting for processing and 3 months total time in the system. Average waiting times and maximum waiting times can also be easily calculated, as well as average queue lengths and system utilization.

For the above example, the maximum number of orders in the queue was 1, the maximum number of orders in the system was 2, the maximum time an order spent in the system waiting to be processed was 4 months (order number 17), and the maximum time an order spent in the system was 9 months (also order number 17). The average queue length was 0.13, the average number in system was 0.81, the percent of the time the system was empty and idle was 32%, and the average waiting times in queue and system were 0.6 and 3.7 months, respectively.

Input Distribution Selection and Random Variate Generation

Keeping in mind the old acronym, GIGO (Garbage In, Garbage Out), care must be given to choosing the distributions that best describe the environment being modeled. This involves knowing as much about the modeling environment as possible and valid data analyses. There are some cases where data are not available nor can they be collected (e.g., design of a new system) and for these, only domain knowledge can be used. Sensitivity of output performance measures to specific input distributions is still an area of active research; some discussion is provided later.

Once the appropriate distributions are chosen (discussed in the entry on distribution selection for stochastic modeling), it is necessary to be able to generate representative samples from these distributions for running the simulation. Much study has been done in this area, as discussed in Banks et al. (2010), Fishman (1978), Law and Kelton (2000).

The basis for generating random variates from a desired probability distribution lies in being able to generate random numbers U_1, U_2, U_3,... which are independent and identically distributed (i.i.d.) on the interval [0,1], written as $U(0,1)$, where $U(a,b)$ denotes the uniform distribution on $[a, b]$. This is generally done via a pseudorandom number generator, which uses a mathematical recursion to generate a sequence of integers that statistically look as if they are random. These integers can then be normalized to the interval [0,1]. Perhaps the simplest generator is based on a linear congruence equation of the form

$$Y_i = (aY_{i-1} + c) \bmod m,$$

where a is called the constant multiplier, c is the increment, and m is the modulus in modulo arithmetic (the quantity in the parentheses is divided by m and only the remainder kept). Since the relation is recursive, a starting point called the seed, Y_0, is required. The numbers generated from this recursion will be in the interval $[0, m-1)$ and thus dividing by m normalizes the values to [0,1). When simulation was first used, these were the most reliable random number generators, but current practice employs much more sophisticated generators that are described in the random number generation entry; see also Chapters 3 and 6 of Henderson and Nelson (2006).

Using i.i.d. $U(0,1)$ random numbers, random variates from virtually any probability distribution (including empirical data) can be generated. One procedure for doing this is via the inverse (CDF) transform method. Given a random number $U \sim U(0,1)$ and the cumulative distribution function (CDF) F, the inverse transform method generates a random variate $X \sim F$ via

$$X = F^{-1}(U),$$

where F^{-1} denotes the inverse function (not $1/F$). The inversion algorithm can be viewed graphically as follows: find the U value on the y-axis, project horizontally to the CDF curve, and then project vertically down to the x-axis to read off the corresponding X value. As an example, to generate exponentially distributed random variates with mean θ, the CDF is given by $F(x) = 1 - e^{-x/\theta}$, so solving for X in $U = 1 - e^{-X/\theta}$ yields

$$X = -\theta \ln(1 - U).$$

Since (1- U) has the same distribution as U, the usual implementation is to use

$$X = -\theta \ln U,$$

since it saves one arithmetic operation. Another simple example is the algorithm for the general uniform

distribution, i.e., $X \sim U(a, b)$, which simply scales and shifts the random number:

$$X = a + (b - a)U.$$

The inverse transform method can be used for both discrete and continuous probability distributions, including empirical distributions. However, for most continuous distributions, the CDF F is not analytically invertible, e.g., the normal and most gamma distributions, though a simple numerical procedure can often be used. Other methods for generating random variates include acceptance-rejection, convolution, and composition. One advantage that the inverse transform has over other methods is that a single random number generates a single random variate, whereas in the other methods multiple random numbers may be needed to produce a single random variate; in the case of acceptance-rejection, the number of random numbers required is generally itself random. In addition to efficiency, the one-to-one correspondence can also be of great benefit when it comes to implementing variance reduction techniques, e.g., common random numbers. The entire volume of Devroye (1986), available online from the author's own Web site for free download, is devoted to input variate generation; see also Fishman (1996) and Chapters 4 and 5 of Henderson and Nelson (2006).

Simulation Programming Languages/ Modeling Software

The simulation modeler has a large variety of languages and packages from which to choose. These can be categorized into three main types: general-purpose languages, simulation languages, and simulation modeling software packages. General-purpose languages such as FORTRAN, BASIC, C, C++, Pascal, and Java allow the most flexibility in modeling but require the most effort to program. One can get a feel from the earlier very simple example of what might be involved in generating variates from the input probability distributions and programming the next-event routines and statistical calculations needed for obtaining output measures of performance. Numerical computing languages/environments such as MATLAB and R can also be used to program simulation models.

Simulation languages such as GPSS, SIMAN, SIMSCRIPT, and SLAM were developed to automatically include the components that stochastic discrete-event simulation models have in common, as illustrated by the simple example presented earlier, e.g., random variate generation, next-event logic execution, statistical counters, and output analyses. These simulation language packages make building a simulation model much easier, although some flexibility in modeling is sacrificed since the model must fit into the specific language environment being used. As a general rule, one can expect that the easier the programming becomes, the less flexibility there is in deviating from the language environment, although many software packages allow linkage to general purpose languages, thereby greatly increasing their modeling and analysis capabilities.

Even easier to use than simulation languages are simulation modeling software packages, the successors to what used to be called simulators. These are completely self-contained and require very little, if any, programming, as the model is generally built in a graphical user interface by choosing among icons in pull-down menus or from toolbars, and the software automatically includes animation capabilities that will allow the user to observe the system evolving over time. Well-known simulation software packages include Arena, SIMPROCESS, ProModel, WITNESS, Simio, SIMUL8, and AnyLogic. Examples of software packages tailored to a particular application area include AutoMod and SIMFACTORY.

Falling somewhere between self-contained simulation software packages and simulation languages are add-in packages that enable simulation modeling and analysis. Examples include SimEvents in MATLAB and SAS Simulation Studio for JMP. In addition, for stochastic (Monte Carlo, generally static) spreadsheet simulation, two of the most common add-in software packages are Crystal Ball and @RISK.

Output Analysis

Making valid conclusions from simulation output requires sound experimental design and statistical analysis. This section presents some basic procedures for analyzing simulation output.

There are two major types of simulation models: terminating and non-terminating. A terminating model

has a natural start and stop time, e.g., a bank opens its doors at 9:00 am and closes its doors at 3:00 pm. On the other hand, a non-terminating model does not have a start and stop time, e.g., a semiconductor manufacturing fabrication facility that essentially runs continually. In non-terminating simulations, steady-state results are usually of interest, and in simulating such a system, a determination must be made as to when the initial transients have dampened out and the simulation is in steady state.

For terminating simulations, ***Once is not enough***! In other words, a single replication does not provide enough information to make any statistical statements. For example, the maximum waiting time is a single observation, that is, a sample size of one. Multiple independent replications (repeated runs) of the experiment are required. For the example presented earlier, different random number streams (e.g., two dice rather than one) are used for the order arrival and processing times for each replication, which generates a sample of independent output observations to which classical statistics can be applied. Thus, n replications would generate n values for the maximum waiting times, say, $w_1, w_2, \ldots, w_n$. Assuming n is large enough to employ the central limit theorem, a $100(1-\alpha)\%$ confidence interval (CI) is formed by calculating the respective sample mean and sample standard deviation of the maximum waiting time by $\overline{w} = \frac{1}{n}\sum_{i=1}^{n} w_i$,

$$s_w = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(w_i - \overline{w})^2},$$

and then obtaining the CI as

$$\overline{w} \pm t_{n-1,1-\alpha/2}\frac{s_w}{\sqrt{n}},$$

where $t_{n-1,1-\alpha/2}$ is the upper $1-\alpha/2$ critical value for the t-distribution with $n-1$ degrees of freedom.

Non-terminating simulations aimed at steady-state estimation face two major challenges: initialization bias and serial correlation. The former has to do with determining when steady state is reached, and the latter has to do with how to carry out the simulations for the purpose of forming CIs. The main approaches in non-terminating simulations are independent replications, batch means, and the regenerative method. The first two approaches are the most commonly used and must address both challenges. The regenerative method can actually eliminate both major challenges but faces its own unique challenges.

Assume for the moment that the first challenge is solved, and the simulation is run for n transactions after reaching steady state to obtain n values for the time a customer spends waiting in a particular queue for service, again denoted by w_i (now these are actual waits, not maximum waits). It might be tempting to calculate the average and standard deviation of these n values and proceed as above to form a CI. However, these w_i are generally positively correlated, so that using the above formula for s_w would be a poor (biased low) estimate for the true variance. This is one version of the serial correlation problem.

One way around the serial correlation problem is to follow the same procedure as in the terminating simulation setting, and run m independent replications, as in the terminating case. For each independent replication, again calculate the mean of the w_i, denoting the mean for the jth replication by

$$\overline{w}_j = \sum_{i=1}^{n} w_{ij}/n$$

where w_{ij} is the waiting time for transaction i in replication j, $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$. An approximate $100(1-\alpha)\%$ CI is given by

$$\overline{w} \pm t_{m-1,1-\alpha/2}\frac{s_{\overline{w}_j}}{\sqrt{m}}$$

where

$$\overline{w} = \frac{1}{m}\sum_{j=1}^{m}\overline{w}_j$$

and

$$s_{\overline{w}_j} = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(w_i - \overline{w})^2}.$$

Both the run length (n) and the number of replications (m) influence the size of the standard error used in forming the CI. The standard error goes down by the square root of m, so that the more replications made, the tighter (narrower) the CI. As the run length n is increased, the computed value

of $s_{\overline{w}_j}$ itself will be smaller for a given number of replications, so that longer run lengths will also increase the precision of the CI. Thus for a fixed amount of computer running time, there is a trade-off between the sizes of n and m. CI coverage for steady-state simulation was first addressed in Schruben (1980).

The other major challenge in non-terminating simulations is the determination of the warm-up period, the initial amount of simulation time required to bring the process near steady-state conditions and hence eliminate initialization bias. Data are not collected until after the end of the warm-up period. A variety of procedures have been developed and the reader is referred to the basic simulation texts referenced previously for more details and further references.

In the procedure just described, the warm-up period is wasted in each of the m independent replications. The method of batch means (Law 1977; Schmeiser 1982) avoids this by using one single long run that is broken up into m segments (batches) of n transactions each, so that only a single warm-up period is discarded rather than m as in the independent replications procedure. However, this procedure brings back the earlier problem, since there is serial correlation between segments. However, the performance measures for each segment become statistically less correlated as the length of each individual segment increases, since the distance between transactions in each segment is further apart. If the segment length is sufficiently large, then the methodology for determining CIs is identical to that for m independent replications. The trade-off here is between the length of the batches (n) and the number of segments (m). Assuming the product mn is kept constant, larger m means tighter CIs but increased serial correlation, because the length of each segment is small.

The regenerative method can eliminate both problems of initialization bias and serial correlation. The main challenges for the regenerative method include finding regenerative points, which can be difficult for large systems, and the length of regenerative cycles, which are stochastic and can be long for large systems. Since these cycles are used to form CIs, with each cycle essentially corresponding to a replication, the efficiency of the method can be drastically reduced when regenerative cycles are long. See the accompanying entry for more details on the regenerative method. Other methods for steady-state simulation include autoregressive methods, spectral analysis, and standardized time series and time series analyses, which are summarized in Law and Kelton (2000); see Chapter 8 in Henderson and Nelson (2006) for an in-depth presentation of statistical techniques for simulation output analysis, including quantile and density estimation.

In comparing two alternative system designs, the technique most commonly used is a paired-t CI on the difference of a given performance measure. The number of replications for both designs must be the same, and the mean and sample variance are computed in the same manner as before, after forming the pairwise differences on the performance measure. The difference in performance of the two designs is considered statistically significant if the resulting CI does not contain 0 (zero); if it does contain 0, then it is generally desirable to carry out more replications to tighten the CI to be able to detect a statistical difference. Also, in forming pairwise comparisons, whenever possible the same random number stream(s) should be used for each design *within* a replication, so that the difference observed depends only on the design parameter change and not on the variation due to the randomness of the random variates generated. Different random numbers streams are still used between the replications. This is a variance reduction technique called common random numbers (CRN) and can be quite effective in tightening CIs. Critical to the success of CRN is the notion of synchronization, which intuitively means that as much as possible the same random numbers are used for the parts of the two system designs that are similar (or identical).

Comparing more than two designs necessitates using multiple comparison techniques. The most straightforward method would be comparisons in a pairwise fashion using the methodology for comparing only two systems. However, making all pairwise CIs among k designs would require $k(k\text{-}1)/2$ comparisons, and if the confidence level of a single pairwise CI is $1-\alpha$, and there are N CIs, the confidence associated with a statement concerning all the pairs simultaneously drops to a lower bound of $1-N\alpha$ (Bonferroni inequality). So, for example, if there are 5 designs being compared in all possible pairs (10 in all), and an overall 95% confidence level is desired, then each CI level should be 99.5%.

Since all pairwise comparisons is not very efficient if the number of alternatives is more than a few, it is better to consider other statistical ranking and selection procedures, such as selecting the best of k systems (or multiple comparisons with the best), selecting a subset of size r containing the best of the k systems, and selecting the r best of k systems. This topic is treated in detail in the entry on statistical ranking and selection; see also Law and Kelton (2000).

Variance Reduction Techniques

Unlike sampling from the real world, the simulation modeler has control over the randomness generated in the system. Often, purposely introducing correlation among certain of the random variates in a simulation run can reduce variance and provide tighter CIs. One example of this was shown above in forming a paired-t CI for the difference between two systems by using common random numbers within a replication, which introduces positive correlation between the two performance measures within a replication, yielding a smaller variance for the mean difference.

Another technique, called antithetic variates, introduces negative correlation between two successive replications of a given design with the idea that a large random value in one of the pairs will be offset by a small random value in the other. The performance measures for the pairs are averaged to give a single observed performance measure. Hence, if m replications are run, there are $m/2$ independent values being averaged for the CI calculation, but the variance of these values should be lower than m independent observations. Caution must be exercised when combining antithetic variates with common random numbers (see Law and Kelton 2000).

Indirect estimation is another simple approach for reducing variance, using known relationships between quantities whose performance is being estimated by the simulation model. For example, the mean time spent in the system is the sum of the mean time spent in queue and the mean service time. Since the latter is known exactly, it should be clear that for estimating the mean system time, it is better – in terms of obtaining tighter CIs – to estimate the mean queue time and add this to the known mean service time rather than directly averaging the individual system times. As another example, Little's Law relates mean queue time to the mean queue length, whereas again both can be estimated from the simulation. It turns out it is better in this case to estimate mean queue times directly and use Little's Law to obtain an indirect estimate for mean queue length (see Law and Kelton 2000).

Other effective, albeit more complicated, variance reduction techniques include control variates, conditioning, importance sampling, stratified sampling, and splitting, all of which are described in the variance reduction techniques entry in this Encyclopedia.

Sensitivity Analysis and Optimization

All good modelers appreciate the importance of sensitivity analysis in testing the utility of their model. For example, in the simple example presented earlier, one might be interested in the sensitivity of the average waiting time in queue to the mean service time of the orders. In simulation, the most direct brute-force way of carrying out sensitivity analysis is to perturb the parameter of interest and perform another simulation at the perturbed value of the parameter; in other words, resimulation. Clearly if the number of parameters is large, this becomes very inefficient. As a result, much research has been carried out since the 1970s to find efficient ways of estimating sensitivities, called stochastic gradient estimation (see Chapter 19 in Henderson and Nelson 2006). The two most common approaches are perturbation analysis and the likelihood ratio/score function method, both of which attempt to provide estimators that can be computed on a *single* simulation replication, i.e., *without resimulation*; see the corresponding entries in this Encyclopedia for details. In addition to sensitivity analysis, stochastic gradient estimation can be used in conjunction with the simulation model to carry out optimization for an objective function based on output measures of performance from the simulation model. Methods for doing this are described in the entries on simulation optimization and stochastic approximation.

Model Verification and Validation

Model validation is an essential step in a simulation study. Prior to developing a simulation model,

it behooves the simulation analyst to become very familiar with the system being studied, to involve the managers and operating personnel of the system, and thus to agree on the level of detail required to achieve the goal of the study. The appropriate level of detail is always the coarsest that can still provide the answers required. One problem with simulation modeling is that since any level of detail can be modeled, models are often developed in more detail than necessary, which can be very inefficient and counterproductive.

Validity is closely associated with verification and credibility. Verification has to do with program debugging to make sure the computer program does what is intended. This is generally the most straightforward of the triumvirate to accomplish, as there are well-known and established methods for debugging computer programs. The animation capability in simulation software can be helpful in verification, since the user can observe directly if the evolution is progressing as intended.

Validation deals with how accurate a representation of reality the model provides, and credibility deals with how believable the model is to the users. To establish validity and credibility, users must be involved in the study early and often. Goals of the study, appropriate system performance measures, and level of detail must be agreed upon and kept as simple as possible. A log book of assumptions should be kept, updated frequently, and signed off periodically by the model builders and users. Animation of the system can again be of help in the process of establishing credibility, by convincing users that the simulation model adequately mimics the true system.

When possible, simulation model output should be checked against actual system performance, if the system being modeled is in operation. If the model can duplicate (in a statistical sense) actual data, both validity and credibility are advanced. The model can be run under a variety of conditions and results examined by the users for plausibility. Most simulation texts have at least one chapter devoted to this important topic; see also Gass and Thompson (1980), Sargent (2011).

Other Simulation

For operations research analysts, the most well-known continuous-time simulation paradigm is probably system dynamics. In addition to purely discrete-event or continuous-time models, many systems are best modeled with a state that contains both a discrete-event and continuous-time component, e.g., a flow system that could be in one of many discrete modes (such as up or down) or a queueing system where customers possess characteristics with continuous-valued variables changing over time, giving rise to hybrid system simulations. Many of the simulation software packages can handle these, as well, either directly or through a combination (e.g., Simulink with SimEvents in MATLAB). Another important modeling paradigm is agent-based simulation, where the focus is on the agents in the system rather than the processes or events; these models are widespread not only in the operations research/management science community but also in economics and the behavioral sciences. The latest state-of-the-art developments in stochastic discrete-event simulation and agent-based simulation are presented annually in December at the Winter Simulation Conference, which makes its proceedings freely available online at its Web site.

See

- ▶ Agent-Based Simulation
- ▶ Distribution Selection for Stochastic Modeling
- ▶ Little's Law
- ▶ Monte Carlo Methods
- ▶ Monte Carlo Simulation
- ▶ Networks of Queues
- ▶ Perturbation Analysis
- ▶ Queueing Theory
- ▶ Random Number Generators
- ▶ Rare Event Simulation
- ▶ Regenerative Simulation
- ▶ Response Surface Methodology
- ▶ Score Functions
- ▶ Simulation Optimization
- ▶ Statistical Ranking and Selection
- ▶ Stochastic Approximation
- ▶ Stochastic Input Model Selection
- ▶ System Dynamics
- ▶ Variance Reduction Techniques in Monte Carlo Methods
- ▶ Verification, Validation, and Testing of Models

References

Banks, J., Carson, J. S., Nelson, B. L., & Nichol, D. M. (2010). *Discrete-event system simulation* (5th ed.). Upper Saddle River, NJ: Prentice-Hall.

Cassandras, C. G., & Lafortune, S. (2010). *Introduction to discrete event systems* (2nd ed.). New York: Springer.

Devroye, L. (1986). *Non-uniform random variate generation*. New York: Springer. Also available freely online on the author's Web site (out of print).

Fishman, G. S. (1978). *Principles of discrete event simulation*. New York: Wiley.

Fishman, G. S. (1996). *Monte Carlo: Concepts, algorithms, and applications*. New York: Springer.

Gass, S. I., & Thompson, B. W. (1980). Guidelines for model evaluation: An abridged version of the U.S. general accounting office exposure draft. *Operations Research, 28*, 431–439.

Henderson, S. G., & Nelson, B. L. (Eds.). (2006). *Simulation, Handbook in operations research and management science* (Vol. 13). Amsterdam: North-Holland, Elsevier.

Law, A. M. (1977). Confidence intervals in discrete event simulation: A comparison of replication and batch means. *Naval Research Logistics Quarterly, 27*, 667–678.

Law, A. M., & Kelton, W. D. (2000). *Simulation modeling* (3rd ed.). New York: McGraw-Hill.

Sargent, R. G. (2011). Verification and validation of simulation models. In S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, & M. Fu (Eds.), *Proceedings of the 2011 winter simulation conference* (pp. 183–198). New York: ACM.

Schmeiser, B. W. (1982). Batch size effects in the analysis of simulation output. *Operations Research, 30*, 556–568.

Schruben, L. W. (1980). A coverage function for interval estimators of simulation response. *Management Science, 26*, 18–27.

Simulation Optimization

Michael C. Fu
University of Maryland, College Park, MD, USA

Introduction

Optimization in operations research and management science is generally identified with mathematical programming, where analytical expressions for quantities of interest (comprising the objective function and constraint functions) are assumed to be readily available and relatively easy to evaluate, so that the primary focus is on the *search* for the optimal solution(s). In simulation optimization, the objective function and/or constraints require expensive (in terms of computational effort) simulations to estimate or evaluate. In the stochastic setting considered here, multiple simulation runs (or replications) are used for estimating (through statistical sampling) system performance measures. The usual generic form of the optimization problem takes a form similar to that found in mathematical programming:

$$\min_{\theta \in \Theta} l(\theta), \tag{1}$$

where θ denotes the (vector of) controllable parameters or decision variables and Θ defines the constraint set on θ. Thus, each point $\theta \in \Theta$ represents one possible solution in the feasible solution space Θ. Assume throughout that the objective function is an expectation, i.e.,

$$l(\theta) = E[L(\theta, \omega)],$$

with ω representing a sample path (or simulation replication) and $L(\theta, \omega)$ the corresponding sample performance estimate. Objective functions of other forms (e.g., quantiles) can also be handled in a similar manner, and of course probabilities are simply expectations of indicator functions. In contrast to mathematical programming objective functions, the quantity $l(\theta)$ is not only expensive to evaluate, but it is generally quite nonlinear. Moreover, the constraint set Θ may itself also involve quantities that must be estimated from simulation, although the majority of existing techniques assume that they are available analytically.

As in mathematical programming, the set of solution techniques can be subdivided according to the state space of the controllable parameter θ: discrete or continuous. In addition, for the discrete case, there is a distinction between relatively small state spaces and larger (including infinite) state spaces; furthermore, the discrete state space may not possess a natural ordering/metric, e.g., the integers vs. purely categorical variables. As an example, consider a queueing network, where decision parameters might include the service rate at a station, which is generally continuously valued; the number of servers at a station, which is discrete valued and could be relatively small if considering stations individually or combinatorially large if considering the allocation of a fixed (large) number of servers among all stations in a

large network; and the queue discipline — first-come, first-served vs. various possible (static or dynamic) priority schemes, which involves a discrete categorical choice.

Small State Space

In this situation, the approach is analogous to enumeration in the deterministic setting, but it takes on a statistical flavor since there is randomness involved. Two (related) categories of statistical procedures are most applicable: ranking and selection, and multiple comparisons (Bechhofer et al. 1995). An applicable ranking-and-selection method will provide a sequential procedure that will select the minimizing θ among a finite set according to some statistical criterion such as a pre-specified confidence level. A number of these procedures pertinent to the simulation setting can be found in Law and Kelton (2000). Multiple-comparisons procedures, on the other hand, specify the use of certain pairwise comparisons to make inferences in the form of confidence intervals; they are not inherently sequential procedures. In terms of optimization, the most useful of these procedures are multiple comparisons with the best, which leads to $|\Theta|$ (size of state space) simultaneous confidence intervals. A review of these procedures in the context of simulation can be found in Kim and Nelson (2006). Other related approaches include optimal computing budget allocation (OCBA) and optimal learning, described in Powell and Ryzhov (2012) and Chen and Lee (2010), respectively; see also Branke et al. (2007) for a comparison of many of these procedures.

Large Discrete State Space

The main class of algorithms for this setting are random search methods, which iteratively update a single point by selecting the next point θ_{n+1} from a neighborhood of the present point θ_n. The resulting algorithms differ in specification of (i) neighborhood structure $N(\theta)$, and (ii) updating from θ_n to θ_{n+1}. Variants include the stochastic ruler, as well as those implementing simulated annealing. One recent algorithm using an interesting neighborhood structure is Convergent Optimization via Most-Promising-Area Stochastic Search (COMPASS) proposed in Hong and Nelson (2006). For a more detailed discussion on random search methods with references, see Andradóttir (2006) and Nelson (2010).

Other algorithms that can treat large discrete state spaces include genetic algorithms, estimation of distribution algorithms (Larrañaga and Lozano 2001), the cross-entropy method (Rubinstein and Kroese 2004), the nested partitions method (Shi and Ólafsson 2008), and model reference adaptive search (Hu et al. 2007). Another general approach is ordinal optimization (Ho et al. 1992), which is based on the idea that order converges (exponentially) faster than (statistical) estimation, a notion that can be made mathematically rigorous through large deviations theory. Massive parallel computation (e.g., simultaneously simulating a huge number of different alternatives) is especially suited for implementing this framework.

Continuous State Space

Most of the OR/MS research has focused on this case. Pattern search methods based on deterministic analogs directly adapted to the stochastic setting constitute one set of techniques. Examples include the Nelder-Mead method and its variants [2] where a simplex family of points is updated at each iteration according to some prescribed rules that control movements and possible expansion or contraction; and the Hookes-and-Jeeves method. Random search methods can also sometimes be adapted to the continuous parameter case.

Aside from the aforementioned methods, there are three major approaches to the continuous parameter problem. The first approach, stochastic approximation (SA), uses a stochastic version of gradient-based local improvement to iteratively update a single point. The second approach, sample average approximation (SAA) — also known as sample path optimization, the stochastic counterpart method, or retrospective optimization — uses multiple simulation replications to obtain a sufficiently precise estimate of the objective function (and constraints, if also noisy) in order to apply a deterministic optimization algorithm. The third approach, response surface methodology (RSM), uses simulation replications to fit a surface (e.g., the objective function) in either a global or local manner, on which optimization is performed

globally or sequentially. These approaches are now briefly discussed further in the context of simulation optimization; more details and further references on all three approaches can be found in the corresponding entries in this volume.

Stochastic Approximation

Stochastic approximation methods are the stochastic versions of gradient-based deterministic search algorithms. The basic underlying assumption of stochastic approximation when used in simulation optimization is that the original problem given by (1) can be solved by finding the zero of the gradient, i.e., by solving $\nabla l(\theta) = 0$, where ∇ denotes the gradient operator, which may only give a local optimum. The SA algorithm is of the following form:

$$\theta_{n+1} = \Pi_\Theta\left(\theta_n - a_n \widehat{\nabla} l(\theta_n)\right), \tag{2}$$

where θ_n is the parameter value at the beginning of iteration n, $\widehat{\nabla} l(\theta_n)$ is an estimate of $\nabla l(\theta_n)$ from iteration n, a_n is a (positive) sequence of step size multipliers, which shall henceforth be called the gain sequence, and Π_Θ is a projection onto Θ. When an unbiased estimator is used for $\widehat{\nabla} l(\theta_n)$, (2) is called a Robbins-Monro algorithm and when a finite difference estimate is used, it is called a Kiefer-Wolfowitz algorithm. Sometimes, the term Robbins-Monro-like algorithm is used for those procedures that estimate the gradient with some bias but without resorting to finite differences.

The main considerations in using an SA algorithm for simulation optimization are the following:

- obtaining a gradient estimate $\widehat{\nabla} l(\theta_n)$;
- selecting the gain sequence $\{a_n\}$;
- choosing a stopping rule.

Stopping rules are based on the progression of the iterates, the gradient estimates, or some combination. When considering long-run (steady-state) performance measures, there is an additional consideration of choosing the observation length for each iteration. Each of the first two items are now discussed in more detail.

Gradient estimation in stochastic simulation has been a very active research field, starting in the 1980's. Approaches that provide an unbiased estimate of the gradient (leading to Robbins-Monro SA algorithms) rely on some knowledge of the underlying system, and include perturbation analysis, the likelihood ratio/score function method, and weak derivatives; see Fu (2006, 2008) for recent surveys/tutorials with references. These techniques are sometimes referred to as "white box" approaches to simulation optimization (Pflug 1996).

When the simulator is treated as a black box of inputs and outputs, the usual approach is to use finite differences, either one-sided or symmetrical, given respectively by

$$\frac{L(\theta_n + c_n e_i, \omega_n^{i+}) - L(\theta_n, \omega_n)}{c_n}, \tag{3}$$

$$\frac{L(\theta_n + c_n e_i, \omega_n^{i+}) - L(\theta_n - c_n e_i, \omega_n^{i-})}{2c_n}, \tag{4}$$

where e_i denotes the unit vector in the ith direction, $\{c_n\}$ is a positive sequence converging to 0, ω_n^{i+} and ω_n^{i-} denote the pair of sample paths (simulation replications) used for the ith component of the n th iterate of the algorithm, and ω_n denotes the original sample path (replication) used to estimate the performance measure itself. The method of common random numbers employs $\omega_n^{i-} = \omega_n^{i+} = \omega_n$. Other approaches that also treat the simulator as a black box include harmonic differences based on frequency domain experimentation and simultaneous perturbations (Spall 2003). The latter approach has the advantage that it only requires two simulation replications per gradient estimate, regardless of the dimension of the parameter vector.

Convergence of SA algorithms can be guaranteed by establishing conditions on the objective function $l(\theta)$, the gain sequence $\{a_n\}$, and the bias and variance of the gradient estimator $\widehat{\nabla} l(\theta)$. Generally, the objective function should be differentiable and either convex or unimodal; in the unconstrained version (without the projection operator), additional conditions are needed. The gain sequence should be diminishing at an appropriate rate: too fast will lead to premature convergence to a suboptimal solution, and too slow will not guarantee (almost sure or with probability 1) convergence to the optimum. The bias of the gradient estimate must go to zero, and the variance must generally be uniformly bounded. As opposed to conditions placed on the gain sequence,

conditions on the objective function and gradient estimator may not be directly verifiable in simulation optimization.

One set of common assumptions satisfying the gain sequence conditions for convergence w.p. 1 is $\sum_n a_n = \infty, \sum_n a_n^2 < \infty$, which for example the harmonic series $a_n = a/n$ (for some constant a) satisfies. In the harmonic series sequence of step sizes, a decrease is taken at every iteration, but this may lead to rather slow convergence, so in practice, sequences that decrease more gradually, or even constant step sizes, are often employed. The gain sequences are generally of the form $a_n = a/n^{\alpha}$ and $c_n = c/n^{\beta}$, where α, β, a, and c are constants to be selected, subject to $\alpha \leq 1$ and $\alpha - \beta > 0.5$. Under these conditions, the optimal asymptotic convergence rates can be achieved: $n^{-1/2}$ for the Robbins-Monro algorithm, and $n^{-1/3}$ ($n^{-1/4}$) for the Kiefer-Wolfowitz symmetric (one-sided) differences.

One empirical observation to note is that iterate averaging often demonstrates superior performance over simply using the iterate itself, i.e., using $\overline{\theta}_n = \sum_{i=1}^{n} \theta_i/n$ as the estimate of the optimum; see Pflug (1996) for further discussion and references.

Sample Average Approximation

The basic idea of the sample average approximation approach is to replace expectations in the objective function and/or constraints with their sample averages, where the samples are obtained through simulation replications, and then solving the resulting problem formulation using deterministic optimization techniques, e.g., from mathematical programming. For example, in the formulation given by (1), where $l(\theta) = E[L(\theta, \omega)]$, if a sample of N independent and identically distributed versions ω_i, $i = 1, \ldots, N$, are obtained through simulation, then one would simply solve the sample average problem (sometimes called the stochastic counterpart)

$$\min_{\theta \in \Theta} \frac{1}{N} \sum_{i=1}^{N} L(\theta, \omega_i),$$

where the formulation written in this manner implicitly assumes the constraint set Θ does not involve quantities estimated from simulation. In many (if not most) SAA applications, much more is known about the form of L than in the settings assumed by other procedures described in this entry, e.g., linearity or convexity, because the types of problems that are solved by this method generally arise in mathematical programming settings where there is structural knowledge about the problem. For example, SAA is used for stochastic programming when the scenario structure makes it impractical to calculate the expectation analytically, so that sampling becomes the preferred computational method. More details on convergence and statistical properties can be found in the accompanying SAA entry.

Response Surface Methodology

Response surface methodology is based on statistical design of experiments methodology. The approach falls into the black box category and attempts to fit a polynomial of appropriate degree (possibly after some initial transformation on the input parameters, called factors) to the performance measure of interest (called the response). The application of RSM to simulation optimization takes one of two forms:

- metamodels,
- sequential procedures.

Using a metamodel for optimization means simply dividing the problem into two separate problems of estimation and optimization, similar in philosophy to the sample average approximation approach, the big difference being that in metamodeling an estimate of the output L is obtained for specific values of θ to be determined through appropriate statistical design of experiments. After choosing the design points of θ at which to simulate, the outputs are used to fit a global response curve called the metamodel – a complete functional relationship between the performance measure and the parameters of interest – which is then treated as a deterministic function and optimized using applicable deterministic procedures. An extensive discussion of the statistical issues involved can be found in Kleijnen (2008), which also discusses alternatives to polynomial regression such as kriging.

The more common use of RSM for simulation optimization is a sequential procedure. Instead of exploring the entire feasible region, which may be

inefficient or impractical, small subregions are explored in succession according to their potential improvement. A point – usually the center of the subregion currently being explored – would represent the current best value of the parameter. The basic algorithm can be described as follows:

Phase I (iterated a number of times)

- First-order experimental designs are used to obtain a least-squares fit linear model. Then, a steepest descent direction is estimated from the model, and a new subregion chosen to explore via

$$\theta_{n+1} = \theta_n - a_n \widehat{\nabla} l(\theta_n),$$

where θ_n is the representative point of the n th explored subregion, $\widehat{\nabla} l(\theta_n)$ is the estimated (from the fitted linear response) gradient direction, and a_n represents the step size multiplier determined by a line search or some other means. This is repeated until the linear response surface becomes inadequate, which is indicated when the slope is approximately zero, by which the interaction effects become larger than the main effects.

Phase II (performed once)

- A higher order (e.g., quadratic) response surface is fitted using more detailed second-order experimental designs, and then the optimum determined analytically from this fit.

Since Phase I is iterative, it is desirable to carry out fewer simulation replications if possible, whereas in Phase II, the region should be explored quite thoroughly by using a large number of replications. The iterative algorithm in Phase I is identical in form to SA as given by (2), although in RSM, θ is simply a representative point of the current subregion being explored, and $\{a_n\}$ is not generally a decreasing sequence nor is it held constant.

Concluding Remarks

Due to the rapid advances in computational power, the possibilities for simulation optimization have made this topic a very active area of research of OR/MS; more technical details on the areas touched upon here can be found in Fu (2013). Much of the most current work is reported in the annual *Proceedings of the Winter Simulation Conference*. Discussion of issues separating theory and practice can be found in Fu (2002, 2007).

A closely related field that has shown great promise in large-scale problems is the use of simulation in conjunction with stochastic dynamic programming problems, as opposed to the static optimization setting described here. In particular, simulation is used to estimate the optimal cost-to-go or value function. More details on this approach called approximate dynamic programming, which grew out of ideas in the artificial intelligence community in a field called reinforcement learning, can be found in Gosavi (2003) and Powell (2011); see also Chang et al. (2007) for the Markov decision process setting.

See

- ▶ Approximate Dynamic Programming
- ▶ Cross-Entropy Method
- ▶ Markov Decision Processes
- ▶ Nested Partitions Method
- ▶ Nonlinear Programming
- ▶ Optimization
- ▶ Perturbation Analysis
- ▶ Response Surface Methodology
- ▶ Sample Average Approximation
- ▶ Score Functions
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Statistical Ranking and Selection
- ▶ Stochastic Approximation
- ▶ Stochastic Programming
- ▶ Variance Reduction Techniques in Monte Carlo Methods

References

Andradóttir, S. (2006). Chapter 20. An overview of simulation optimization with random search. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science: Simulation* (pp. 617–632). Amsterdam: North Holland, Elsevier.

Barton, R., & Ivey, J. (1996). Nelder-Mead simplex modifications for simulation optimization. *Management Science, 42*, 954–973.

Bechhofer, R. E., Santner, T. J., & Goldsman, D. M. (1995). *Design and analysis of experiments for statistical selection, screening, and multiple comparisons*. New York: Wiley.
Chang, H. S., Fu, M. C., Hu, J., & Marcus, S. I. (2007). *Simulation-based algorithms for Markov decision processes*. London: Springer (2nd edn, 2013).
Fu, M. C. (2002). Optimization for simulation: Theory vs. Practice (Feature Article). *INFORMS Journal on Computing, 14*(3), 192–215.
Fu, M. C. (2006). Chapter 19. Gradient estimation. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science: Simulation* (pp. 575–616). Amsterdam: Elsevier.
Fu, M. C. (2007). Are we there yet? The marriage between simulation & optimization. *OR/MS Today*, June 16–17, 2007.
Fu, M. C. (2008). What you should know about simulation and derivatives. *Naval Research Logistics, 55*(8), 723–736.
Fu, M. C. (Ed.) (2013). Handbook on simulation. New York: Springer.
Gosavi, A. (2003). *Simulation-based optimization: Parametric optimization techniques and reinforcement learning*. Dordrecht: Kluwer.
Ho, Y. C., Sreenivas, R. S., & Vakili, P. (1992). Ordinal optimization of DEDS. *Journal of Discrete Event Dynamic Systems, 2*(2), 61–88.
Hong, L. J., & Nelson, B. L. (2006). Discrete optimization via simulation using COMPASS. *Operations Research, 54*, 115–129.
Hu, J., Fu, M. C., & Marcus, S. I. (2007). A model reference adaptive search method for global optimization. *Operations Research, 55*(3), 549–568.
Kim, S.-H., & Nelson, B. L. (2006). Chapter 17. Selecting the best system. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science: Simulation* (pp. 501–534). Amsterdam: Elsevier.
Kleijnen, J. (2008). *Design and analysis of simulation experiments*. New York: Springer.
Larrañaga, P., & Lozano, J. A. (2001). *Estimation of distribution algorithms: A new tool for evolutionary computation*. New York: Springer.
Law, A. M., & Kelton, W. D. (2000). *Simulation modeling and analysis* (3rd ed.). New York: McGraw-Hill.
Nelson, B. L. (2010). Optimization via simulation over discrete decision variables. In J. J. Hasenbein (Ed.), *Tutorials in operations research: Risk and optimization in an uncertain world* (pp. 193–207). Hanover, MD: INFORMS.
Pflug, G. C. (1996). *Optimization of stochastic models*. London: Kluwer.
Powell, W. B. (2011). *Approximate Dynamic Programming: Solving the Curses of Dimensionality* 2nd edition, Wiley, New York, NY.
Powell, W. B., & Ryzhov, I. O. (2012). *Optimal Learning*. Wiley, New York.
Rubinstein, R. Y., & Kroese, D. P. (2004). *The cross-entropy method: A unified approach to combinatorial optimization, Monte-Carlo simulation, and machine learning*. New York: Springer.
Shi, L., & Ólafsson, S. (2008). *Nested partitions optimization: Methodology and applications*. New York: Springer.
Spall, J. C. (2003). *Introduction to stochastic search and optimization*. New York: Wiley-Interscience.

Simulator

An artificial means used to model a real-world system, generally falling into two categories: physical simulator or computer simulator. The former category typically consists of machines – e.g., flight or other vehicle simulators, or military weapons systems – which mimic the real world by generating cues for the operator(s), accepting inputs by the operator(s), and simulating realistic responses by the machine. The cues may include visual, auditory, and tactile sensations. The operator inputs consist of manipulation of control systems similar to those of the equipment being simulated. A computer simulator refers to the underlying computer code or software of a mathematical simulation model, e.g., Monte Carlo or discrete-event simulation. Obviously, machine simulators incorporate computer simulation, as well. Simulators are also commonly used for gaming.

See

- ▶ Control Theory
- ▶ Gaming
- ▶ Simulation of Stochastic Discrete-Event Systems

Single-server Network

A queueing network with one or more nodes and one server servicing all of them. Examples are a token-ring system with (rotating) possession of the token giving a particular node the right to access the server, and polling systems whereby a rule governs the movement of the server around a sequence or loop of queueing stations. Practical applications include local area computer networks, robots working on a production line, and sequential physical service operations.

See

- ▶ Networks of Queues
- ▶ Queueing Theory

Singular Matrix

A square matrix whose determinant is equal to 0, which means that the columns (and rows) of the matrix are linearly dependent.

See

- ▶ Matrices and Matrix Algebra

Sink Node

A node in a network through which all (or some) of the flow in the network leaves the network.

See

- ▶ Network

SIRO

Service in random order; a queueing service discipline in which customers are served in random order unrelated to their arrival times.

See

- ▶ Queueing Theory

Six Sigma

- ▶ Quality Control

Skew-symmetric Matrix

A square matrix $\boldsymbol{A} = (a_{ij})$ is skew-symmetric if $a_{ij} = -a_{ji}$. Thus, all diagonal elements are zero.

See

- ▶ Matrices and Matrix Algebra
- ▶ Symmetric Zero-Sum Two-Person Game

Slack Variable

A nonnegative variable that is added to a linear inequality of the form $\sum_j a_{ij} x_j \leq b_i$ to transform the inequality into an equation. The slack variable measures the difference between the right-and left-hand-sides of the inequality.

See

- ▶ Logical Variables
- ▶ Slack Vector
- ▶ Surplus Variable

Slack Vector

The column representation of a slack variable in a linear-programming problem.

See

- ▶ Slack Variable

SLP

Successive linear programming.

Smooth Patterns of Production

In a production-planning problem with many production cycles, one usually attempts to have the

amounts produced in the sequence of cycles to be the same or as similar as possible. Such a smooth production pattern (one with small fluctuations) tends to be more cost efficient than one that has large fluctuations. Many such production problems can be formulated as linear-programming problems.

See

- ▶ Inventory Modeling
- ▶ Operations Management
- ▶ Production Management

Societal Complexity

Dorien J. DeTombe
International Research Society on Methodology of Societal Complexity, Amsterdam, The Netherlands

Introduction

The methodology of societal complexity is in essence a field of Operational Research in the way the founders of Operational Research wanted to look at the world and wanted to help the problems in the world (Ackoff 1974, 1978). The field of methodology of societal complexity concentrates on handling real life problems, everyday problems that are described on the front page of newspapers.

Complex societal problems are worldwide natural problems caused by viruses like flu pandemics, fowl plagues and HIV/AIDS, local natural disasters such as earthquakes, hurricanes, avalanches and floods, technical dangers caused by industry like pollution (CO2), traffic, nuclear power plants, climate change and agricultural business, manmade threats like (world) wars, terrorism, Internet vulnerability and stock exchange manipulation, credit crisis and identity theft. These problems cause much trouble to the people, the economy and the stability of states. Handling these kinds of problems belongs to the field of methodology for societal complexity. The claim of this field is that complex societal problems should be handled in according to the approaches, methods and tools of this field.

The Field of Methodology of Societal Complexity

The field of methodology of societal complexity focuses on handling complex societal problems. A complex societal problem can be defined as

> a complex interdisciplinary societal problem is a real life problem, which concerns a real life situation. The problem can be in the present or in the (near) future, latent or manifest, structural or incidental. The problem is often undefined. The problem concerns many domains. The problem is dynamic and imbedded in a dynamic world. There are many phenomena and parties involved. The problem has a large impact on each level of aggregation of the society and provokes much emotion. Due to the many aspects of the problem the problem is complex. The problem can be urgent or less urgent. A solution is not easy at hand and the desired situation is not always clear and difficult to find.

and problem handling as

> Handling a problem is the process of analyzing, defining or changing a problem in order to gain more insight into the problem, whether or not this leads to influencing the problem in order to reach the desired situation. This process can be performed actively or passively, consciously or unconsciously, routinely or once-only, whether it is by circumventing or by forgetting the problem, by shifting the problem to another party or by (partly) changing the problem, by imagination or in real life, whether through thinking, applying tools and/or methods.

Not every complex societal problem will be handled. It depends whether it will be put on the political agenda. If the problem-handling process is examined it can be distinguished between several phases of the problem-handling process (Fig. 1).

After becoming aware of a complex societal problems and reflecting on the problem to be able to do some real-life interventions, the problem should be placed and accepted on the political agenda of a recognized problem owner who has credibility to handle this type of problem. For instance, reflection on the CO2 emission problem in relation to climate change, a recognized problem owner could be the Kyoto convention of 1997 or the decision of the G20.

When one decides to handle the complex societal problem, one should look at the field of societal complexity for a methodology the guide the problem-handling process.

Complex Societal Problems

Complex societal problems are commonly handled as mono-disciplinary problems by considering just

Societal Complexity, Fig. 1 The sub-cycles of the problem handling process

Sub-cycle I: defining the problem

phase 1.1	becoming aware of the problem and forming a (vague) mental idea of the problem
phase 1.2	extending the mental idea by hearing, thinking, reading, talking and asking questions about the problem
phase 1.3	putting the problem on the political agenda and deciding to handle the problem
phase 1.4	forming a problem-handling team and starting to analyze the problem
phase 1.5	gathering data, exchanging knowledge and formulating hypotheses about the problem
phase 1.6	formulating the conceptual model of the problem

Sub-cycle II: changing the problem

phase 2.1	constructing the empirical model and establishing the desired goal
phase 2.2	defining the handling space
phase 2.3	constructing and evaluating scenarios
phase 2.4	formulating hypotheses and suggesting interventions
phase 2.5	implementing interventions
phase 2.6	evaluating interventions and the problem handling process

a single disciplinary department and by asking single disciplinary experts for advice in handling the problem. However more and more politicians are aware of the limitations of the boundaries of their department and of disciplines to handle complex societal problems. Complex societal problems often extend the boundaries of their primary field.

For example, handling a natural disaster like a hurricane, a tsunami or an earthquake takes preparation to mitigate the damage beforehand and needs much coordination at the moment of the disaster and support afterwards. Not only are scenarios and plans required, but also training, communications and coordination are vital. There is a distinction between prevention and disaster, in case of a tsunami, building flood free safe houses in protected areas, preparation for the moment of a disaster, handling on the moment of the disaster and shortly after the disaster and reflecting on the effects in the years afterwards the disaster. To be prepared for disasters, multidisciplinary groups of experts should discuss how science, updated technology and communication tools can support the prevention, the moment of and the support afterwards in a disaster. The support of methods and tools of the field of societal complexity can help prevent unnecessary damages when disaster threatens, coordinate support, and protect people and goods.

The healthcare system is also a complex societal system. Therefore some of the interventions in the healthcare system should be done along the lines of the field of societal complexity. The healthcare system includes many aspects like prevention, healthcare of the patient, and patient follow-up afterwards. The doctor-patient relation is not a one-to-one relationship; it is part of a network of relationships. The patient is imbedded in a societal relation as daughter, mother, grandmother, employee, and citizen. A doctor is imbedded in a network of professional assistance, such as social work, pharmacy, medical specialism, hospital, medical research medical industry, assurance company.

Prevention is preferred to curing. The chain of patient safety starts with prenatal baby care and is closely related to social policy to prevent pollution, connected with sustainable food production, preventing smoking, traffic safety, preventing child and elderly abuse, education on drinking, etc. In this chain the medical world should give feedback to the society of the diseases they encounter. Improving patient safety is improving people safety. Specialization and division in tasks is fruitful. However, a patient is not just a hip, or a leg, the patient is a human being and a member of the society.

The medical world is a part of the horizontal and a vertical chain. The horizontal chain is, for instance, on the level of family doctor: the social worker, employer, dentist family members. The vertical chain is the family doctor, specialist, hospital and insurance companies. Medical professionals and non medical professionals on micro, meso and macro level.

The medical world is imbedded in the social world, and has aspects of law, psychology, sociology, policy on local, state and international level. Therefore some of the healthcare problem should not be handled mono-disciplinary but should be approached as a complex societal problem using methods and tools of the field of societal complexity.

Compram Methodology

One of the methodologies for handling societal complexity is the Compram methodology, developed by DeTombe (1994–2010) (DeTombe 1994, 1999, 2001, 2008a, b, 2010).

The Compram is the next step after Soft Systems Methodology (Checkland 1982) and System Dynamics. It combines aspects of different approaches into a structured interactive approach for policy making in collaboration with experts and stakeholders in order to find possible transitions of the situation that can be mutually accepted and implemented into real life.

The Compram methodology is based on the idea that societal problems must be handled multidisciplinary and cooperatively with experts, policy makers and stakeholders together. These difficult and complicated group processes are guided and structured by a facilitator in a six step approach. Multidisciplinary experts, stakeholders and policymakers discuss the content and possible solutions based on a cooperative (simulation) model of the problem. The method emphasizes facilitating the exchange of knowledge, and understanding and communication among the experts, stakeholders and politicians.

Knowledge, power and emotions are the basic elements in handing complex societal problems. The Compram methodology is a prescriptive framework method to which all kind of sub-methods can be applied. The Compram methodology has been used as a theoretical basis for handling over sixty real life cases in the field of societal policy making and in real life complex societal problems in several countries of all continents. The Compram methodology is advised by the OECD (July 2006) to handle complex societal issues. The 'Final consensus report' from the OECD Global Science Forum Workshop held in Tokyo, Japan, on December 5–6, 2005 organized by the JST-RISTEX (Research Institute for Science and Technology for Society, Japan Science and Technology Agency).

Many complex societal problems are a threat to people, the economy and the stability of the state, but most of all the quality of life. In order to create a safer society one needs to know where the danger comes from and what causes the threat. Each threat has different causes and different effects on different elements in society. Therefore, one has to carefully analyze the situation, make a distinction between causes and effects, see the elements and how they are related, see which power groups are involved, and to find out which package of sustainable changes can have the desired effects.

To find out what is known about the problem one has to analyze who is effected by it, which parties are involved, who benefits and who suffers, and what emotions and political vulnerability are going on. This needs an interdisciplinary approach. An interdisciplinary group of knowledge experts should analyze the situation and discuss possible changes. Then stakeholders should discuss the issue and give their opinion on the situation. Together the experts and stakeholders should find some fruitful changes. The interventions should be carefully implemented and evaluated on their desired effect on the problem. Each complex societal problem has knowledge, power and an emotional element.

The Compram methodology starts when the problem owner invites a facilitator to guide the problem by handling the process according to the Compram methodology.

Handling complex societal problems needs a special approach. Handling societal problems in an interdisciplinary way has become a must for society and a challenge for the human sciences. The problems society is confronted with are difficult to handle. There is a growing gap between the complexity of these problems and the human capacity to deal with them. There is a need for better methods and tools, more knowledge and imagination. Scientific knowledge is needed to survive amidst these problems.

Some of the scientific reasons for this special approach are that the problems are seldom defined, change during their development, many stakeholders are involved often with a different view on the problem, with different interest and with different 'solutions' in mind. Societal reasons for this special approach are the importance of these problems for society, the impact they have on many people, and the large amount of money involved. Combining the

effort of scientists who are working in this field is an inspiring serious challenge from the perspective of a number of disciplines. Combining existing knowledge and creating new insights with methods and tools for supporting complex societal problems is a challenge for scientists from different fields.

Most political problems are handled directly in relation with powerful groups involved in the problem, who by way of lobbyists influence the decisions of the political interventions. However, this directs the definition of the problem and therefore the solution of the problem directly to the definition of the most powerful groups in the problem handling process neglecting the more powerless groups. In order to give the powerless group also a chance and in order to be able to really see how the problem looks like a team of so-called neutral experts should first define the problem before the problem is handled. These neutral experts are selected based on their knowledge of a part of the field of which the problem is.

The facilitator therefore selects a group of neutral experts, neutral toward a certain definition and a certain solution to the problem and invited them to make a definition of the problem. In several meeting the experts each coming from a different discipline defines the problem. They do this guided by the facilitator with the help of a seven layer communication model (see Fig. 2), in which they first describe the problem in words, then define the concepts , then identify the knowledge. Is what the experts say based on theory, assumption or experience? Then base on the description of the problem they make together a simulation model. The relations of the phenomena and their effect on each other can then be carefully described and simulated. In the conceptual model next to the phenomena the power groups and their emotions are identified and described. This conceptual model can then after agreeing on the content be altered in an empirical model.

After the empirical model the handling space will be discussed. How much space is there for the problem owner to change the problem? Should this be here and now concerning a country, a continent or the world? Then, based on several scenarios the group of experts can suggest several intervention and intervention strategies.

The Compram methodology consists of 6 steps of interventions of a complex societal problem (see Fig. 3).

The second step is inviting the power groups. Power groups with much influence and less influential power groups.

Groups of stakeholders who benefit from the problem and groups of stakeholders who suffer from the problem. The facilitator invite each group separate and stimulated them to undergo the same process of problem handling described above to define their own view of the problem, their own definition and to describe the interventions they want to do and their intervention strategies.

Based on these outcome mutual meetings of neutral experts and stakeholders groups are then invited to look at the problem and find mutual accepted interventions. Then groups are formed to formulate implementations strategies for changing the problem and to guide and later on evaluate implementations for changing the problem (Fig. 3).

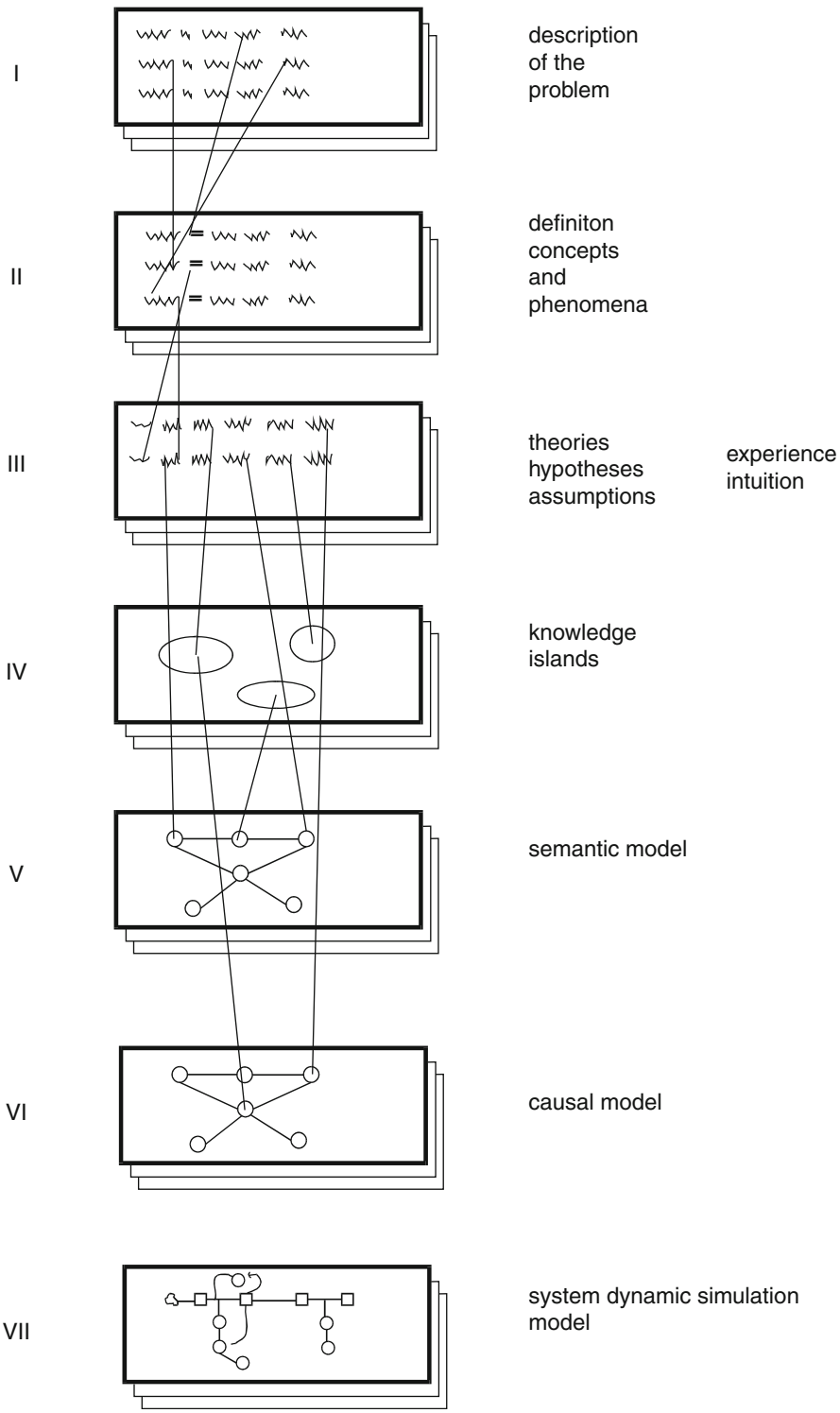
The Use of the Methodologies of the Field of Societal Complexity in Real Life

The Compram methodology is recognized by the OECD for handling global safety. That means that in order to develop and combine the knowledge, the methods and tools for handling societal complexity problems, special multidisciplinary knowledge institutes should be created that can become aware of future and now-a-days dangers and threats. These institutes perform multidisciplinary research and advice policy makers how to handle global safety issues in an integrated multidisciplinary, multi actor approach. In order to accomplice this, each country should establish multidisciplinary centers for research on societal complexity. These institutes should focus on their own specific local complex societal problems in cooperation with the already existing local institutes on safety. International they should cooperate with same kind of institutes on global threats.

These centers should be closely connected to the university. Inside the university a department for societal complexity may be established. In order to give some ideas this department can start with a team of scientists mainly interested or educated in methodology, coming from alpha, beta and gamma sciences.

Although researchers in this field are convinced that complex societal problems should be handled according to the direction of the field of methodology of societal complexity, relatively few complex societal problems are handled in this way. There are

Societal Complexity,
Fig. 2 The seven-layer model



step 1 analysis and description of the problem by a team of neutral content experts
step 2 analysis and description of the problem by different teams of stakeholders
step 3 identification of interventions by experts and stakeholders
step 4 anticipation of the societal reactions
step 5 implementation of the interventions
step 6 evaluation of the changes

Societal Complexity, Fig. 3 The six steps of the Compram methodology

several reasons for the reluctance of the politicians to handle complex societal problems.

First of all, most of the politicians are not aware of or unfamiliar with the methodology. This could be met by a more structured approach to teach managers, future politician and university students in their basic study in the field of for instance agriculture, healthcare, economy and transport. The main concepts and ideas of the field of societal complexity could be included in their basic university education. So that they are later on in their professional life at least aware of a more fruitful and sustainable approach towards complex societal problems.

A second point is that politicians want to jump to conclusions and do not want to spend too much time in defining the problem. They want directly to deal with the powerful stakeholders to find mutual accepted solutions. They like to start directly with problem handling phase 2.3 and 2.4 (see Fig. 1).

Another point is the transparency of, for instance, Compram. This methodology is based on a democratic decision-making process and is made transparent by prescribing that all the activities in each step should be open reported afterwards including the result, who were involved in the decision process, and what has be discussed in this process. Not all politicians want this openness.

Yet another point is the structure of most of the government departments. These departments are separated from each other and it is very hard due to competition and power fights and budgets to work together on a mutual problem.

See

- ▶ Complex Problem Analyzing Method (Compram)
- ▶ Practice of Operations Research and Management Science
- ▶ Problem Structuring Methods
- ▶ Robustness Analysis
- ▶ Soft Systems Methodology
- ▶ Strategic Choice Approach (SCA)
- ▶ Strategic Options Development and Analysis (SODA)
- ▶ System Dynamics
- ▶ Wicked Problems

References

Ackoff, R. L. (1974). *Redesigning the future*. New York: Wiley.
Ackoff, R. L. (1978). *The art of problem solving*. New York: Wiley.
Checkland, P. B. (1982). Soft systems methodology as process: A reply to M.C. Jackson. *Journal of Applied Systems Analysis, 9*, 37–39.
DeTombe, D. J. (1994). Defining complex interdisciplinary societal problems. *A theoretical study for constructing a co-operative problem analyzing method: the method compram*. Amsterdam: Thesis publishers Amsterdam (thesis), pp 439. ISBN 90 5170 302–3.
DeTombe, D. J. (1999). Facilitating complex technical policy problems. In E. Stuhler & D. J. DeTombe (Eds.), *Cognitive psychological issues and environment policy application, research on cases and theories* (Vol. 5, pp. 119–127). Munchen/Mering: HamppVerlag. ISBN ISBN3-87988-355 -6, ISSN 0940–2829.
DeTombe, D. J. (2001). Compram a method for handling complex societal problems. *European Journal of Operational Research, 129*, 2, 16 March 2001.
DeTombe, D. (2008a). Towards sustainable development: A complex process. *International Journal on Environment and Sustainable Development, 7*(1), 49–62.
DeTombe, D. (2008b). 'Climate change: a complex societal process; analysing a problem according to the Compram methodology'. *Journal of Transformation & Social Change*, 5.3, (pp. 235–266), doi:10.1386/jots5.3.235/1.
DeTombe, Dorien J. (2010). Global safety. *Pesquisa Operacional, 30*(2): 387–404, Maio a Agosto de 2010 387. versão impressa ISSN 0101–7438 / versão online ISSN 1678–5142

Soft Systems Methodology

Peter Checkland
Lancaster University, Lancaster, UK

Introduction

Soft Systems Methodology (SSM) is an approach to tackling the kind of problematical situations with

which managers of all kinds and at all levels have to deal in their professional lives. As its name implies, SSM is based upon systems ideas, and a systems approach, but not in the conventional sense in which those phrases are usually used. Normally those words imply taking parts of the real world to be systems, and improving the performance of those systems in meeting their declared objectives. This is the approach found, for example, in systems engineering, systems analysis, classical operations research (OR), and in most management science textbooks. SSM was developed in the 1970s and 1980s in management situations in which objectives were themselves problematic and the engineering/optimizing approaches developed in the 1950s and 1960 could not be used unchanged. It is thus complementary to the earlier methods.

SSM was developed in a program of action research in the kinds of real-world situations that are often referred to as wicked problems (Rittel and Webber 1973) or messes (Ackoff 1974), and it is SSM which is the main source of the distinction now commonly made between hard and soft methods (discussed below); and it is a main component of what is thought of (especially in Europe) as soft OR.

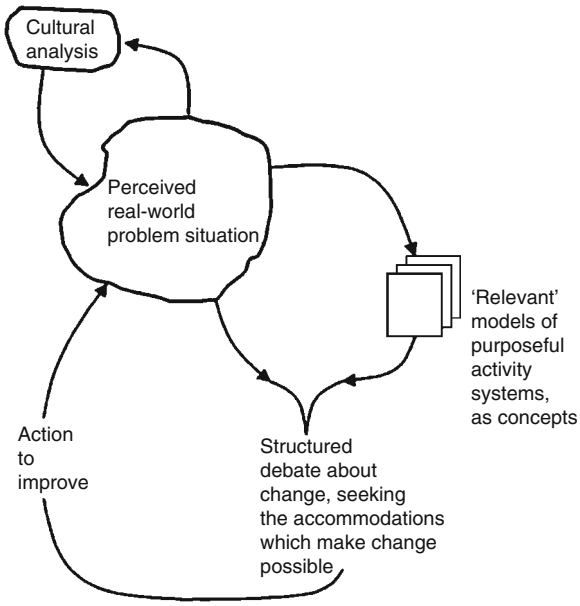
Origins and Development

At Lancaster University in the UK in the mid-1960s, a postgraduate department of systems engineering was established. The first external project carried out examined the possibility of controlling a paper-making line by computer, a classic systems engineering study. However, the late Gwlym Jenkins, who established the department, always interpreted the word engineering in its broad sense, the sense in which you can engineer a meeting with someone, or the release of hostages. A program of action research was therefore established by Checkland in order to investigate the possibility of using systems engineering methods not in technically defined problems but in the kind of general problem situations faced by managers. It was discovered that in such situations the straightforward definition of the relevant system and its defined objectives was not possible. Individuals and groups of people having different interests were always involved in such situations; what was problematical, and how, was always a matter of judgment; and the idea of finding solutions that eliminated problems was too simplistic a concept. For example, an early study examined the then-current Anglo-French Concorde project, already beginning to overrun its cost and time estimates, and a matter for much public debate in the U.K. It was not enough to think of such a development only as "a system to develop the world's first supersonic passenger-carrying aircraft." The problem situation as a whole included political, cultural, economic, environmental and employment issues, having been set up when President de Gaulle of France was vetoing British entry into the European Common Market.

Progress was made in such situations by realizing that all real-world problem situations, large or small, public-sector or private-sector, have at least one characteristic in common: they contain people seeking to take purposeful action. Methods were developed of making systems models of human-activity systems. However, it was soon also realized that any purposeful action is always open to many different interpretations, one observer's terrorism being another's freedom-fighting. Therefore the approach emerged of first making a number of models thought relevant to the different interests at work in a problem situation, each model being based upon an explicit declared world view. Such a clutch of models could then be used to structure a debate with people in the problem situation, the debate focusing on a search for action likely to bring about improvement in the situation. The structuring of the debate was done by using the models as a source of questions to ask of the real situation, or a source of possible scenarios that could be compared with recent happenings in the situation in question.

SSM thus developed as a learning, or inquiring system having the form shown in Fig. 1, one based on models of human activity systems as an explicit structuring device which could bring rigor to a debate about change. The change itself might be procedural, structural, attitudinal or, as is common, some mix of all three. It is important to note that the models are never taken to be descriptions of the real world; they are only devices to help structure debate.

The first paper on SSM was published in the early 1970s (Checkland 1972), and there is now a large literature, both primary and secondary. The development of SSM over more than 40 years has



Soft Systems Methodology, Fig. 1 The learning system which is Soft Systems Methodology (reprinted with permission of John Wiley and Sons from Checkland and Holwell's *Information, Systems and Information Systems*, 1997)

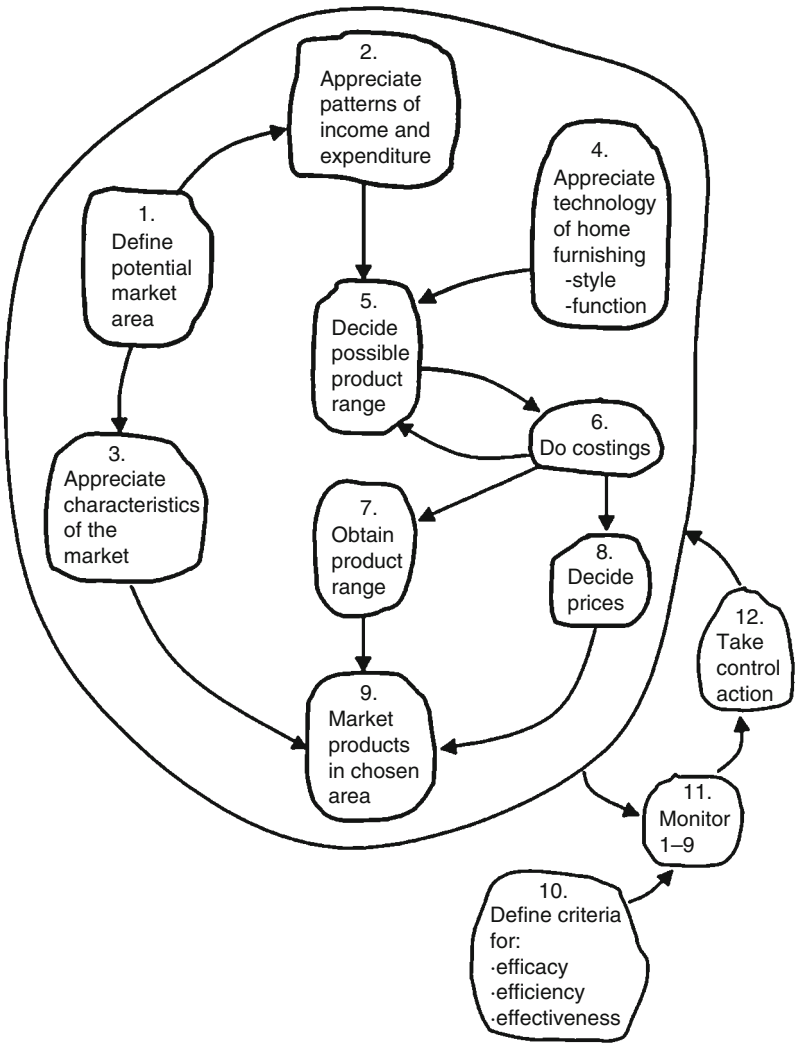
entailed the refinement, in practice, of the different aspects of the inquiring system: ways of finding out about a problem situation; choosing relevant models; model building; conducting the debate; defining useful change. These aspects will be briefly described. See Checkland and Poulter (2006) for a detailed account.

The Process of SSM

Overall, a use of SSM can be thought of as an intervention in a human situation that has arisen among a group of people in a particular setting, in order to find accommodations which will enable action to improve the situation to be taken. It therefore consists of analysis to understand the story of the situation and how it has arisen, and analysis to find out (and help to create) the accommodations between different interests and world views which enable action to be taken. Explicit use of the approach leaves a recoverable trail that can be traced by anyone interested. It has to be accepted that action in human situations cannot match the repeatability criterion of natural science carried out in laboratories. But SSM allows the full recoverability of the intellectual process that enables action-to-improve to be taken (Checkland and Holwell 1998).

Three analyses serve the finding out about the problem situation, something which will inevitably continue throughout a study. Discovering the history of the situation is reinforced, in Analysis One, by an analysis of the intervention itself. Who caused it to take place?; Who is doing something about it?; and most important – Who could the issue owners be taken to be? A list of plausible issue owner's is a useful source of choices of models of human-activity systems likely to be relevant to exploring the situation. (Discovering which models are truly relevant will emerge in moving round the learning cycle of Fig. 1, something which will be done many times during a serious study.) Analyses Two and Three feed an understanding of the context of the problem situation. The former explores the social reality: the roles, norms, and values taken seriously by people in the situation; the latter finds out the ways in which power is manifest in the situation. Both will enrich understanding of what action might be taken to bring about improvement.

SSM's stream of logical (task-oriented) analysis starts by building the devices that are human-activity system models. They consist of a structured set of activities that as a whole would constitute purposeful activity, together with monitoring and control activities which would in principle enable the purposeful whole to adapt and survive in a changing environment. Such models are normally built from a statement of a purposeful activity, known as a root definition, which embodies a declared Weltanschauung or world-view. A full root definition will usefully use the formula: a system to do P by means of Q in order to help achieve R. This answers What? How? and Why? questions about the activity and leads to the monitoring and control of the system being in terms of criteria for efficacy (is the output achieved?) efficiency (are minimum resources used?) and effectiveness (is this contributing to a longer term aim?). Figure 2 (Checkland and Holwell 1997) shows a model built by taking the existing mission statement of the Scandinavian home furnishing company IKEA as a root definition. IKEA's worldview is that a successful home furnishings business can be operated which combines elegant design with good functionality, and, what is more, can do this for a mass market. The model assembles the activities necessary to do this and links them by arrows that



Soft Systems Methodology, Fig. 2 An SSM-style model from the IKEA mission statement, concerned with marketing home furnishing items of good design and function at prices low enough to enable a majority of people to afford them (after Checkland and Holwell 1997)

show how the activities are contingent upon each other. Each activity could now be expanded into a model of higher resolution.

A final aid to formulating root definitions and building models is a set of elements that are relevant to all purposeful activity. They make up the mnemonic CATWOE, the central T of which indicates that any purposeful activity can be expressed as a transformation of an input into an output, not infrequently a need into that need met. The other elements are: customers (those directly affected by the activity, whether as victims or beneficiaries); actors (who would carry out the activities); the world-view which makes sense of this purpose; the owner who could stop the activity; and the constraints from the environment which this activity takes as given.

Once models have been built they can be used as intellectual devices to structure a questioning of the real situation. The models are of course much simpler than any description of actual real-world purposeful action (which will always embody more than one world view) but they serve to tease out often unquestioned assumptions and to initiate, as well as structure, a learning process within the situation studied.

The purpose of the comparing of models with perceptions of the real situation is, by structuring debate, to get assent to changes that are both desirable and culturally feasible, and would improve the problem situation. Obtaining that assent of course changes the original situation, so the cyclic learning process is in principle never-ending.

SSM in Use

SSM is normally taught as a stepwise process, and users at first normally approach it in this way, but that is not how experienced practitioners use it. SSM is, as its name indicates, a methodology, that is, a logos of method: a set of principles of method. This means that although uses of it will bear a family resemblance, it has to be adapted by a particular practitioner to a specific situation (Atkinson 1984). For experienced practitioners the methodological principles become tacit knowledge, and they will react to an evolving situation as its fluctuations demand. They will use Fig. 1 and will draw on the specific features of SSM in whatever ways are appropriate in their particular situation.

SSM can be seen as building upon work done in the systems and management science area by Churchman (1971), Ackoff (1974) and, especially, Vickers (1965). Its primary and secondary literatures contain many descriptions of its use, usually in organizational settings, the primary literature, in book form, being that which describes the action research program in which it has been developed: Checkland (1981), Checkland and Scholes (1990) and Checkland and Holwell (1997), Checkland and Poulter (2006).

Hard and Soft Systems Thinking

It is frequently stated that such hard problem-solving approaches as systems engineering and classical OR are most appropriate in well-defined situations in which objectives are known, and it is ways of achieving them which are problematical. On the other hand, a soft approach such as SSM is said to be appropriate in confused, unstructured situations in which both what to do and how to do it are problematical. This is not untrue, but it fails to make a sharp distinction between the two kinds of approach.

The core difference between hard and soft approaches lies in the different ways in which they use systems ideas (Checkland 1985). A very influential early textbook of OR, written in the 1950s (Churchman et al. 1957) argues that "the comprehensiveness of OR's aim is an example of a systems approach" and that "OR is concerned with as much of the whole system as it can encompass." This latter statement sees OR as intervening in systems assumed to exist in the real world. This is a familiar thought, deeply embedded in all of us by the way in which the word is used system in everyday language. One casually speaks of the legal system, health care systems, or the education system, assuming that these systems truly exist in the world. This is the hard assumption: that the world contains systems. In reality these areas of human activity only very rarely approach the concept of system as a fully-integrated adaptive whole. The soft approaches, on the other hand, do not assume that systemicity lies in the world. There the assumption is that whatever the perceived real world consists of (on which they are neutral), the process of inquiry into the world can be organized as a learning system. Thus, in SSM, the system is the process of inquiry itself – though SSM happens also to make use of systems models of purposeful activity, though these are not would-be descriptions of anything in the world, only devices used to structure debate.

It is this different answer to the question, "where are systems to be found?" which marks the difference between hard and soft approaches. This makes the two complementary to each other and powerful in combination.

Extensions and Advances

SSM was developed in the action research program at Lancaster University, which ran for 30 years and included more than 300 projects carried out in real situations. So it can fairly be described as a mature and well-tested process. But such a process, formed through real-world experience, can never be regarded as having reached a final state. So it is appropriate to indicate some developments that have extended the process.

As project experience accumulated it was realized that a user of SSM always faces two problematic situations. Obviously one of these is the real-world situation in which the methodology is to be used; but also there is the situation in which the user has to decide how to do the study, how to craft the principles of the methodology into an approach to this particular situation, involving the requisite people, with their particular history and world-views, now. It was realized that SSM could be used in both situations. Thus, every use of SSM could, if it seemed appropriate, entail both SSM using models to explore

the content of the situation, SSM(c), and SSM using models to explore the process of tackling the issues giving rise to the study, SSM(p). Discussion of this is found in Checkland and Winter (2006), and Checkland and Poulter (2006). It is frequently the case that as ideas for models relevant to the situation content emerge, the first model to be built is of a system to do the study, i.e. part of a use of SSM(p).

Also, the use of SSM(p) as an overall approach to project management has been taken further by Winter and collaborators. See their papers in a special issue of the International Journal of Project Management edited by Maylor (2006): Rethinking Project Management; see also Winter and Checkland (2003) and Winter and Szczepanek (2009).

Related to the recognition of SSM(c) and SSM(p) is another example of learning from experience which arose during SSM's development. Retrospective examination of a large number of root definitions from which models had been built in earlier studies revealed that most of them were of purposeful activity which was present in the real world in organizational form, as departments or divisions, or sections within organizations. (This limitation was probably due to the initial legacy of the kind of thinking embodied in hard systems engineering, namely that the real world contains systems which can be engineered, the stance which had to be replaced as SSM emerged and the fundamentals of the hard/soft distinction were recognized.) It became important to note that although organizations could create purposeful functions (such as Production, Marketing or Distribution) and embody them as structures, every organization has to carry out many more important actions beyond those that could be embodied in functional structures. For example, a major UK charity, Oxfam, grew from its origins in relief work (beginning as the Oxford Committee for Famine Relief) into a major source of development projects in the Third World. A permanent issue in the charity is the balance of resources between these two very different activities. Thus, in an SSM study of Oxfam, a highly relevant purposeful activity worth modeling is a notional system to balance resource allocation between relief and development, though this is not the function of any one structure within the organization. Such a root definition is called an Issue-based definition, whereas root definitions that relate to organization structures are known as primary task definitions (Checkland and Poulter 2006). Much experience has shown that it is best always to work with both kinds of model, not least because the boundary of an issue-based model will cut across organizational boundaries. Since internal organizational boundaries always relate to issues of politics and power within organizations, issue-based models always raise awareness and attract attention and energy.

A third area of SSM which remains to be cautiously mined concerns the nature of the very subtle process which is initiated when purposeful activity models, built according to different world-views, are used as a source of questions to explore a real-world situation. The caution stems from the fact that no two projects using SSM ever have more than a broad family resemblance to each other, simply because no two situations involving human beings are ever exactly the same. Neither are human situations ever static. In addition, the discussion and debate initiated by consideration of models based on different world-views provokes and stimulates a level of discussion beyond that which is normal in most Western organizations. Mental furniture tends to get shifted in such a debate. Unsurprisingly, attention to this aspect of SSM has arisen from Eastern interest in the approach, especially in the work of Kenichi Uchiyama, which interprets SSM from the Japanese point of view in his book, *The Theory and Practice of Actuality* (Uchiyama 2003). He came to SSM as an experienced manager. Working in both Casio and in IBM Japan, he observed two very different ways of thinking about a market. IBM conceptualized a market as an external thing outside themselves that they would enter and try to capture. In Casio, the chief executive, addressing managers in the company, would say "We must find ourselves in our market", a very different concept to that of IBM.

Uchiyama's work is centred on making the distinction between reality and actuality, a distinction that he takes from the work of Bin Kimura, an eminent Japanese psychiatrist. Uchiyama uses the example of listening to music to illustrate the reality/actuality relationship. The reality of a musical performance is that a succession of external notes (vibrations in the air) are produced and delivered. What is heard, however, is not that, but a melody, as a whole entity, and hearing it stems from a two-way link between this particular performance and one's whole already-existing experience of listening to music. It is heard in actuality, not reality. Uchiyama argues that,

via the modelling and debate stages, SSM is best seen as a methodology to bridge reality (where most Western thinking is focused) and the actuality of the people in the problem situation addressed, of which Eastern thought is more aware. This is the most insightful contribution that has so far been made to the error-strewn secondary literature on SSM, and it opens up new developmental possibilities. Uchiyama sees SSM as closer to Japanese models of thought than it is to Western thinking, and this may explain the extraordinary difficulty many people in Europe and America have in grasping, for example, the difference between hard and soft systems thinking. In the West, it is evidently extremely difficult for many people to give up the idea that systems are things out there in the world, rather than being world-view-based epistemological devices for trying to understand the world.

Expositions of SSM come in books from the Open University in the U.K., where thinking and systems methodologies have been taught since the early 1970s. The book edited by Reynolds and Holwell (2010) describes SSM and other systems approaches; Ramage and Shipp (2009) in their book, *Systems Thinkers*, describe the work of thirty selected people in the field, including Checkland, in short chapters which introduce the work and include an extract from the writings of the chosen thirty. For a discussion of the difference in treatment of SSM between the U.S. and the U.K., see Paucar-Caceres (2011).

Probably the best attempt to express the essence of SSM in a single phrase comes from J.M. Gvishiani, who was head of the Moscow Research Institute for Systems Analysis when Checkland was invited to spend a week there giving lectures and seminars. "I see your approach," he said, "as a rigorous approach to the subjective."

See

- Community OR
- Learning
- Model
- Practice of Operations Research and Management Science
- Problem Structuring Methods
- Robustness Analysis
- Strategic Choice Approach (SCA)
- Strategic Options Development and Analysis (SODA)
- Systems Analysis
- System Dynamics
- Wicked Problems

References

Ackoff, R. L. (1974). *Redesigning the future*. New York: Wiley.

Atkinson, C. J. (1984). *Metaphor and systemic praxis*. Ph.D. Dissertation, Lancaster University, UK.

Checkland, P. (1972). Towards a systems-based methodology for real-world problem solving. *Journal of Systems Engineering, 3*(2), 87–116.

Checkland, P. (1981). *Systems thinking, systems practice*. Chichester, UK: Wiley.

Checkland, P. (1985). From optimizing to learning: A development of systems thinking for the 1990s. *Journal of the Operational Research Society, 36*, 821–831.

Checkland, P., & Holwell, S. (1997). *Information systems and information systems*. Chichester, UK: Wiley.

Checkland, P., & Holwell, S. (1998). Action research: Its nature and validity. *Systemic Practice and Action Research, 11*(1), 9–21.

Checkland, P., & Poulter, J. (2006). *Learning for action*. Chichester, UK: Wiley.

Checkland, P., & Scholes, J. (1990). *Soft systems methodology in action*. Chichester, UK: Wiley.

Checkland, P., & Winter, M. (2006). Process and content: Two ways of using SSM. *Journal of the Operational Research Society, 57*, 1435–1441.

Churchman, C. W. (1971). *The design of inquiring systems*. New York: Basic Books.

Churchman, C. W., Ackoff, R. L., & Arnoff, E. L. (1957). *Introduction to operations research*. New York: Wiley.

Maylor, H. (Ed.) (2006). Rethinking project management. *Special Issue of International Journal of Project Management*, *24*(8).

Paucar-Caceres, A. (2011). The development of management sciences/operational research discourses: Surveying the trends in the US and UK. *Journal of the Operational Research Society, 62*, 1452–1470.

Ramage, M., & Shipp, K. (2009). *Systems thinkers*. London: Springer.

Reynolds, M., & Holwell, S. (Eds.). (2010). *Systems approaches to managing change*. London: Springer.

Rittel, H. W. J., & Webber, M. M. (1973). Dilemmas in a general theory of planning. *Policy Science, 4*, 155–169.

Uchiyama, K. (2003). *The theory and practice of actuality*. Tokyo: Institute of Business Research, Daito Bunka University.

Vickers, G. (1965). *The art of judgement*. London: Chapman and Hall (republished 1983, Harper and Row, London, and 1995, Sage, London).

Winter, M., & Checkland, P. (2003). Soft systems: A fresh perspective for project management. *Civil Engineering, 156*(4), 187–192.

Winter, M., & Szczepanek, T. (2009). *Images of projects*. Farnham, UK: Gower.

Sojourn Time

The total time spent in a queueing system, including both the delay and service times; sometimes called the total waiting time or just waiting time. Often used as the time spent in a visit to a state of a stochastic process such as a Markov chain.

See

- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Queueing Theory

Solution

A set of values for the variables of a problem that satisfy all the constraints of the problem.

See

- ▶ Feasible Solution

Solution Space

For a constrained mathematical programming problem, the solution space is a portion of Euclidean space defined by all the constraints of the problem. For a linear-programming problem, the solution space is defined by the intersection of the nonnegative portion of Euclidean space and the constraints of the problem.

See

- ▶ Linear Programming
- ▶ Mathematical Programming
- ▶ Mathematical-Programming Problem
- ▶ Nonlinear Programming

SOS

- ▶ Special-Ordered Sets (SOS)

Source Node

A node in a network from which all (or some) of the flow in the network enters the network.

See

- ▶ Network

SPA

Smoothed Perturbation Analysis.

See

- ▶ Perturbation Analysis

Space

Gerald W. Evans[1] and Mansooreh Mollaghasemi[2]
[1]University of Louisville, Louisville, KY, USA
[2]University of Central Florida, Orlando, FL, USA

Introduction

Man's venture into outer space began on October 4, 1957, with the successful launch by the U.S.S.R. of the first artificial Earth satellite, Sputnik I. This was soon followed on November 3 by the dog (Laika)-manned Sputnik II. The first American satellite, Explorer I, was launched on January 31, 1958. From this period on, both the U.S. and the U.S.S.R. have mounted extensive research and development activities for putting manned space vehicles into Earth orbit. In particular, the U.S. established the National Aeronautics and

Space Administration (NASA) on October 1, 1958 with the mission "... to achieve at the earliest practicable date orbital flight and successful recovery of a manned satellite, and to investigate the capabilities of man in this environment," (Swenson et al. 1966, p. 111). From this beginning, the U.S. man-in-space program can be traced from Project Mercury (Glenn in Earth orbit on February 20, 1962), to Project Gemini (two-manned space capsule in Earth orbit), to Project Apollo (the moon-landing on July 20, 1969), and to the Space Shuttle. However, the Soviet effort succeeded in sending the first manned satellite into Earth orbit with the launch of Gagarin on April 12, 1961.

During the 1970s, countries in addition to the United States and the Soviet Union/Russia became involved in the exploration of outer space. For example, the European Space Agency (ESA), established in 1975, has 18 member states and a staff of approximately 2000. The ESA has engaged in both manned exploration, mainly through its participation in the International Space Station Progam, and unmanned exploration of the moon and other planets. In addition, the People's Republic of China was the third country to independently send humans into space in 2003 (Clark 2004). For information about U.S. activities with respect to space exploration prior to the formation of NASA, see Roland (1985). For historical perspectives on space flight, see Emme (1977) and Siddiqi (2010).

Missions to outer space might be classified as involving: (1) space shuttle/other manned space flights involving travel to and from the International Space Station, maintenance of the Hubble Telescope, or independent earth orbit, (2) satellites launched into earth orbit for communication, scientific experimentation, and/or defense, or (3) unmanned spacecraft with a destination of the moon of earth/moons of other planets and/or the other planets of the solar system. In the last category, the missions can involve the actual landing on and exploration of planetary surfaces and/or atmospheres (e.g., see the discussion of the Cassini mission to Saturn by Savory and Saghi 1997).

Throughout this time, OR/MS techniques have been used by NASA and the space industry in general in the management and analysis of their space activities. It is especially interesting to compare the level of technological sophistication of space-project planning in 1962 with what accompanied Senator Glenn when he went back into space in October 1998. Applications over the years have included project management, forecasting, scheduling, cost estimating, optimization, simulation, and multi-objective decision analysis.

Applications

Motivation

Many of the activities, projects and programs associated with space exploration are characterized by (1) high degrees of risk, with respect to both human life and cost, (2) the use of advanced technology, some facets of which may not have even been developed at the onset of a mission, (3) the long time frame associated with many missions (e.g., a manned mission to Mars would require a time frame of 20–40 years), (4) organizational challenges resulting from required inputs of diverse public and private organizations, and (5) the multiple objectives in the areas of safety, cost, scheduling and performance which must be considered. These characteristics would imply that the modeling techniques associated with OR/MS are even more important for space exploration, than they are for earth-based endeavors. For example, the high cost of the Cassini-Huygens Mission to Saturn, descibed below, motivated the use of sophisticated models to test various scheduling policies for its hardware data control system in order to minimize risk of failure.

Applications Involving the Space Shuttle and Potential Future Systems

NASA's Space Transportation System consists of an orbiter and its three engines (i.e., the Space Shuttle), two solid rocket boosters, and an external fuel tank. The shuttle began operational flights in 1982 and was retired from service in 2011, after 135 launches, all from the Kennedy Space Center in Florida. Each flight of a Space Shuttle requires the scheduling of thousands of activities that have certain precedence relationships and require the use of various types of scarce resources. Scheduling these activities to meet criteria relating to time, cost, and quality is a complex process. Developing feasible schedules requires the use of sophisticated project management techniques, including project networks, heuristic scheduling rules, and. Paté-Cornell and Fischbeck (1994) used

a probabilistic risk analysis procedure to set priorities for the maintenance of the heat shield tiles of the space shuttle orbiter. They showed that implementation of the policy suggested by their procedure would result in a 70% reduction in the probability of a shuttle accident attributable to tile failure. Morris and White (1987) employed a SLAM II simulation model (Pritsker 1986) to analyze the operational support requirements of the space shuttle under a delivery payload scenario from the earth to the proposed space station. The model consists of three major modules: one each for ground-base operations, space station operations, and orbital operations. The main inputs to the model include the delivery requirements at the space station. The model can be used to help determine the delivery rate capability of the system, the support resources required, and the utilization of various system resources. Bell (1994) developed a heuristic approach for scheduling the training sessions in shuttle cockpit simulators.

Numerous new launch systems have been proposed. Kaylani et al. (2008) have developed GEM-FLO (A Generic Environment for Modeling Future Launch Operations), a generic simulation-based modeling approach for accurately predicting processing turnaround times and other performance measures in order to support key program decisions in the selection of a new launch system. The system assumes that any new launch vehicle would be composed of several major components; e.g., the major components of the STS are noted above. Each major component follows a generic conceptual flow process, which is integrated into a generic flow process for the launch vehicle as a whole. The model was validated with historical data from space shuttle operations.

Padula et al. (2006) address the problem of optimization of aerospace design under conditions of uncertainty. The specific applications addressed involved three that were undertaken at the NASA Langley Research Center: (1) impact dynamics for airframes, (2) transonic airfoil design for low drag, and (3) coupled aerodynamic and structures optimization for a 3-D wing.

Applications Involving the International Space Station

The International Space Station (ISS), the largest artificial satellite that has ever orbited earth, is a joint project involving participation of five different space agencies: NASA, the ESA, the Russian Federal Space Agency, the Japanese Aerospace Exploration Agency, and the Canadian Space Agency. Its on-orbit construction began in 1998, and it is scheduled for completion in 2011. The cosmonauts and astronauts who man the station conduct a variety of scientific experiments. The initial design process of ISS was obviously a complex one. For example, Quirk et al. (1989) addressed the problem of selecting one of two energy module alternatives (photovoltaic or solar dynamic) for the space station. A chance-constrained programming model was developed to select the system that minimized the expected cost, subject to the constraint that the probability that the net output would be less than or equal to some given net output would be no more than a prespecified value. The structure underlying the model is a stochastic Leontief system. Inputs to the model include subjective probability distributions of energy requirements associated with various activities, as given by Johnson Space Center engineers. Hence, the model accounted for the inherent uncertainties associated with each alternative. Groen et al. (2006) describe a probabilistic risk assessment for the ISS using a PC-based software package, the Quantitative Risk Assessment System (QRAS). The algorithms embodied in the package employ event sequence diagrams (ESDs) as opposed to event trees to model various scenarios; the authors note that the use of ESDs allow for better communication between mangers and engineers.

Applications Involving a Proposed Manned Mission to Mars

An important long range program for NASA is a manned mission to Mars. In this regard, Tavana (2004) examined three alternative mission architectures for such a mission: (1) split mission scenario (pre-deployment of mission assets to Mars, followed by the mission crew), (2) combo lander scenario (mission assets travel with the crew), and (3) dual scenario (a combination of the previous two scenarios). The approach employs the analytic hierarchy process (AHP), subjective probability assessments (attained from personnel at the Johnson Space Center), and the entropy method in order to consider the risks and benefits of the seven phases of such a mission: earth vicinity/departure, Mars transfer, Mars arrival, planetary surface, Mars vicinity/departure, Earth transfer, and Earth arrival.

In another study related to a manned mission to Mars, Chamitoff et al. (2005) discuss a software package, Planetary Resource Optimization and Mapping Tool (PROMT), which provides as output a global map indicating the relative value of various Martian landing sites. The software is illustrated through the use of data provided by the Mars global surveyor and the Odyssey Spacecraft. The paper notes that one important aspect in the evaluation of landing sites is the location of indigeneous resources that can be used by the mission.

Miscellaneous Applications Involving Scheduling

There are a myriad of applications of scheduling in space exploration activities, one of which has already been described above (Bell 1994). The papers discussed in the following paragraphs will describe applications involving the scheduling of activities for the Hubble Space Telescope, the scheduling of data transmissions from satellites to earth receiving stations, and the scheduling of a software simulator for the Cassini spacecraft.

Muscettola et al. (1992) described the Heuristic Scheduling Testbed System (HSTS) used for generating observation schedules for the Hubble Space Telescope (HST). The HST is a $1.4 billion observatory, with an expected operational lifetime of 15 years. It was placed into earth orbit in 1990. In any period of time, there are many different observational requests for the telescope's resources. Scheduling these requests is a difficult process for several reasons. A particular observation may require that several different operations be performed with six different scientific instruments that make up the HST. Several observations may be grouped together within a particular window of opportunity, depending on the locations of the telescope and the space objects to be observed. Parallel observations may be made with different viewing instruments; yet, not all six instruments may be turned on simultaneously because of energy constraints. The HSTS employs artificial intelligence procedures that provide a flexible approach to scheduling HST operations. This approach allows the "effective balancing of conflicting scheduling constraints and objectives."

Bell (1996) illustrated the practical use of a new approach to finding the dual prices in a Lagrangian relaxation problem. The practical application involves the transmission of data from satellites to receiving stations on earth. Such transmissions require a "line of sight" from the satellite to the station, and hence can only be accomplished at certain times of the day. Given the increasing number of satellites as compared to the number of receiving stations, this problem is becoming increasingly complex. The approach was illustrated with a case study involving 12 satellites and three receiving stations over a 1 week period.

Savory and Saghi (1997) developed a model to simulate various queue scheduling policies to improve the performance of a software simulator. The simulator was used to emulate the hardware data control system of the Cassini spacecraft, which was launched towards Saturn in 1997. The mission itself consists of an orbiter, which entered orbit around Saturn in 2004 and continues currently, and a probe which was the first man-made device to accomplish a landing in the outer solar system, on one of Saturn's moons, Titan. At the time of its launch, the Cassini was thought to be "the best instrumented planetary probe ever developed." Its cost (estimated at more than $3 billion) and importance mandated the use of extensive testing of its various systems, including its data control system. In order to perform these tests, scientists at NASA's Jet Propulsion Laboratory developed a computer program to represent the spacecraft's data control system. Implementation of the simulation results combined with a software code redesign resulted in greatly improved performance of the data control system.

Applications Involving Organizational Design, Project/Program Selection, or Project/Program Management

The organizational challenges associated with the planning, execution, and control of long range missions for space exploration are daunting. These challenges include coordinating the activities of numerous, geographically dispersed technical experts over an extended period of time. In this regard, Carroll et al. (2006) describe a study for developing a new organizational design tool for the NASA Systems Analysis Integrated Discipline Team (SAIDT). The study considered the various interfaces between the organization, activities, and the environment, and involved the use of various simulation tools.

Strategic (long range) planning for any organization which plans and conducts missions to outer space is

especially important (and difficult) for the reasons noted above. Decisions include which missions to undertake, which programs to fund, the amount of funding to give to each program, what designs to use for new spacecraft, etc. For example, a long-range planning problem encountered by NASA is discussed in Evans and Fairbairn (1989). This research addressed the problem of determining which missions, out of many possibilities, NASA should undertake during the next decades. A 0-1 integer linear programming model was formulated in which the decision variables determine whether or not to include a particular mission in NASA's long-range plan. The model allows for the consideration of several criteria relating to benefits derived in various areas (e.g., intellectual, humanistic, and utilitarian), as well as cost. In addition, the model implicitly considers the dependence among the various missions in the plan by specifying appropriate constraints. An example of dependence would be the fact that a manned mission to Mars would require the undertaking of several precursor missions. See the NASA Office of External Relations (1986) report and Paine (1991) for a discussion of potential missions to space during the twenty first century.

As implied in the previous paragraph, the concept of group decision making is important at NASA. Tavana (2003) developed a group decision-making tool called CROSS (consensus-ranking organizational support system) for evaluating and ranking advanced technology projects using the Analytic Hierarchy Process, subjective probabilities, and the entropy concept. The process includes decision makers and stakeholders and has three phases: an interaction phase, an integration phase, and an interpretation phase. The interaction phase involves identifying the stakeholders and gathering information from them on their criteria well as the probabilities of occurrence of each criterion for each project. Also in this phase, the decision makers use AHP to weight each of the stakeholder's respective departments. The integration phase calibration of results and the use of the maximum agreement heuristic in order to achieve a consensus ranking. Finally, in the interpretation phase, the decision makers make a final recommendation to management who makes the final decision. A case study involving ten projects, six stakeholder departments, and 38 criteria is used to illustrate the methodology.

Because of the high cost and wide variety of tasks associated with space flight, cost estimation, budget allocation, cost accounting and control are obviously important, and difficult, aspects in this area. Dillon et al. (2003) described the Advanced Programmatic Risk Analysis and Management (APRAM) model, a framework for allocating the budget for a program among its various dependent engineering projects. The process allows for the consideration of tradeoffs between technical failure risk and managerial failure risk. The process is illustrated with an application to the Mars Explorer Program.

Castillo et al. (1992) discussed GOST, a modeling system for cost estimation and mission planning. Berente and Youngjin (2010) discussed various issues associated with the implementation in 2004 of NASA's Full Cost, an activity-based accounting program. They noted that "some elements (termed dressage by control) of Full Cost . . .were geared towards satisfying disciplinary requirements without necessarily contributing towards productive activity." Their main conclusion was that the ultimate goal of full enterprise control is not attainable.

Bearden (2003) studied the relationships among risk, cost, and schedule for 45 low-cost small satellite, planetary missions over the period of 1990–1999, such as the Mars Polar Lander, the Mars Climate Orbiter, and the Mars Global Surveyor. One motivation for Bearden's efforts was the debate concerning NASA's Faster, Better, Cheaper (FBC) approach to satellite missions and the thought that, while the approach may have resulted in improvements to criteria related to cost and schedule, this was at the expense of a lower probability of mission success. As part of his efforts, Bearden developed a complexity index that could be derived for a specific mission as a function of its performance, mass, power, and technology choices. This complexity index was used to normalize development time and spacecraft cost across missions. The two main conclusions derived by Bearden through his study were (1) there is a clear dependence of success rate on system complexity, and (2) "that low-cost, planetary missions cost more, are developed faster, and fail more often than do Earth-orbiting missions".

Applications Involving Risk Analysis

Risk analysis is another important aspect of space exploration. Saunders et al. (2003), described

a process for estimating the likelihood of success of the missions associated with NASA's Explorer and Discovery Programs. Four evaluation criteria are considered by the process: scientific merit, feasibility of achieving the mission's scientific objectives, feasibility of the mission implementation approach, and social benefits associated with the mission.

Newman (2001) used a systems engineering approach to examine the causes of the failures associated with 50 space systems occurring from 1960 to 2000. About 70% of the cases examined were from 1990 to 2000. Newman noted that these failures can be classified according to whether or not the proximate (i.e., immediate) cause(s) are known. For example the proximate cause for the space shuttle Challenger failure was an o-ring failure, but that the proximate cause of the 1999 loss of the Mars Polar Lander is not known for certain. However, Newman notes that whether or not the proximate failure cause is known for certain, the basic purpose of any mission failure analysis is to emerge with something on which to act, and this requires an analysis which moves backwards from proximate cause(s) through multiple, possibly intersecting, paths to a variety of root causes.

See

- ▶ Analytic Hierarchy Process
- ▶ Chance-Constrained Programming
- ▶ Leontief Matrix
- ▶ Project Management
- ▶ Risk Assessment
- ▶ Risk Management for Software Engineering
- ▶ Scheduling and Sequencing
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Bearden, D. A. (2003). A complexity-based risk assessment of low-cost planetary missions: When is a mission too fast and too cheap? *Acta Astronautica, 52*, 371–379.

Bell, C. E. (1994). Weighted matching with vertex weights: An application to scheduling training sessions in NASA space shuttle cockpit simulators. *European Journal of Operational Research, 73*, 443–449.

Bell, C. E. (1996). Finding improving directions in lagrangean relaxation by fictitious play: A NASA scheduling application. *European Journal of Operational Research, 88*, 550–562.

Berente, N., & Youngjin, Y. (2010). Dressage, control, and enterprise systems: The case of NASA's full cost initiative. *European Journal of Information Systems, 19*, 21–34.

Carroll, T. N., Gormley, T. J., Bilardo, V. J., Burton, R. M., & Woodman, K. L. (2006). Designing a new organization at NASA: An organization design process using simulation. *Organization Science, 17*, 202–214.

Castillo, D. G., Dolk, D. R., & Kridel, D. J. (1992). GOST: An active modeling system for costing and planning NASA space programs. *Journal of Management Information Systems, 8*, 151–169.

Chamitoff, G., James, G., Barker, D., & Dershowitz, A. (2005). Martian resource locations: Identification and optimization. *Acta Astronautica, 56*, 755–769.

Clark, P. S. (2004). The development of China's piloted space program: From sounding rocket to Shenzou. *Journal of the British Interplanetary Space Society, 57*, 391–426.

Dillon, R. L., Pate-Cornell, M. E., & Guikema, S. D. (2003). Programmatic risk analysis for critical engineering systems under tight resource constraints. *Operations Research, 51*, 354–370.

Emme, E. M. (Ed.). (1977). *Two hundred years of flight in America: A bicentennial survey*. San Diego, CA: Univelt.

Evans, G. W., & Fairbairn, R. (1989). Selection and scheduling of advanced missions for NASA using 0-1 integer linear programming. *Journal of the Operational Research Society, 40*, 971–982.

Groen, F. J., Smidts, C., & Mosleh, A. (2006). QRAS – The quantitative risk assessment system. *Reliability Engineering & Systems Safety, 91*, 292–304.

Kaylani, A., Mollaghasemi, M., Cope, D., Fayez, S., Rabadi, G., & Steele, M. (2008). A generic environment for modelling future launch operations-GEM-FLO: A success story in generic modelling. *Journal of the Operational Research Society, 59*, 1312–1320.

Morris, W. D., & White, N. H. (1987). *Space transportation system operations model*. Hampton, VA: NASA Technical Memorandum, NASA Langley Research Center.

Muscettola, N., Smith, S. F., Cesta, A., & d'Aloisi, D. (1992). Coordinating space telescope operations in an integrated planning and scheduling architecture. *IEEE Transactions on Control Systems, 12*, 28–37.

NASA Office of External Relations (Ed.). (1986). *NASA space plans and scenarios to 2000 and beyond*. Park Ridge, NJ: Noyes Publications.

Newman, J. S. (2001). Failure-space: A systems engineering look at 50 space system failures. *Acta Astronautica, 48*, 517–527.

Padula, S. L., Gumbert, C. R., & Li, W. (2006). Aerospace applications of optimization under uncertainty. *Optimization and Engineering, 7*, 317–328.

Paine, T. O. (Ed.). (1991). *Leaving the cradle: Human exploration of space in the 21st century* (Science and technology series, Vol. 28). San Diego, CA: Univelt.

Paté-Cornell, M.-E., & Fischbeck, P. S. (1994). Risk management for the tiles of the space shuttle. *Interfaces, 24*, 64–86.

Pritsker, A. A. B. (1986). *Introduction to simulation and SLAM II*. West Lafayette, IN: Systems Publishing Corporation.

Quirk, J., Olson, M., Habib-Agahi, H., & Fox, G. (1989). Uncertainty and Leontief systems: An application to the selection of space station system designs. *Management Science, 35*, 585–596.

Roland, A. (1985). *Model research: The national advisory committee for aeronautics, 1915–1958. Vol. 1: The NASA history series*. Washington, DC: Scientific and Technical Information Branch, NASA.
Saunders, M., Richie, W., Rogers, J., & Moore, A. (2003). Prediciting mission success in small satellite missions. *Acta Astronautica, 52*, 361–370.
Savory, P. A., & Saghi, G. (1997). Simulating queue scheduling policies for a spacecraft simulator. *Interfaces, 27*, 1–8.
Siddiqi, A. A. (2010). Competing technologies, National(ist) narratives, and universal claims: Toward a global history of space exploration. *Technology and Culture, 51*, 425–443.
Swenson, L. S., Jr., Grimwood, J. M., & Alexander, C. C. (1966). *This new ocean: A history of project mercury*. Washington, DC: National Aeronautics and Space Administration.
Tavana, M. (2003). CROSS: A multicriteria group-decision-making model for evaluating and prioritizing advanced-technology projects at NASA. *Interfaces, 33*, 40–56.
Tavana, M. (2004). A subjective assessment of alternative mission architectures for the human exploration of mars at NASA using multicriteria decision making. *Computers and Operations Research, 31*, 1147–1164.

Spanning Tree

A subnetwork (graph) of a given network that connects all the nodes of the network and which has the property that once a path travels through a node, it cannot return to that node (the path has no cycles). A spanning tree is a tree of the network. If a network has n nodes, then the spanning tree has $n - 1$ arcs.

See

- ▶ Kruskal's Algorithm
- ▶ Minimum Spanning Tree Problem
- ▶ Network Optimization
- ▶ Primal-Dual Algorithm
- ▶ Prim's Algorithm
- ▶ Tree

Sparse Matrix

A matrix whose elements are mostly zero.

See

- ▶ Density
- ▶ Super-Sparsity

Sparsity

- ▶ Density
- ▶ Large-scale Systems
- ▶ Sparse Matrix
- ▶ Super-Sparsity

Special-Ordered Sets (SOS)

Types of constraints in optimization models. SOS of type 1 require that only one variable in the set may be nonzero; SOS of type 2 require that only two variables in the set may be nonzero and they must be adjacent. SOS of type 1 are used in problems in which the variables in the set are binary and only one of them can be equal to one (e.g., assignment of personnel). SOS of type 2 occur when transforming a separable-programming problem into an equivalent linear structure. Special computational approaches are used to simplify the handling of both types of SOS problems.

See

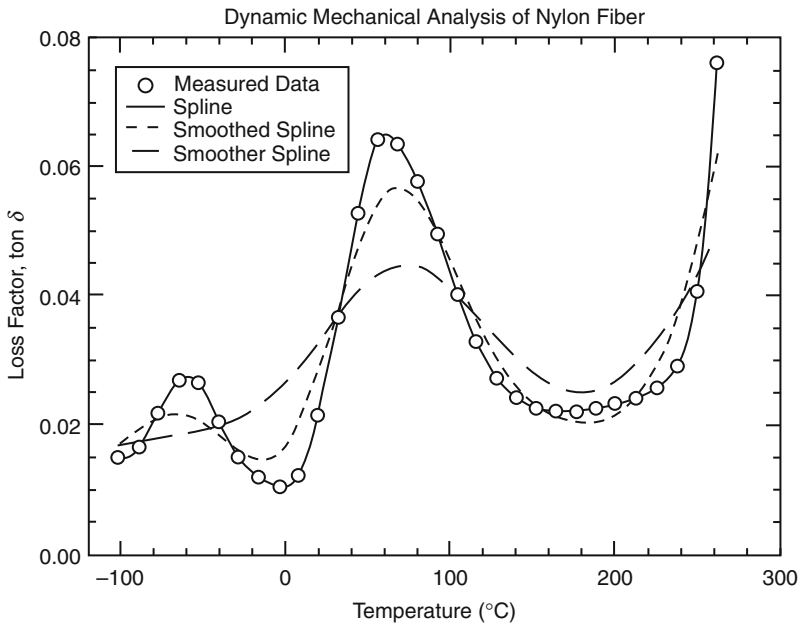
- ▶ Separable-Programming Problem

Splines

Sharon A. Johnson
Worcester Polytechnic Institute, Worcester, MA, USA

Introduction

Splines are an important class of mathematical functions used for approximation. A spline is a piecewise polynomial function that is commonly described as being "as smooth as it can be without reducing to a polynomial" (de Boor 2001). For example, the cubic spline shown as the solid line



Splines, Fig. 1 Spline approximations to data collected from a dynamic mechanical analysis of nylon fiber (courtesy of Monsanto Chemical Company)

in Fig. 1 is composed of individual cubic polynomials, each defined between two adjacent data points, such that the function values and first and second derivatives of adjoining polynomial pieces are the same. In general, a function defined on an interval $[a, b]$ is defined as a polynomial spline of degree k, having knots $t_1, \ldots, t_r$, if the following three conditions hold: (i) $a < t_1 < \ldots < t_r < b$, so the knots $t_1, \ldots, t_r$ partition the interval $[a, b]$ into $r + 1$ smaller subintervals, (ii) on each subinterval $[t_i, t_{i+1}]$, the spline is given by a polynomial function of at most degree k, and (iii) the spline and its derivatives up to order $k - 1$ are all continuous on $[a, b]$. The definition of splines is sometimes extended to allow the knots to be coincident (e.g., so that $t_i = t_{i+1}$), in which case the spline has less continuity at that knot (de Boor 2001). Recognition in the early 1960s that splines could mathematically model the physical process of drawing a smooth curve with a mechanical spline resulted in further investigation of their approximation and methods for computing them efficiently (Schumaker 2007).

The benefits of splines as approximating functions are discussed below, followed by an application to illustrate when spline approximation can be effective. The B-spline representation of a spline is then introduced. Finally, multivariate approximation is briefly addressed.

Splines as Approximating Functions

Scientists create mathematical models to describe a physical system or problem, then experiment with the model to draw conclusions about that system. Functions that relate decisions or independent variables to output or dependent variables form the basis of these models. While these underlying functions are sometimes known explicitly, they are often created by collecting discrete data about the system, then constructing an approximation to the unknown underlying response function. Suppose that the value of a function is measured at points $x_1, \ldots, x_n$, yielding values $f(x_1), \ldots, f(x_n)$. In interpolation problems, the goal is to find an approximating function $s(x)$ that passes through the points $f(x_1), \ldots, f(x_n)$, so that $s(x_i) = f(x_i)$. When the measured values $f(x_1), \ldots, f(x_n)$ contain errors, an approximating function $s(x)$ is created to balance the desire to obtain an approximation with smooth behavior and the desire to fit the data closely enough (Dierckx 1993). For example, the smoothing splines shown with dashed lines in Fig. 1 are constructed with a smoothing factor that allows deviations from the data (de Boor 2001); permitting larger deviations results in a cubic spline with a smaller second derivative. Broken curve regression can also be used (Seber and Wild 1989) to fit data with errors.

The data fitting problems described above are one major category of valuable approximation (Schumaker 2007). Other common approximation problems involve replacing a known function with one that is easy to compute, or estimating solutions to models involving differential equations, which can only be solved explicitly for simple cases.

Spline functions are effective approximations because they are relatively smooth, and splines of low degree usually provide an adequate fit with reasonable computational effort (Dierckx 1993). Every continuous function on an interval can be approximated arbitrarily well by a polynomial spline of a particular order, provided a sufficient number of knots are allowed. For example, adding more segments to a linear spline (a piecewise linear function) will better fit a curve. Low order splines are flexible and do not exhibit the oscillations usually associated with polynomials. The ease with which splines can be stored and evaluated on a computer makes them powerful for a variety of applications. Spline fitting routines are included in most general software libraries (e.g., the NAG subroutine libraries developed by the Numerical Algorithms Group, Inc., or the Spline Toolbox for MATLAB developed by Mathworks, Inc.).

An Application

The cost-to-go function $f_t(x_t)$ and optimal solution for a finite horizon dynamic programming problem can be found by solving the functional (optimality) equation:

$$f_t(x_t) = \max_{R_t} \{B_t(x_t, R_t) + E_{q_t}[f_{t+1}(x_{t+1})]\} \quad (1)$$

which in period t represents the expected benefit from operating the system from period t to the end of the horizon, given the system begins period t in state x_t. Such problems arise in managing inventory or planning water reservoir operations, where the state x_t would represent the amount in inventory or the volume of water in a reservoir. In stochastic problems, the system is subject to random influences q_t, such as uncertain demand or the inflow to a reservoir. In each period t, the decision r_t is made to maximize both the benefit B_t occurring from operation in the current period as well as expected future benefits $E_q[f_{t+1}]$. For example, the decision r_t might correspond to how much to produce or how much to release. The status of the system x_{t+1} at the end of each period t is determined by a transition function $g(x_t, q_t, r_t)$; for example, in inventory planning, g is $x_t + r_t - q_t$.

When the state vector x_t is continuous, the cost-to-go function f_t and policy r_t are often found by discretizing x_t and recursively solving (1) backward, for $t = T$ (the last time period) to $t = 1$. Because the function $f_{t+1}(x_{t+1})$ is only known at a finite number of points, interpolation can be used to generate a value when it is needed at other points. Using cubic splines to approximate the cost-to-go function can significantly reduce the effort required to solve such dynamic programming problems, particularly when the state vector is of high dimension, because of their accuracy relative to piecewise linear approximations (so fewer knots are needed) and because their smoothness allows efficient optimization methods to be used to find the decisions r_t (Johnson et al. 1993). Splines can also be applied to more general dynamic programming algorithms (Schweitzer and Seidmann 1985). They are also used extensively in computer-aided design and visualization, as well as in regression and statistical applications.

The B-Spline Representation

Any spline $s(x)$ of degree k can be written as a linear combination of B-splines $B_i(x)$:

$$s(x) = \sum_i a_i B_i(x),$$

where each B-spline $B_i(x)$ is a spline of degree k (de Boor 2001; Dierckx 1993). B-splines permit the efficient evaluation of a spline and its derivatives because they have local support, i.e., outside of a small range, they take the value of zero.

A particular spline is selected as an approximating function by choosing the degree k of the spline, the number and position of the knots t_i, and the coefficients a_i. Choosing the number and/or the position of the knots is often a matter of trial and error. Theory may suggest points in the data where the underlying model changes (Smith 1979). More knots should generally be placed in those regions where the underlying data change rapidly. Algorithms have been developed for some spline problems where the knot locations are

treated as parameters and optimized. Because the problem is nonlinear, the computational effort in these algorithms increases significantly as the number of knots increases (Dierckx 1993).

The conditions that determine the coefficients a_i of the spline depend on how closely the spline is expected to fit the data, the desired smoothness, and specified boundary conditions. In problems where the function underlying the data are known to have certain properties, such as convexity or monotonicity, it may be desirable to develop an approximation with the same properties constraining the coefficients a_i. Such shape-preserving approximations may also be beneficial because they prevent undesirable oscillations.

Multivariate Approximation

Tensor products are an efficient way to construct and evaluate multivariate approximations because they allow a multivariate problem to be solved as a series of one-dimensional problems (de Boor 2001). A bicubic spline constructed using tensor product methods would be a cubic spline in each coordinate direction. The major drawback of such splines is that they require the approximation domain to be a rectangle, or easily transformed to a rectangle (Dierckx 1993). In addition, constructing tensor product approximations is appropriate only when it makes sense to have preferred directions in the approximant. For example, a bicubic spline could efficiently approximate a peak that occurred along one axis. However, many knots would be required in each dimension if the peak occurred along a diagonal.

When tensor product methods are not appropriate, constructing multivariate spline approximations is much more complex and thus computationally less attractive. First, an appropriate partition of the data must be chosen. Next an appropriate set of basis functions (similar to B-splines) must be defined that permit efficient evaluation of the approximating function. Dierckx (1993) described two spline generalizations based on triangularizations of a surface. Chen et al. (1999) used multivariate adaptive regression splines to approximate functions in dynamic programming applications; computational effort is reduced by constructing the spline using discrete points determined by orthogonal array experimental designs.

See

- Computer Science and Operations Research Interfaces
- Dynamic Programming
- Numerical Analysis
- Regression Analysis

References

Chen, V. C., Ruppert, D., & Shoemaker, C. A. (1999). Applying experimental design and regression splines to high-dimensional continuous-state dynamic programming. *Operations Research, 47*, 38–53.

de Boor, C. (2001). *A practical guide to splines*. New York: Springer. Revised Edition (original edition 1978).

Dierckx, P. (1993). *Curve and surface fitting with splines*. New York: Oxford University Press.

Johnson, S. A., Stedinger, J. R., Shoemaker, C. A., Li, Y., & Tejada-Guibert, J. A. (1993). Numerical solution of continuous-state dynamic programs using linear and spline interpolation. *Operations Research, 41*, 484–500.

Schumaker, L. L. (2007). *Spline functions: Basic theory* (3rd ed.). New York: Wiley (paperback).

Schweitzer, P. J., & Seidmann, A. (1985). Generalized polynomial approximations in markovian decision processes. *Journal of Mathematical Analysis and Applications, 110*, 568–582.

Seber, G. A. F., & Wild, C. J. (1989). *Nonlinear regression*. New York: John Wiley.

Smith, P. L. (1979). Splines as a useful and convenient statistical tool. *The American Statistician, 33*, 57–62.

Sports

Shaul P. Ladany
Ben-Gurion University of the Negev, Beer Sheva, Israel

Introduction

The history of the applications of quantitative methods and systems analysis to sports events is very much the history of systems analysis and its applications to many fields of human endeavor. For a thorough review of all the sports applications up to 1976, see Ladany and Machol (1977), while for the second half of the same two-pronged effort which culminated in invited research articles of the mid 1970s,

see Machol et al. (1976). A further review, incorporating most of the later applications, can be found in Gerchak (1994). In 2005 the *Journal of Quantitative Analysis in Sports* was established, publishing a wide range of relevant O.R. articles. A Special Issue of *Sports Management*, containing a variety of new O.R. sports papers, was edited by Ladany (2006). Many of the summaries incorporated here are based on the overview by Cochran (2008).

The first studies of sports were purely descriptive; the earliest such technical articles, on cricket, by Elderton (1909, 1927, 1945) and Wood (1945), are described in Pollard (1977). The application of sophisticated statistical analysis started with Mosteller (1952), who estimated the probability that the better team wins in the baseball World Series competition. The next stage was Mottley's (1954) suggestion that operations research could be profitably applied to sports, specifying football and basketball examples. Bona fide optimization applications followed when Howard (1960) and Bellman (1964) applied dynamic programming to baseball.

In line with the beginning, most of the studies were applied to team sports, initially dealing with issues of individual teams, later moving to organizational matters preoccupying leagues and associations. The applications to individual sports have been much fewer.

Team Sports

Baseball – Most of the studies were applied to baseball, a sport particularly suitable for OR approaches because the action occurs in discrete events, and because the state of the game is simple to specify. The benefit of the strategy of intentional walk and base-stealing was thoroughly investigated by Lindsey (1959, 1961, 1963, 1977). The best batting order was analyzed by Cook and Garner (1964); Cook and Fink (1972); Freeze (1975) and Peterson (1977), by Bukiet et al. (1997) using a Markov chain method, and finally by Sokol (2003) using a robust heuristic. A team's elimination from playoff consideration was explored by Robinson (1991). Persistence of racial discrimination was discussed by Kolpin and Singell (1993) using a game-theoretic model, and Anderson and Sharp (1997) used Data Envelopment Analysis to create an alternative to the traditional batting statistics. Brimberg and Hurley (2004) dealt with a decision problem, final offer arbitration applied to Major League Baseball players salary negotiation was addressed by Greenstein et al. (2004) and by Hannany et al. (2007); investigated whether the pitcher or the batter control home plate. The popular book that was made into a movie, *Moneyball* (Lewis 2003), highlighted the use of quantitative statistical methods in baseball in evaluating the total value of players.

Basketball – Based on analysis of playing statistics, players are classified by Ghosh and Steckel (1993) as filling distinct roles, providing guidelines for selecting draft choices and executing trades. Sinuany-Stern et al. (2006) applied the analytic hierarchy process for the evaluation of basketball teams, while Kvam and Sokol (2006) and Brown and Sokol (2010) used a logistic Regression/Markov chain model for NCAA basketball prediction.

Cricket – The decision whether to run when the batsman is next to strike, was analyzed by Clarke and Norman (1998a, b) using dynamic programming. A method for setting revised target scores at a match, which has been forcibly shortened, is described by Duckworth and Lewis (1998; Carter and Guthrie (2004) analyzed fairness and incentive in limited overs cricket matches, Barr and Kantor (2004) suggested a criterion for comparing and selecting batsmen, while Scarf and Shi (2005) modeled match outcomes and the setting of final innings target.

Football – The value of field position was investigated by Carter (a former National Football League quarterback) and Machol (1971, 1978). The value of a tie and extra-point strategy was analyzed by Porter (1967) and Bierman (1968). For the Australian rules football Clarke and Norman (1998a, b) suggested a new strategy for defending teams, using a dynamic programming model, while Tomecko and Filar (1998) analyzed player assignments using an analytic hierarchy process. Morrison and Kalwani (1993) analyzed the ability of field goal kickers, Bilder and Loughin (1998) investigated the probability of success for placekicks, Brimberg et al. (1999) discussed the punt returner location problem, Hurley (1998) derived optimal sequential decisions, Brimberg and Hurley (2006) have shown that championships are won based on the ability of a team to run and to defend the run, while Rosen and Wilson (2007) analyzed the defense first strategy in overtime games.

Hockey – The problem when to pull the goalie was considered by Morrison (1976); Morrison and Wheat (1986); Erkut (1987); Nydic and Weiss (1989), and Washburn (1991). League playoff strategies were analyzed by Monahan and Berger (1977); Thomas (2007) analyzed the time between goals, Cochran and Blackstock (2009) applied the Pythagoras model of the win/loss percentage and goals scored to the National Hockey League (NHL), Brimberg and Hurley (2009a, b) investigated whether NHL referees represent Markovian behavior and questioned the importance of the first goal in an NHL game.

Soccer – Rivett (1975) modeled the attendance at soccer matches and suggested changes in the organization of clubs. Shikata (1977) tried to analyze the motions of ball and players in a four-dimensional space. Mehrez et al. (1987) evaluated a new point system for the soccer leagues, while Mehrez and Hu (1995) constructed predictors for outcomes in the league, and Mosheiov (1998) used an integer program to the generalized knapsack problem to find the optimal "Dream Team." Hirotsu and Wright (2002) used a Markov model for optimal timing of substitution of players, and (2003) determined the best strategy for changing the configuration of a team. Hope (2003) investigated the conditions for firing a team manager, Wright and Hirotsu (2003) discussed the tactics and deterrents of fouls, McHale and Scarf (2007) modeled soccer matches, while Scarf and Shi (2008) measured the importance of a match in a tournament. In a sequence of articles, with psychological orientations, Bar-Eli et al. (2007) and Azar and Bar-Eli (2008) analyze the game between the goal keeper and the penalty kicker, from the view point of the goal keeper, while Bar-Eli and Azar (2009) treat it from the point of view of the kicker, the irrationality of the performances of the kickers and the goal keepers is investigated by Bar-Eli et al. (2009), and in Azar and Bar-Eli (2011) mixed-strategy Nash equilibrium model is used to predict the results of penalty kicks.

General League Issues – The problem of planning the schedule of the games for the entire season with the objective to minimize traveling distance and/or number of tours, under various constraints, was attacked – for various leagues – by Campbell and Chen (1976); Ball and Webster (1977); Bean and Birge (1980); Schreuder (1980, 1992); Ferland and Fleurent (1991); Russell and Leung (1994); Wright (1994); Kostuk (1997) developed a decision support system for elimination tournament scheduling. Andreu and Corominas (1989) scheduled the Olympic Games, Armstrong and Willis (1993) scheduled the Cricket World Cup, Costa (1995) used a tabu search algorithm to schedule the National Hockey League, Nemhauser and Trick (1998), Heinz (2001), and Voorhis (2002) dealt with college basketball scheduling, and Kendall (2007) scheduled an English soccer league over holiday periods. The related counterpart problem of scheduling umpires was considered by Evans (1988); Wright (1991), and Farmer et al. (2007) scheduled umpire crews for tennis tournaments. The optimal realignment of the teams in the National Football League to minimize total intradivisional travel is discussed by Saltzman and Bradford (1996) and Smith et al. (2006) optimized team travel in a basketball tournament. Horen and Riezman (1985) dealt with drawing methods for single elimination tournaments, Clarke and Allsiopp (2001) attempted scheduling cricket tournaments fairly, Fleurent and Ferland (1993) dealt with allocating games in the National Hockey League, analyzed the effects of home-away sequencing on the length of best-of-seven game playoff series, Smith et al. (2006) used bracket assignments for basketball and baseball tournaments, while Briskoin (2008) summarized the sports leagues scheduling models in a book.

Draft issues prevailing in North American professional sports were the subject of investigations by Price and Rao (1976); Brams and Straffin (1979).

The presence of streaks in baseball was analyzed and rejected by Tversky and Gilovich (1989), as well as by Albright (1992). The advantage of splitting a league into smaller leagues to lead to more pennant races was studied by Winston and Soni (1982). Finding the optimal location of a new arena using an analytic hierarchy process is discussed by Carlsson and Walden (1995).

The problem of ranking teams (and applicable also to individuals) has drawn considerable attention by researchers. Leake (1976) used electrical network theory, to rank football teams, while Wilson (1995) used a neural network approach. Ushakov (1976) presented a methodology for ranking participants playing in a round robin tournament like in chess. Applying the analytic hierarchy approach to predict the true strength of sport teams was discussed by Sinuany-Stern (1988) and by Takahashi (1990).

The effect of choosing different point values in ranked voting systems with scoring was analyzed, using a stochastic dominance analysis, by Stein et al. (1994). A system for selecting the best among a set of competing players or teams, that minimizes the number of rounds, was developed by Adler et al. (1994). Finally, Stefani (1999) provided a general taxonomy of sports rating systems.

The playing order to avoid dead finals and the calculation of premiership odds for the Australian Football League's final eight playoff was discussed by Clarke (1996), while the problem of fair assignment of season tickets was solved with mixed integer programming by Grandine (1998).

Evaluation of individual player performances were discussed by Fry et al. (2009); Terpstra and Schauer (2007), and by Alamar and Weinstein-Gould (2008), while entire teams were evaluated by Rosner (1976); Coleman and Lynch (2001); Martinich (2002); Horowitz (2004); Cassady et al. (2005); Coleman (2005); Baker and Scarf (2006), and by Rump (2008).

Individual Sports

Track and Field – The derivation of optimal training plans for pentathlon (applicable also to decathlon, triathlon, biathlon, etc.) was pioneered by Ladany (1975b) using linear programming with physiological constraints. Whether Bob Beamon's miracle jump in Mexico City was affected by the altitude was first analyzed by Brearley (1972). Related jump decision problems, such as aiming at take off, were studied by Ladany et al. (1975); Sphicas and Ladany (1976); Ladany and Singh (1978), and Mehrez and Ladany (1987). The tactical issues in pole-vaulting (which are similar to those prevailing in high-jump) for the selection of the optimal starting height were investigated by Ladany (1975a), and after change in the rules reinvestigated by Hersh and Ladany (1989) using dynamic programming. The sequential and competitive nature of several athletic events led to the coinage of the term games of boldness and to their analysis by Gerchak and Henig (1986); Henig and O'Neill (1992), and Gerchak and Kilgour (1992). Optimal assignments of runners (or swimmers) to relay teams, were put forward by Machol (1970) and Heffley (1977), advancing from the use of the simple deterministic assignment model to conditional and stochastic treatments. Strategy in fell running was analyzed by Hayes and Norman (1994); Friedman et al. (2006) and Mizrahi et al. (2006) developed a set of models for determining and analyzing the optimal threshold in athletic games.

Brimberg et al. (2006) analyzed the optimal allocation of effort among the stages in the triple jump, while Gerchack (2000) proposed a method applicable to decathlon and pentathlon scoring tables – for athletes rewards based on difficulty of achievements.

Optimization of the biomechanical aspects were investigated in various fields. The optimal angle to release a shot put or a hammer were discussed by Townend (1984). The influence of slope gradient on running uphill and the change of the optimal strategy with the slope were discussed by Davey et al. (1995).

Golf – The evaluation of the handicap system and its fairness occupied all researchers in the field, starting with Scheid (1972), and followed by Pollock (1974, 1977). Handicapping was applied also to other sports events; Camni and Grogan (1988) applied it to road-running races using frontier analysis.

Levy (1976) estimated a golfer's tournament score, while Hurley (2002) investigated the impact of the order of the golfers on the final day of the Ryder Cup matches.

Tennis – Analysis of the most important points was performed by Morris (1977); Gale (1971) investigated the optimal serving strategies, and justified the greater risk taken on the first serve. Norman (1985) applied dynamic programming to determine when to use a fast serve. Blackman and Casey (1980) suggested a player rating system.

Other Sports – Selection of teams for gymnastic competition was dealt with bivalent integer programming by Ellis and by Corn (1984) and by Eilon (1986). Optimal weight-lifting policies were derived by Lilien (1976). Oar arrangements in rowing eights to prevent fish-tail behavior were analyzed using the mechanical theory of moments by Brearley (1977). The unfairness of the existing scoring systems for jai-alai was evaluated using simulation by Hannan and by Smith (1981) and by Skiena (1988); Henig and O'Neill (1992) considered games where the player performing the hardest task wins, Larkey et al. (1997) investigated the importance of skill in games, Beis et al. (2006) described the use of O.R. to manage the 2006 Athens Olympic Games, Percy (2007)

analyzed the badminton scoring system, Scarf (2007) discussed the choice of the route in mountaineering, Scarf and Grehan (2005) evaluated the route choice in cycling.

See

- Analytic Hierarchy Process
- Decision Analysis
- Dynamic Programming
- Integer and Combinatorial Optimization
- Linear Programming
- Simulation of Stochastic Discrete-Event Systems
- Systems Analysis

References

Adler, M., Gemmell, P., Harchol, B. M., Karp, R. M., & Kenyon, C. (1994). Selection in the presence of noise: The design of playoff systems. In *Proceedings of the fifth ACM-SIAM Symposium on Discrete Algorithms* (pp. 564–572). New York: ACM.

Alamar, B. C., & Weinstein-Gould, J. (2008). Isolating the effect of individual linemen on the passing game in the national football league, *Journal of Quantitative Analysis in Sports*, *4*(2). paper 10.

Albright, C. (1992). Streaks & slumps. *OR/MS Today, 19*(2), 94–95.

Anderson, T. R., & Sharp, G. P. (1997). A new measure of baseball batters using DEA. *Operations Research, 73*, 141–155.

Andreu, R., & Caraminas, A. (1989). SUCCESS 92: A DSS for scheduling the Olympic Games. *Interfaces, 19*(1), 1–12.

Armstrong, J., & Willis, R. J. (1993). Scheduling the cricket world cup – A case-study. *Journal of the Operational Research Society, 44*, 1067–1072.

Azar, O. H., & Bar-Eli, M. (2008). Biased decisions of professional soccer players: Do goalkeepers dive too much during penalty kicks? In P. Andersson, P. Ayton, & C. Schmidt (Eds.), *Myths and facts about football: The economics and psychology of the world's greatest sport* (pp. 93–111). Newcastle upon Tyne, UK: Cambridge Scholars Publishing.

Azar, O. H., & Bar-Eli, M. (2011). Do soccer players play the mixed-strategy Nash equilibrium? *Applied Economics, 43*(25), 3591–3601.

Baker, R., & Scarf, P. A. (2006). Modelling the outcomes of annual sporting contests. *Journal of the Royal Statistical Society, Series C, 55*, 225–239.

Ball, B. C., & Webster, D. B. (1977). Optimal scheduling for even-numbered team athletic conferences. *IIE Transactions, 9*, 161–167.

Bar-Eli, M., & Azar, O. H. (2009). Penalty kicks in soccer: An empirical analysis of shooting strategies and Goalkeepers' preferences. *Soccer & Society, 10*(2), 183–191.

Bar-Eli, M., Azar, O. H., & Lurie, Y. (2009). (Ir) rationality in action: Do soccer players and goalkeepers fail to learn how to best perform during a penalty kick. *Progress in Brain Research, 174*, 97–108.

Bar-Eli, M., Azar, O. H., Ritov, I., Keidar-Levin, Y., & Shein, G. (2007). Action bias among elite soccer goalkeepers: The case of penalty kicks. *Economic Psychology, 28*, 606–621.

Barr, G. D. I., & Kantor, B. S. (2004). A criterion for comparing and selecting batsmen in limited overs cricket. *Journal of the Operational Research Society, 55*, 1266–1274.

Bean, J. C., & Birge, J. R. (1980). Reducing traveling costs and player fatigue in the national basketball association. *Interfaces, 10*(3), 98–102.

Beis, D. A., Loucopoulos, P., Pygriotis, Y., & Zografos, K. G. (2006). PLATO helps Athens win gold: Olympic Games knowledge modelling for organizational change and resource management. *Interfaces, 36*, 26–42.

Bellman, R. E. (1964). Dynamic programming and Markovian decision processes with particular application to baseball and chess, Ch. 7. In E. Beckenbach (Ed.), *Applied combinatorial mathematics*. New York: Wiley.

Bierman, H. (1968). A letter to the editor. *Management Science, 14*, B281–B282.

Bilder, C. R., & Loughin, T. M. (1998). It's good! An analytic analysis of the probability of success for placekicks. *Chance, 11*(2), 20–30.

Blackman, S. S., & Casey, J. W. (1980). Developing of a rating system for all tennis players. *Operations Research, 28*, 489–502.

Brams, S. J., & Straffin, P. D., Jr. (1979). Prisoner's dilemma and professional sports drafts. *The American Mathematical Monthly, 86*, 80–88.

Brearley, M. N. (1977). Oar arrangements in rowing eights. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 184–185). Amsterdam: North-Holland.

Brearly, M. N. (1972). The long jump miracle of Mexico City. *Mathematics Magazine, 45*, 241–246.

Brimberg, J., & Hurley, W. (2004). A baseball decision problem. *INFORMS Transactions on Education, 5*, 1.

Brimberg, J., & Hurley, W. (2006). Strategic considerations in coaching of North American football. In Ladany, S. P. (Ed.), *Sport Management* special issue. *Sport Management and Marketing*, *1*(3), 279–287.

Brimberg, J., & Hurley, W. J. (2009a). Are national hockey league referees Markov? *OR Insight, 22*(4), 234–243.

Brimberg, J., & Hurley, W. J. (2009b). A note on the importance of the first goal in a national hockey league game. *International Journal of Operational Research, 6*(2), 282–287.

Brimberg, J., Hurley, W., & Johnson, R. E. (1999). A punt returner location problem. *Operations Research, 47*(3), 482–487.

Brimberg, J., Hurley, B., & Ladany, S. P. (2006). An operations research approach to the triple jump. In Ladany, S. P. (Ed.), *Sport Management* special issue. *Sport Management and Marketing*, *1*(3), 208–214.

Briskorn, D. (2008). *Sports leagues scheduling models, combinatorial properties and optimization algorithms*. Berlin: Springer.

Brown, M., & Sokol, J. (2010). An improved LRMC method for NCAA basketball prediction. *Journal of Quantitative Analysis in Sports, 6*, 3.

Bukiet, B., Harold, E. R., & Palacios, J. L. (1997). A Markov chain approach to baseball. *Operations Research, 45*, 14–23.

Camm, J. D., & Grogan, T. J. (1988). An application of frontier analysis: Handicapping running races. *Interfaces, 18*(6), 52–60.

Campbell, R. T., & Chen, D. S. (1976). A minimum distance basketball scheduling problem. In R. E. Machol, S. P. Ladany, & D. G. Morrison (Eds.), *Management science in sport* (TIMS studies in the management sciences, Vol. 4, pp. 15–26). Amsterdam: North-Holland.

Carlsson, C., & Walden, P. (1995). AHP in political group decisions: A study in the art of possibilities. *Interfaces, 25*(4), 14–29.

Carter, M., & Guthrie, G. (2004). Cricket interruptions: Fairness and incentive in limited overs cricket matches. *Journal of the Operational Research Society, 55*, 822–829.

Carter, V., & Machol, R. E. (1971). Operations research in football. *Operations Research, 19*, 541–544.

Carter, V., & Machol, R. E. (1978). Optimal strategies on fourth down. *Management Science, 24*, 1758–1762.

Cassady, C. R., Maillart, L. M., & Salman, S. (2005). Ranking sport teams: A customizable quadratic assignment approach. *Interfaces, 35*, 497–510.

Clarke, S. R. (1996). Calculating premiership odds by computer: An analysis of the AFL final eight playoff system. *Operational Research, 13*, 89–104.

Clarke, S. R., & Allsopp, P. (2001). Fair measures of performance: The world cup of cricket. *Journal of the Operational Research Society, 52*, 471–479.

Clarke, S. R., & Norman, J. M. (1998a). Dynamic programming in cricket: Protecting the weaker batsman. *Operational Research, 15*, 93–108.

Clarke, S. R., & Norman, J. M. (1998b). When to rush a Behind' in Australian rules football: A dynamic programming approach. *Operational Research Society, 49*, 530–536.

Cochran, J. J. (2008). Operations research and sports. *StatOR, 8*(2), 1–13.

Cochran, J. J., & Blackstock, R. (2009). Pythagoras and the National Hockey League. *Journal of Quantitative Analysis in Sports*, *5*(2), Art. 11.

Coleman, B. J. (2005). Minimizing game score violations in college football rankings. *Interfaces, 35*(6), 483–496.

Coleman, B. J., & Lynch, A. K. (2001). Identifying the NCAA tournament 'dance card'. *Interfaces, 31*, 76–86.

Cook, E., & Fink, D. L. (1972). *Percentage baseball and the computer*. Baltimore: Waverly Press.

Cook, E., & Garner, W. R. (1964). *Percentage baseball*. Cambridge, MA: MIT Press.

Costa, B. J. (1995). An evolutionary tabu search algorithm and the NHL scheduling problem. *INFOR, 33*, 161–178.

Davey, R. C., Hayes, M., & Norman, J. M. (1995). Speed, gradient and workrate in uphill running. *Journal of the Operational Research Society, 46*, 43–49.

Duckworth, F. C., & Lewis, A. J. (1998). A fair method for resetting the target in interrupted one-day cricket matches. *Journal of the Operational Research Society, 49*, 220–227.

Eilon, S. (1986). Note: Further gymnastics. *Interfaces, 16*(2), 69–71.

Elderton, W. P. (1927). *Frequency curves and correlation* (2nd ed.). London: Layton.

Elderton, W. P. (1945). Cricket scores and some skew correlation distributions. *Journal of the Royal Statistical Society A, 108*, 1–11.

Elderton, W. P., & Elderton, E. M. (1909). *Primer of statistics*. London: Black.

Ellis, P. M., & Corn, R. W. (1984). Using bivalent integer programming to select teams for intercollegiate women's gymnastics competition. *Interfaces, 14*(3), 41–46.

Erkut, E. (1987). More on Morrison and Wheat's 'pulling the goalie revisited'. *Interfaces, 17*(5), 121–123.

Evans, J. R. (1988). A microcomputer-based decision support system for scheduling umpires in the American baseball league. *Interfaces, 18*(6), 42–51.

Farmer, A., Smith, J. S., & Miller, L. T. (2007). Scheduling umpire crews for professional tennis tournaments. *Interfaces, 37*, 187–196.

Ferland, J. A., & Fleurent, C. (1991). Computer aided scheduling for a sports league. *INFOR, 29*, 14–24.

Fleurent, C., & Ferland, J. A. (1993). Allocating games for the NHL using integer programming. *Operations Research, 41*, 649–654.

Freeze, A. R. (1975). Monte Carlo analysis of baseball batting order. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 63–67). Amsterdam: North-Holland.

Friedman, L., Sinuany-Stern, Z., & Mehrez, A. (2006). Optimal thresholds in symmetric multi-stage multi-players competitions. In Ladany, S. P. (Ed.), *Sports Management* special issue. *Sports Management and Marketing*, *1*(3), 239–254.

Fry, M. J., Lundberg, A. W., & Ohlmann, J. W. (2009). A player selection heuristic for a sports league draft. *Journal of Quantitative Analysis in Sports*, *3*(2), paper 5.

Gale, D. (1971). Optimal strategy for serving in tennis. *Mathematics Magazine, 44*, 197–199.

Gerchak, Y. (1994). Operations research in sports. In S. M. Pollock et al. (Eds.), *Handbooks in OR & MS* (Vol. 6, pp. 507–527). Amsterdam: Elsevier Science.

Gerchak, Y. (2000). On the 'proper' relative size of prizes in competitions. *Chance, 13*(1), 38–44.

Gerchak, Y., & Henig, M. (1986). The basketball shootout: Strategy and winning probabilities. *Operations Research Letters, 5*, 241–244.

Gerchak, Y., & Kilgour, M. (1992). Sequential competitions with nondecreasing levels of difficulty. *Operations Research Letters, 13*, 49–58.

Gerchak, Y., Mausser, H. E., & Magazine, M. J. (1995). The evolution of draft lotteries in professional sports: Back to moral hazard? *Interfaces, 25*(6), 30–38.

Ghosh, A., & Steckel, J. H. (1993). Roles in the NBA: There's always room for a big man, but his role has changed. *Interfaces, 23*(4), 43–55.

Grandine, A. T. (1998). Assigning season tickets fairly. *Interfaces, 28*(4), 15–20.

Greenstein, E., Weissman, I., & Gerchak, Y. (2004). Estimating arbitrator's hidden judgement in final offer arbitration. *Group Decision and Negotiation, 13*, 291–298.

Hanany, E., Kilgour, D. M., & Gerchak, Y. (2007). How the prospect of final-offer arbitration affects bargaining. *Management Science, 53*, 1785–1792.

Hannan, E. L., & Smith, L. A. (1981). A simulation of the effects of alternative rule systems for Jai Alai. *Decision Sciences, 12*, 75–84.

Hayes, M., & Norman, J. M. (1994). Strategy in fell running: An analysis of the Bob Graham round. *Operational Research Society, 45*, 1123–1130.

Heffley, D. R. (1977). Assigning runners to a relay team. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 169–171). Amsterdam: North-Holland.

Henig, M., & O'Neill, B. (1992). Games of boldness, where the player performing the hardest task wins. *Operations Research, 40*, 76–87.

Henz, M. (2001). Scheduling a major college basketball conference – Revisited. *Operations Research, 49*(1), 163–168.

Hersh, M., & Ladany, S. P. (1989). Optimal pole-vaulting strategy. *Operations Research, 37*, 172–175.

Hirotsu, N., & Wright, M. (2002). Using a Markov process model of an association football match to determine the optimal timing of substitution and tactical decisions. *Journal of the Operational Research Society, 53*, 88–96.

Hirotsu, N., & Wright, M. (2003). Determining the best strategy for changing the configuration of a football team. *Journal of the Operational Research Society, 54*, 878–887.

Hope, C. (2003). When should you sack a football manager? Results from a simple model applied to the English premiership. *Journal of the Operational Research Society, 54*, 1167–1176.

Horen, J., & Riezman, R. (1985). Comparing draws for single elimination tournaments. *Operations Research, 33*(2), 249–262.

Horowitz, I. (2004). Aggregating expert ratings using preference-neutral weights: The case of the college football polls. *Interfaces, 34*(4), 314–322.

Howard, A. (1960). *Dynamic programming and Markov processes*. New York: MIT Press/Wiley.

Hurley, W. J. (1998). Optimal sequential decisions and the content of the fourth and goal conference. *Interfaces, 22*(6), 19–22.

Hurley, W. J. (2002). How should team captains order golfers on the final day of the Ryder Cup matches. *Interfaces, 32*(2), 74–77.

Kendall, G. (2007). Scheduling English football fixtures over holiday periods. *Journal of the Operational Research Society, 59*, 743–755.

Kolpin, V., & Singell, L. D., Jr. (1993). Strategic behavior and the persistence of the discrimination in professional baseball. *Mathematical Social Sciences, 26*, 299–315.

Kostuk, K. J. (1997). A decision support system for a large, multi-event tournament. *INFOR, 35*, 183–196.

Kvam, P., & Sokol, J. S. (2006). A logistic regression/Markov chain model for NCAA basketball. *Naval Research Logistics, 53*, 788–803.

Ladany, S. P. (1975a). Optimal starting height for pole-vaulting. *Operations Research, 23*(5), 968–978.

Ladany, S. P. (1975b). Optimization of pentathlon training plans. *Management Science, 21*, 1144–1155.

Ladany, S. P. (Ed.) (2006). "Sport Management" special issue. *Sport Management and Marketing*, *1*(3), 191–287.

Ladany, S. P., Humes, J. W., & Sphicas, G. P. (1975). The optimal aiming line. *Operational Research Quarterly, 26*(3), 495–506.

Ladany, S. P., & Levi, O. (2010, March 23–24) *Optimal routes in sailing competitions*, 16th Industrial Engineering and Management Conference, ORTRA, Tel-Aviv.

Ladany, S. P., & Machol, R. E. (Eds.). (1977). *Optimal strategies in sports*. Amsterdam: North-Holland.

Ladany, S. P., & Singh, J. (1978). On maximizing the probability of jumping over a ditch. *SIAM Review, 20*, 171–177.

Larkey, P., Kadane, J. B., Austin, R., & Zamir, S. (1997). Skill in games. *Management Science, 43*(5), 596–609.

Leake, R. J. (1976). A method of ranking teams: With an application to college football. In R. E. Machol, S. P. Ladany, & D. G. Morrison (Eds.), *Management science in sports* (TIMS studies in the management sciences, Vol. 4, pp. 27–46). Amsterdam: North-Holland.

Levy, F. K. (1976). Anti-trust and the links – Estimating a golfer's tournament score. *Interfaces, 6*(3), 5–17.

Lewis, M. (2003). *Moneyball: The art of winning an unfair game*. New York: W.W. Norton & Company.

Lilien, G. L. (1976). Optimal weightlifting. In R. E. Machol, S. P. Ladany, & D. G. Morrison (Eds.), *Management science in sports* (TIMS studies in the management sciences, Vol. 4, pp. 101–112). Amsterdam: North-Holland.

Lindsey, G. R. (1959). Statistical data useful for the operation of a baseball team. *Operations Research, 7*, 197–207.

Lindsey, G. R. (1961). The progress of the score during a baseball game. *Journal of the American Statistical Association, 56*, 703–728.

Lindsey, G. R. (1963). An investigation of strategies in baseball. *Operations Research, 11*, 477–501.

Lindsey, G. R. (1977). A scientific approach to strategy in baseball. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 169–171). Amsterdam: North-Holland.

Machol, R. E. (1970). An application of the assignment problem. *Operations Research, 18*, 745–746.

Machol, R. E., Ladany, S. P., & Morrison, D. G. (Eds.). (1976). *Management science in sports* (TIMS studies in the management sciences, Vol. 4). Amsterdam: North-Holland.

Martinich, J. (2002). College football rankings: Do the computers know best? *Interfaces, 32*(5), 85–94.

Mchale, I. G., & Scarf, P. A. (2007). Modelling soccer matches using bivariate discrete distributions with general dependence structure. *Statistica Nederlandica, 61*, 432–445.

Mehrez, A., & Hu, M. Y. (1995). Predictors of outcomes on a soccer game–a normative analysis illustrated for the Israeli soccer league. *Mathematical Methods of Operations Research, 42*, 361–372.

Mehrez, A., & Ladany, S. P. (1987). The utility model for evaluation of optimal behavior of a long jump competitor. *Simulation & Games, 18*, 344–359.

Mehrez, A., Pliskin, J. S., & Mercer, A. (1987). A new point system for soccer leagues: Have expectations been realized? *European Journal of Operational Research, 28*, 154–157.

Mizrahi, S., Mehrez, A., & Friedman, L. (2006). Game theory and sport sciences: Setting an optimal threshold level in spot competitions. In S. P. Ladany (Ed.), *Sport Management* special issue. *Sport Management and Marketing*, *1*(3), 255–262.

Monahan, J. P., & Berger, P. D. (1977). Playoff structures in the national hockey league. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 123–128). Amsterdam: North-Holland.

Morris, C. (1977). The most important points in tennis. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 131–140). Amsterdam: North-Holland.

Morrison, D. G. (1976). On the optimal time to pull the goalie: A Poisson model applied to a common strategy used in ice hockey. In R. E. Machol, S. P. Ladany, & D. G. Morrison (Eds.), *Management science in sports* (TIMS studies in the management sciences, Vol. 4, pp. 137–144). Amsterdam: North-Holland.

Morrison, D. G., & Kalwani, M. U. (1993). The best NFL field goal kickers: Are they lucky or good. *Chance, 6*(3), 30–37.

Morrison, D. G., & Wheat, R. D. (1986). Pulling the goalie revisited. *Interfaces, 16*, 28–34.

Mosheiov, G. (1998). The solution of the soccer 'dream League' game. *Mathematical and Computer Modelling, 27*, 79–83.

Mosteller, F. (1952). The world series competition. *Journal of the American Statistical Association, 47*, 355–380.

Mottley, M. (1954). The application of operations research methods to athletic games. *Japan Overseas Rolling Stock Association, 2*, 335–338.

Nemhauser, G. L., & Trick, M. A. (1998). Scheduling a major college basketball conference. *Operations Research, 46*(1), 1–8.

Norman, J. M. (1985). Dynamic programming in tennis: When to use a fast serve. *Journal of the Operational Research Society, 36*, 75–77.

Nydic, R. L., Jr., & Weiss, H. J. (1989). More on Erkut's 'more on Morrison and Wheat's pulling the goalie revisited'. *Interfaces, 19*, 45–48.

Percy, D. F. (2007). A mathematical analysis of badminton scoring systems. *Journal of the Operational Research Society, 60*, 63–71.

Peterson, A. V., Jr. (1977). Comparing the run-scoring abilities of two different batting orders: Results of a simulation. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 86–88). Amsterdam: North-Holland.

Pollard, R. (1977). Cricket and statistics. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 129–130). Amsterdam: North-Holland.

Pollock, S. M. (1974). A model for evaluating golf handicapping. *Operations Research, 22*, 1040–1050.

Pollock, S. M. (1977). A model of the USGA handicap system and 'fairness' of medal and match play. In S. P. Ladany & R. E. Machol (Eds.), *Optimal strategies in sports* (pp. 141–150). Amsterdam: North-Holland.

Porter, R. C. (1967). Extra-point strategy in football. *The American Statistician, 21*, 14–15.

Price, B., & Rao, A. G. (1976). Alternative rules for drafting in professional sports. In R. E. Machol, S. P. Ladany, & D. G. Morrison (Eds.), *Management science in sports* (TIMS studies in the management sciences, Vol. 4, pp. 79–90). Amsterdam: North-Holland.

Rivett, B. H. (1975). The structure of league football. *Operational Research Quarterly, 26*, 801–812.

Robinson, L. W. (1991). Baseball playoff eliminations: An application of linear programming. *Operations Research Letters, 10*, 67–74.

Rosen, P. A., & Wilson, R. L. (2007). An analysis of the defense first strategy in college football overtime games. *Journal of Quantitative Analysis in Sports, 3*, 2.

Rosner, B. (1976). An analysis of professional football scores. In R. E. Machol, S. P. Ladany, & D. G. Morrison (Eds.), *Management science in sports* (pp. 67–78). Amsterdam: North-Holland.

Rump, C. M. (2008). Data clustering for fitting parameters of a Markov chain model of multi-game playoff series. *Journal of Quantitative Analysis in Sports, 4*, 1.

Russell, R. A., & Leung, J. M. Y. (1994). Devising a cost effective schedule for a baseball league. *Operations Research, 42*, 614–625.

Saltzman, R. M., & Bradford, R. M. (1996). Optimal realignments of the teams in the national football league. *Operational Research, 93*, 469–475.

Scarf, P. A. (2007). Route choice in mountain navigation, Naismith's rule and the equivalence of distance and climb. *Journal of Sports Sciences, 25*, 719–726.

Scarf, P. A., & Greehan, P. (2005). An empirical basis for route choice in cycling. *Journal of Sports Sciences, 23*, 919–925.

Scarf, P. A., & Shi, X. (2005). Modelling match outcomes and decision support for setting a final innings target in test cricket. *IMA Journal of Management Mathematics, 16*, 161–178.

Scarf, P. A., & Shi, X. (2008). Measuring the importance of a match in a tournament. *Computers and Operations Research, 35*, 2406–2418.

Scheid, F. (1972). A least-squares family of cubic curves with application to golf handicapping. *SIAM Journal on Applied Mathematics, 22*, 77–83.

Schreuder, J. A. M. (1980). Constructing timetables for sports competitions. *Mathematical Programming Studies, 13*, 58–67.

Schreuder, J. A. M. (1992). Combinatorial aspects of construction of competition Dutch professional football leagues. *Discrete Applied Mathematics, 35*, 301–312.

Shikata, M. (1977). Information theory in soccer. *Journal of Humanities and Natural Sciences, 46*, 35–94.

Sinuany-Stern, Z. (1988). Ranking of sport teams via the AHP. *Journal of the Operational Research Society, 39*, 661–667.

Sinuany-Stern, Z., Israeli, Y., & Bar-Eli, M. (2006). Application of the analytic hierarchy process for the evaluation of basketball teams. In S. P. Ladany (Ed.), *Sport Management* special issue. *Sport Management and Marketing*, *1*(3), 193–207.

Skiena, S. S. (1988). A fairer scoring system for Jai-Alai. *Interfaces, 18*(6), 35–41.

Smith, J. C., Fraticelli, B. M. P., & Rainwater, C. (2006). A bracket assignment problem for the NCAA men's basketball tournament. *Transactions in Operational Research, 13*(3), 253–271.

Sokol, J. S. (2003). A robust heuristic for batting order optimization under uncertainty. *Journal of Heuristics, 9*, 353–370.

Sphicas, G. P., & Ladany, S. P. (1976). Dynamic policies in the long jump. In R. E. Machol, S. P. Ladany, & D. G. Morrison (Eds.), *Management science in sports* (TIMS studies in the management sciences, Vol. 4, pp. 113–124). Amsterdam: North-Holland.

Stefani, R. T. (1999). Taxonomy of sport rating systems. *IEEE Transactions on Systems Man and Cybernetics, Part A: Systems and Humans, 29*, 116–120.

Stein, W. E., Mizzi, P. J., & Pfaffenberger, R. C. (1994). A stochastic dominance analysis of ranked voting systems with scoring. *Operational Research, 74*, 78–85.

Takahashi, I. (1990). AHP applied to binary and ternary comparisons. *Operational Research Society, 33*, 199–206.

Terpstra, J. T., & Schauer, N. D. (2007). A simple random walk model for predicting track and field world records. *Journal of Quantitative Analysis in Sports*, *3*(3), paper 4.
Thomas, A. C. (2007). Inter-arrival times of goals in ice hockey. *Journal of Quantitative Analysis in Sports, 3*, 3.
Tomecko, N., & Filar, J. A. (1998). *Player assignments in Australian rules football*. Proceedings of meeting on mathematics and computers in sport (pp. 171–179), Gold Coast, Queensland, Australia.
Townsend, M. S. (1984). *Mathematics in sport*. Chichester, UK: Ellis Horwood.
Tversky, A., & Gilovich, T. (1989). The cold facts about the 'hot hand' in basketball. *Chance, 2*, 16–21.
Ushakov, I. A. (1976). The problem of choosing the preferred element: An application to sport games. In R. E. Machol, S. P. Ladany, & D. G. Morrison (Eds.), *Management science in sports* (TIMS studies in the management sciences, Vol. 4, pp. 153–162). Amsterdam: North-Holland.
Voorhis, T. V. (2002). Highly constrained college basketball scheduling. *Journal of the Operational Research Society, 53*, 603–609.
Washburn, A. (1991). Still more on pulling the goalie. *Interfaces, 21*(2), 59–64.
Willis, R. J., & Terrill, B. J. (1994). Scheduling the Australian state cricket season using simulated annealing. *Operational Research Society, 45*, 276–280.
Wilson, R. L. (1995). Ranking college football teams: A neural network approach. *Interfaces, 25*(4), 44–59.
Winston, W., & Soni, A. (1982). Does division play lead to more pennant races. *Management Science, 28*, 1432–1440.
Wood, G. H. (1945). Cricket scores and geometrical progression. *Journal of the Royal Statistical Society A, 108*, 12–22.
Wright, M. B. (1991). Scheduling English cricket umpires. *Journal of the Operational Research Society, 42*, 447–452.
Wright, M. (1994). Timetabling county cricket fixtures using a form of tabu search. *Journal of the Operational Research Society, 45*, 758–770.
Wright, M., & Hirotsu, N. (2003). The professional foul in football: Tactics and deterrents. *Journal of the Operational Research Society, 54*, 213–221.

Spreadsheets

Donald R. Plane[1] and Cliff T. Ragsdale[2]
[1]Rollins College, Winter Park, FL, USA
[2]Virginia Polytechnic Institute and State University, Blacksburg, VA, USA

Introduction

The electronic spreadsheet is a computer application that displays on a computer screen one or more worksheets. A worksheet is a rectangular grid of numeric and text information. The row and column organization of worksheets mimics many situations, such as an accountant's worksheet, a teacher's grade book, an invoice, or a scientist's data journal. Through the use of the computer keyboard and pointing device (mouse), the user is able to manipulate the information using mathematical, logical, and text operations. The worksheet is the foundation for graphs and reports designed to communicate information in a user-friendly manner.

The computer spreadsheet, operating primarily on personal computers, is a serious and powerful business tool. It is used for many applications including from printing address labels and developing simple budgets to conducting cash flow analysis, financial planning, optimizing investments, tracking production, forecasting, and facilities analysis.

VisiCalc, the first personal computer spreadsheet, was introduced in October, 1979 (Saffo 1989). Until then, the personal computer had been viewed more as a hobbyist's interest than as a serious office machine. Users soon began to realize that the electronic spreadsheet enabled them to change one or more numbers and immediately see the results of the changes in other parts of the worksheet. This capability was instrumental in causing the personal computer, with spreadsheet software, to become an important management tool. The 1982 introduction of Lotus Development Corporation's spreadsheet software called Lotus 1-2-3 marked the availability of a computer application that combined three functions (worksheet, graphics, and database) into software designed for the then new IBM personal computer. The electronic spreadsheet was a very important driving force in the development of the personal computer industry, with the dominant spreadsheet soon being Microsoft's Excel, surpassing Corel's Quattro Pro and IBM's Lotus 1-2-3. Apache OpenOffice (formerly under Sun, then Oracle) and Google Docs are alternatives offering free or inexpensive packages that provide basic spreadsheet functionality while lacking some more advanced features offered by more mature commercial products.

Example

A monthly budget demonstrates the organization of a worksheet and its basic capabilities. The columns of the worksheet show expenditures by monthly, while

Spreadsheets, Table 1 A simple budget spreadsheet

B7	= B2 + B3 + B4 + B5				
	A	B	C	D	E
1		Jan	Feb	Mar	Total
2	Food	220	230	300	750
3	Housing	400	400	400	1,200
4	Clothing	200	50	75	325
5	Entertainment	150	300	75	525
6					
7	Total	970	980	850	

the rows display expenditures by budget category. Totals are shown for each expenditure category and for each month. On a computer screen a very simple worksheet might appear as shown in Table 1. The worksheet columns are identified by letters; the rows are identified by numbers. Cells are identified by the column letter(s) followed by the row number. In Table 1, cell B7 (column B, row 7) contains the total expenditures for January.

The user can change any monthly expenditure item, and the spreadsheet will automatically recalculate the row and column totals. As the size and complexity of a worksheet grow, this ability to recalculate the entire spreadsheet rapidly and automatically replaces many hours of manual labor. Often, a worksheet contains many more columns and rows and the screen can display at one time. The mouse or keystrokes can be used to show any part of the worksheet on the screen.

Prior to the development of spreadsheets, a user wishing to solve a problem using a computer had to write a program using a very precise programming language, with each command, or statement, fed into the computer in a definite order. Such programs often had to be run several times to discover programming and logical errors before they could be used to solve a problem or prepare an analysis. Spreadsheets, on the other hand, allow users to enter spreadsheet cell information in any order and location on the worksheet, even if that organization does not mimic familiar manual computation procedures.

Basic Operations

Formulas, data, and text labels are entered from the keyboard. The formulas define the relationships and logic for each value calculated by the spreadsheet. Data values in various cells are used by the formulas. Text labels are important to users of the spreadsheet and users of reports generated by the spreadsheet. The formulas in a worksheet are the driving force behind a spreadsheet. In Table 1, cell B7 is selected to show its contents, which is the formula defining this cell:

$$= B2 + B3 + B4 + B5$$

where the cells B2, B3, B4, and B5 contain values for food, housing, clothing, and entertainment expenditures for January. Similar formulas are used for other calculated cells in the worksheet.

There are many aspects of modern electronic spreadsheets that enhance their usefulness to people not trained in computer programming. Among the more important:

(a) Formulas can be copied from one cell to another, without disturbing the logic of the formula. In the budget example, the formula for January total can be copied to corresponding cells for February and March.

(b) The worksheet can be organized in a way meaningful to the user, rather than in a way required by computational procedures. A user may elect to put monthly totals at the top, and category totals in another part of the worksheet, without regard to the fact that these values depend upon cells below and to the right. This may be contrary to the way one would manually calculate the budget, which might be column by column or row by row. This natural order of recalculation is intelligence within the program that frees the user from the procedural steps followed in computer programming.

(c) Reports can be printed by the spreadsheet. The spreadsheets available today provide substantial flexibility in formatting the report to meet the needs of the users. Most spreadsheets include spelling checkers to assist in report preparation. Spreadsheets also contain sophisticated formatting capabilities, allowing the user to change the appearance of the screen and report. The spreadsheet may permit the user to change many appearance items, including color, typeface, and character size. Reports can also include graphical images, lines, arrows, boxes, shading, and other visual enhancements often associated with desktop publishing.

(d) Graphs can be created from the numbers in or calculated by the spreadsheet. Graphs are an

integral and dynamic part of the spreadsheet, so that changes in the numbers on the worksheet also appear as changes in the graph. The graphs can be displayed on the screen as separate images, on paper, or as part of the spreadsheet. Modern spreadsheets allow small graphs (called sparklines) to be displayed in individual cells. By including a graphical image as a part of the spreadsheet screen display, the graph may become a part of a report. Some spreadsheets include tools to assist in a slide-show presentation of a sequence of screen images displaying tables, graphs, and text.

Capabilities

Spreadsheets can be used in many different ways, by users with a wide variety of skills. A beginning spreadsheet user may view a worksheet as a way of saving time that would otherwise be spent using a calculator. This capability of spreadsheets is likely the initial reason for their popularity. As spreadsheets have progressed, their capabilities have grown immensely. Some of the more important of these enhanced capabilities are:

(a) Spreadsheets may be linked to external database or websites so that data-intensive applications can be addressed in a spreadsheet environment. Needed information can be selectively retrieved from these data sources and used as inputs for spreadsheet models or summarized using graphs or cross tabulations using pivot tables.

(b) Extensive tools for statistical analysis are included in spreadsheets. Techniques such as regression analysis, correlation analysis, tests of significance, and analysis of variance are typically a part of a spreadsheet's capabilities.

(c) Mathematical capabilities required for engineering and scientific calculations are included in spreadsheets.

(d) Matrix operations (multiply, transpose, invert) can be performed using the commands of a spreadsheet.

(e) Extensive tools for financial analysis are included in spreadsheets.

(f) Optimization algorithms are a part of most spreadsheets. These optimizers, sometimes called Solvers, are capable of addressing linear and nonlinear constrained optimization problems with continuous or discrete decision variables. The method of communicating an optimization problem to a spreadsheet solver may be very different from the traditional methods used by OR/MS practitioners. Instead of formulating a problem as a set of equations and inequalities to be satisfied, the spreadsheet view of an optimization problem might be described by these steps:

1. Construct a model to evaluate or calculate the value of the objective, such as profit, for an arbitrary set of values for the decision variables. Include in the model the values that need to be checked to see if constraints have been exceeded. This includes limiting factors such as raw materials, production capacity, and human resources
2. Identify to the spreadsheet solver the components of the optimization problem:
   - Which cell computes the objective to be maximized or minimized;
   - Which cells are to be adjusted by the optimizer (the decision variables), and
   - Which cells are constraints, and what are the limiting values.
3. Issue the appropriate 'solve' command to the spreadsheet.

In this spreadsheet optimization environment, the spreadsheet is serving as a powerful problem generator, as an optimizing algorithm, and as a report generator to communicate the results of the optimization.

(g) Spreadsheet add-in software is available to expand spreadsheet capabilities in areas such as Monte Carlo risk analysis, optimization (including optimization with genetic algorithms), forecasting, data mining and analysis, and other applications involving neural networks.

(h) Spreadsheets may serve as the environment for developing sophisticated software. The capability to include macros or a set of procedures or steps to follow, gives the spreadsheet many of the structures of traditional programming languages, such as sequence, decision, loop, and case. The application programming interface (API) of spreadsheets can also be used by other programs to access spreadsheet functionality from outside the visible spreadsheet environment.

Concluding Remarks

Spreadsheets have become the primary computer application software for many business managers and other professionals. Many introductory OR/MS textbooks center on a spreadsheet-based approach to presenting the topics (e.g., Ragsdale 2010; Winston and Albright 2012). While this approach may (or may not) ignore the algorithmic and mathematical aspects of OR/MS, it presents the basic tools of the discipline in a user-friendly manner, using spreadsheets as a language that is more comfortable than mathematics to many potential users of OR/MS. This provides both opportunities and pitfalls. As more end-users are aware of the spreadsheet OR/MS tools, these tools will be applied more widely. But as the use expands, those using the tools will be less familiar with mathematics and assumptions behind the tools, leading to a new set of challenges.

See

- ▶ Algebraic Modeling Languages for Optimization
- ▶ Information Systems and Database Design in OR/MS
- ▶ Linear Programming
- ▶ Monte Carlo Simulation
- ▶ Nonlinear Programming
- ▶ Visualization

References

Ragsdale, C. T. (2010). *Spreadsheet modeling and decision analysis: A practical introduction to management science* (6th ed.). Cincinnati, OH: Cengage Learning.

Saffo, P. (1989). Looking at visicalc 10 years later. *Personal Computing, 13*, 233–236.

Winston, W. L., & Albright, S. C. (2012). *Practical management science: Spreadsheet modeling and applications* (4th ed.). Cincinnati, OH: Cengage Learning.

SQC

Statistical quality control.

See

- ▶ Quality Control

Square Root Law

When a model result is proportional to the square root of input variables and/or parameters. One example is the formula that indicates that the average distance that an emergency unit must travel to a call scene is proportional to the square root of the area it services. Another example is the economic order quantity (EOQ) result from inventory modeling.

st or s.t.

Abbreviation: (i) "subject to" as in the linear-programming problem: Minimize $\boldsymbol{cX}$ st $\boldsymbol{Ax} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$. (ii) "such that" as in a mathematical statement: There exists a constant N s.t. for all $n > N \ldots$

St. Petersburg Paradox

Paradox that arises in a simple gambling game in which a fair coin is tossed repeatedly until a heads appears, at which point the payoff is \$2 doubled for each toss. Since the expected value of such a game is given by $(0.5)(2) + (0.5)^2(2)^2 + (0.5)^3(2)^3 + \ldots$, which is infinite, a decision maker who uses expectation to value the game would assign an infinite value. The concepts of utility and risk can be used to resolve this apparent paradox.

See

- ▶ Bayesian Decision Theory, Subjective Probability, and Utility
- ▶ Utility Theory

Stages

The set of sequential steps in a model for either (i) probability distributions, or (ii) dynamic programming. In applied probability models, especially queueing, such modeling allows

non-exponentially distributed random variables such as interarrival and service times to be represented as a sequence of exponentially distributed random variables, each of which is referred to as a stage, thus enabling the system to be modeled by a Markov chain. If there are k stages that are independent and identically distributed, the resulting distribution is called a k-Erlang distribution, represented as E_k in Kendall's queueing notation; if the stages are only independent, the distribution is called a generalized Erlang; further extensions lead to Coxian and phase-type distributions. In dynamic programming, stages are the time subdivisions of a dynamic programming model where decisions are made upon which the state then evolves to the next state; also called periods.

See

- ▶ Coxian Distribution
- ▶ Dynamic Programming
- ▶ Kendall's Notation
- ▶ Markov Chains
- ▶ Method of Stages
- ▶ Phase-type Probability Distributions
- ▶ Queueing Theory

Staircase Structure

A linear-programming problem in which the constraint set can be arranged into connecting blocks such that the first block is connected to the second block by a few variables, the second block is connected to the third block by a few variables, and so on. Staircase structures arise in production problems over time in which the connecting variables are inventories that carry over from one time period to the next. The matrix of coefficients defined by such structures is very sparse.

See

- ▶ Block-Angular System
- ▶ Large-scale Systems
- ▶ Super-Sparsity
- ▶ Weakly-coupled Systems

Stanford-B Model

- ▶ Learning Curves

Stationary Distribution

In a discrete-time Markov chain, the state probability distribution (vector) $\boldsymbol{\pi}$ that satisfies $\boldsymbol{\pi} = \boldsymbol{\pi}\boldsymbol{P}$, where $\boldsymbol{P}$ is the single-step transition matrix. Mathematically, this is equivalent to finding the eigenvector associated with the eigenvalue 1 of the stochastic matrix $\boldsymbol{P}$. Similarly, for a continuous-time Markov chain, the stationary distribution satisfies $\boldsymbol{\pi}\boldsymbol{Q} = \boldsymbol{0}$, where $\boldsymbol{Q}$ is the transition rate matrix. Also known as the invariant distribution. If the Markov chain is ergodic, it has a limiting (or steady-state) distribution that equals the stationary distribution.

See

- ▶ Limiting Distribution
- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Statistical Equilibrium

Stationary Stochastic Process

A stochastic process in which the state probability distributions are invariant over time.

Stationary Transition Probabilities

When the transition probabilities of a Markov chain or Markov process are time-invariant, i.e., for times $s < t$ in the time domain T, and any state x and any set A in the state space, $\Pr\{X(t) \in A | X(s) = x\} = \Pr\{X(t-s) \in A | X(0) = x\}$.

See

- ▶ Markov Chains
- ▶ Markov Processes

Statistical Equilibrium

Let $p_{ij}(t)$ be the probability that a stochastic process takes on value j at time t (discrete or continuous), given that it began at time 0 from state i. If for each j, $p_{ij}(t)$ approaches a limit p_j independent of i by taking $t \to \infty$, then the process is said to reach statistical equilibrium. For an ergodic Markov chain in statistical equilibrium, the corresponding limiting distribution is identical to the stationary distribution.

See

- ▶ Limiting Distribution
- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Stationary Distribution

Statistical Process Control

- ▶ Quality Control

Statistical Ranking and Selection

Seong-Hee Kim
Georgia Institute of Technology, Atlanta, GA, USA

Introduction

Ranking and selection (R&S) procedures are statistical tools for selecting the best system among a finite number of simulated systems. Depending on the definition of the best, there exist at least four classes of R&S problems in simulation studies: selecting the system with the largest or smallest expected performance measure (selection of the best); finding systems whose performance measures are significantly better than a standard and, if there is any, selecting the one with the largest or smallest performance (comparison with a standard); selecting the system with the largest probability of actually being the best performer (multinomial selection); and selecting the system with the largest probability of success (Bernoulli selection).

Approaches to solve R&S problems include subset selection methods, indifference-zone methods, Bayesian methods, and optimal computing budget allocation (OCBA). Subset selection and indifference-zone (IZ) methods find the best system with a guarantee on the probability of correct selection (PCS), whereas the other two methods maximize the PCS under a limited computational budget. The focus here will be on procedures for the selection-of-the-best problem with a guarantee on the PCS. Other classes of R&S problems are briefly discussed in the concluding section; see Chick (2006) for a review of Bayesian and OCBA methods.

Development of efficient R&S procedures has led to combining these procedures with optimization via simulation (OvS) algorithms, so some R&S procedures related to OvS are discussed. R&S procedures can be applied to "clean up" at the end of an OvS search, finding the best among all the solutions actually simulated so far by the search with a statistical guarantee, or they can be embedded within these OvS algorithms to help them move to the improving direction correctly and efficiently.

Also discussed in more detail is a more complicated form of R&S, namely constrained R&S, where the goal is to find the best system under a primary performance measure while also satisfying stochastic constraints on secondary performance measures. For example, the decision maker may want to select a production schedule for a manufacturing system that yields the largest expected throughput among a number of different schedules, while keeping the expected lead time in the system bounded (smaller than or equal to some constant) at the same time.

Problem Setting

Let X_{im} denote the mth observation from system $i (i = 1, 2, \ldots, k)$. The set of all possible systems is defined as $S = \{1 \ldots, k\}$. Let $x_i = E[X_{im}]$ and $\sigma_i^2 = Var[X_{im}]$ be the mean and variance of the outputs from system i, respectively.

The problem is to determine which system has the best performance measure:

$$\arg\max_{i \in S} x_i.$$

Without loss of generality, assume that $x_k \geq x_{k-1} \geq \cdots \geq x_1$, so that (unknown to the decision maker) system k is the best system. Moreover, the $\{X_{im}\}$ are assumed to satisfy the following assumptions.

Assumption 1. *For each* $i = 1,2,\ldots,k$,

$$X_{im} \overset{iid}{\sim} \mathcal{N}\left(x_i, \sigma_i^2\right) \; m = 1, 2, \ldots$$

Where $\overset{iid}{\sim}$ *denotes independent and identically distributed (iid) and* $\mathcal{N}(\mu, \sigma^2)$ *denotes the normal distribution with mean* μ *and variance* σ^2.

Assumption 1 is a common assumption in many R&S procedures. When the outputs from system i (X_{i1}, X_{i2},...) are either within-replication averages or batch means from a single sufficiently long replication after accounting for the elimination of the initialization bias, the iid normality assumption is reasonable (Law and Kelton 2000).

Assumption 2. *For* $(i,m) \neq (j,m')$, X_{im} *and* $X_{jm'}$ *are independent*.

Assumption 2 implies that all systems are simulated independently. Assigning different streams of pseudo-random numbers to the simulation of each system ensures independence of systems. Procedures that require this assumption for statistical validity are reviewed in the next section, while the subsection on common random numbers reviews procedures that do not need the independence assumption.

For the selection of the best, two approaches are considered: subset selection and IZ approaches. Gupta (1965) presented the subset selection formulation of the problem, and Bechhofer (1954) established the IZ formulation.

The subset selection procedures use outputs available for each system to obtain a subset $I \subseteq \{1, 2, \ldots, k\}$ such that

$$\Pr\{k \in I\} \geq 1 - \alpha \tag{1}$$

where $1/k<1-\alpha<1$.

On the other hand, the IZ approach suggested by Bechhofer (1954) attempts to find the single best system k whose mean is at least a user-specified amount better than the means of the other systems with a guarantee on PCS. The IZ parameter is denoted as δ, a practically significant difference worth detecting. Specifically, the IZ procedure should guarantee

$$\Pr\{\text{select } k | x_k - x_{k-1} \geq \delta\} \geq 1 - \alpha \tag{2}$$

where $1/k<1-\alpha<1$. If there are systems whose means are within δ of the best, then the decision maker is indifferent to which of these is selected.

Other notation is defined as follows:

n_0 = first-stage sample size

n_i = sample size available for system i

$\bar{X}_i(n)$ = sample average of n observations of system i, i.e., $\frac{1}{n}\sum_{m=1}^{n} X_{im}$

$S^2_{x_i}(n)$ = sample variance of $\{X_{i1}, \ldots, X_{in}\}$

$S^2_{x_{ij}}(n)$ = sample variance of $\{X_{i1} - X_{j1}, \ldots, X_{in} - X_{jn}\}$

$R(r;v,w,z) = \max\left\{0, \frac{wz}{v} - \frac{v}{2}r\right\}$, for $v,w,z \in \mathbb{R}^+, v \neq 0$

Finding the Best

A few illustrative procedures for the selection-of-the-best problem are provided in this section: one from the subset selection approach and two from the IZ approach.

Subset Selection

Suppose that a number of samples are already available for each system. A subset selection procedure returns a subset that contains the best system with probability at least $1 - \alpha$. A single-stage subset selection procedure that allows for unequal and unknown variances across systems was developed by Nelson et al. (2001), with a generalization that also permits unequal sample sizes presented below.

Extended Screen-to-the-Best Procedure (Boesel et al. 2003)

1. Select the overall desired confidence level $1 - \alpha$ and sample size n_i for system i, $i = 1,2,\ldots,k$. Set $t_i = t_{\frac{1}{(1-\alpha)^{\frac{1}{k-1}}}, n_i - 1}$, where $t_{\beta,\nu}$ is the β quantile of the t distribution with ν degrees of freedom.
2. Obtain n_i outputs $X_{im}(m = 1, 2, \ldots, n_i)$ from each system $i(i = 1, 2, \ldots, k)$.

3. Compute the sample means and variances $\bar{X}_i(N_i)$ and $S^2_{x_i}(n_i)$ for $i = 1,2,\ldots,k$. Let

$$W_{ij} = \left(\frac{t_i^2 S^2_{x_i}(n_i)}{n_i} + \frac{t_j^2 S^2_{x_j}(n_j)}{n_j}\right)^{\frac{1}{2}}, \forall i \neq j.$$

4. Set $I = \{i : 1 \leq i \leq k \text{ and } \bar{X}_i(n_i) \geq \bar{X}_j(n_j) - W_{ij}, \forall j \neq i\}$.
5. Return I as the subset of retained systems.

The above procedure satisfies (1) under Assumptions 1 and 2, and is incorporated into the output analysis package of the simulation software Arena (Rockwell Software). A disadvantage of this procedure is that the size of I is unknown and can be as large as k, but no procedure can guarantee a subset of size one and simultaneously satisfy (1) for arbitrary n_i. The next subsection discusses procedures that return one single system which is likely to be the best or near-best.

Indifference-Zone Procedures

In stochastic simulation, it is impossible to find the true best with certainty when the sample size is finite. Instead many procedures employ the IZ approach as a compromise. The IZ approach guarantees (2), finding the best with high probability whenever the best is at least δ amount better than the means of the other systems. Many IZ procedures are sequential with multiple stages. For example, Rinott's procedure (1978) has two stages. In the first stage, the procedure obtains initial samples for each system and pauses sampling to calculate some statistics (usually sample means and variances). Then it resumes sampling to obtain additional samples for each system in the second stage.

Classical IZ procedures become inefficient when the number of alternatives is large, because they are developed under the Least Favorable Configuration (LFC) condition. If a procedure guarantees at least $1 - \alpha$ PCS under the LFC, it will do so for all other configurations. The Slippage Configuration (SC) is the configuration $\mu_i = \mu_k - \delta$ for all $i \neq k$, which is known to be the LFC in most IZ procedures. When the number of systems is large, it is unlikely that systems face the SC, as means of the systems tend to be spread out rather than all clustered near the best. Thus, a procedure developed under the SC takes more samples than needed when actual differences are greater than δ.

To overcome the inefficiency of IZ procedures, screening is used. The idea is to identify clearly inferior systems after some initial samples and eliminate them from further consideration early. Combining subset selection algorithms with two-stage IZ procedures, the NSGS procedure due to Nelson et al. (2001) ensures the overall PCS $1 - \alpha$ by decomposing the overall error α into α_0 and α_1, and use the decomposed errors for setting up procedure parameters for an initial screening stage and a second ranking stage, respectively.

Procedure NSGS (Nelson et al. 2001)

1. *Setup*. Select the overall desired confidence level $1 - \alpha$, IZ parameter δ, and common first-stage size $n_0 \geq 2$. Set

$$t = t_{1-(1-\alpha/2)^{\frac{1}{k-1}}, n_0-1}$$

and obtain Rinott's constant $h = h(n_0, k, 1 - \alpha/2)$ from Table 8.3 in Goldsman and Nelson (1998).
2. *Initialization*. Obtain n_0 outputs $X_{im}(m = 1, 2, \ldots, n_0)$ from each system $i(i = 1, 2, \ldots, k)$. Calculate $S^2_{x_i}(n_0)$ for $i = 1, 2, \ldots, k$.
3. *Subset Selection*. Calculate the quantity

$$W_{ij} = t\left(\frac{S^2_{x_i}(n_0) + S^2_{x_j}(n_0)}{n_0}\right)^{1/2}, \forall i \neq j.$$

Form the screening subset I, containing every alternative i such that $1 \leq i \leq k$ and

$$\bar{X}_i(n_0) \geq \bar{X}_j(n_0) - \max\{0, W_{ij} - \delta\}$$
for all $j \neq i$.

4. *Ranking*. If $|I| = 1$, then stop and return the system in I as the best. Otherwise, for all $i \in I$, calculate the second-stage sample sizes

$$N_i = \max\left\{n_0, \left\lceil\frac{hS_{x_i}(n_0)}{\delta}\right\rceil^2\right\},$$

where $\lceil\cdot\rceil$ is the ceiling function.

5. Take $N_i - n_0$ additional outputs from all systems $i \in I$.
6. Calculate the overall sample means $\bar{X}_i(N_i)$ for all $i \in I$. Select the system with the largest $\bar{X}_i(N_i)$ as best.

Note that NSGS has two stages and screens out systems only once after the first stage. There are fully sequential procedures that take a single basic observation from each alternative still in play at each stage and eliminate systems from further consideration when it is statistically clear that they are inferior. The KN procedure (Kim and Nelson 2001) is a fully sequential procedure that allows for unequal and unknown variances across systems, useful in simulation environments.

Both NSGS and KN are statistically valid guaranteeing (2) under Assumptions 1 and 2. Also, they have been shown to be efficient when hundreds of systems (up to 500) are compared. Note that $R(r; \cdot)$ in KN defines a triangular continuation region for the partial sum process, $\sum_{m=1}^{r}(X_{im} - X_{jm})$. As long as the partial sum process stays within the triangular continuation region, sampling for systems i and j continues. Otherwise, an elimination decision is made. A different shape of continuation regions can be used. For example, Batur and Kim (2006) present fully sequential procedures with a parabolic continuation region, which show a meaningful improvement over the triangular continuation region.

Procedure KN (Kim and Nelson 2001)

1. *Setup*. Select the overall desired confidence level $1 - \alpha$, IZ parameter δ and common first-stage sample size $n_0 \geq 2$. Set

$$\eta = \frac{1}{2}\left[\left(\frac{2\alpha}{k-1}\right)^{-2/(n_0-1)} - 1\right].$$

2. *Initialization*. Let $I = \{1, 2, \ldots, k\}$ be the set of systems still in contention, and let $h^2 = \eta(n_0 - 1)$.
   Obtain n_0 outputs $X_{im}(m = 1, 2, \ldots, n_0)$ from each system $i(i = 1, 2, \ldots, k)$.
   For all $i \neq j$ calculate $S^2_{x_{ij}}(n_0)$, the sample variance of the difference between systems i and j. Set $r = n_0$.
3. *Screening*. Set $I^{\text{old}} = I$. Let

$$I = \left\{ i : i \in I^{\text{old}} \text{ and } \sum_{m=1}^{r}(X_{im} - X_{jm}) \geq -R\left(r; \delta, h^2, S^2_{x_{ij}}(n_0)\right), \forall j \in I^{\text{old}}, j \neq i \right\}.$$

4. *Stopping Rule*. If $|I| = 1$, then stop and select the system whose index is in I as the best. Otherwise, take one additional output $X_{i,r+1}$ from each system $i \in I$, set $r = r + 1$ and go to *Screening*.

Typically, KN reaches a decision faster than NSGS with fewer number of observations. However, KN tends to require a large number of switches from simulating one system to simulating another, whereas NSGS needs at most $2k - 1$ switches. The cost of stopping and restarting complex simulations can be quite high both in time and storage. In modern computing environments that utilize parallel computing, the switching cost is less of an issue, making KN attractive. The KN procedure is incorporated into the output analysis package of Simio® (Simio LLC).

Some subset selection and IZ procedures are closely related to multiple comparison procedures. For detailed discussion for the connection to the multiple comparison procedures, see Kim and Nelson (2006b).

Efficiency

There are a number of ways to further enhance efficiency of procedures discussed in the previous section.

Common Random Numbers

Many procedures require Assumption 2 that systems are simulated independently. If the same random number streams are assigned to each simulation known as common random numbers (CRN), then under fairly general conditions, positive correlation is induced among the systems, and the variance of the

difference is decreased in observed average performances between two systems. Although CRN makes statistical procedures more complicated when there are more than two systems, CRN makes comparison sharper, meaning spending fewer number of observations until a decision is made.

For subset selection procedures, a special case of the Extended Screen-to-the-Best procedure where $n_i = n$ for all i remains valid under CRN (Nelson et al. 2001) provided $S^2_{x_i}(n)/n + S^2_{x_j}(n)/n$ is replaced by $S^2_{x_{ij}}(n)/n$ and $t = t_{1-\alpha/(k-1),n-1}$. For unequal sample sizes n_i, the statistical validity of the Extended Screen-to-the-Best procedure does not hold under CRN.

For IZ procedures, the procedure of Nelson and Matejcik (1995) extended Rinott's two-stage procedure. Their procedure works in conjunction with CRN under a special structure of the variance-covariance matrix, called sphericity. As the sphericity assumption is often violated for large k, the use of CRN is not recommended in Nelson and Metejcik's procedure when k is large. NSGS requires systems to be simulated independently, whereas KN is statistically valid with or without CRN.

Steady-State Simulations

The R&S procedures discussed so far require Assumption 1, which is appropriate for terminating simulations. In steady-state simulation, the goal is to estimate long-run performance, after the impact of the initial conditions have vanished. The iid normality assumption applies to steady-state simulation experiments if the experimenter is willing to make multiple replications of each system with a good warm up and use the within-replication averages as the basic observations. Or the experimenter can generate a single long replication of each system to avoid estimation bias due to residual effects of the initial conditions. The difficulty in a single-replication design is that the raw outputs within a replication (such as waiting times of individual customers in a queueing system) are typically neither normally distributed nor independent. In order to achieve the iid normality, one can take batch means of many individual raw outputs as the basic observations when only a single replication is made. Then, batch means are approximately iid normal for a large enough batch size. See Law and Kelton (2000) for a more detailed discussion of replication versus batching in steady-state simulation.

Both of these remedies for dependent data are inefficient. Multiple replications require warm up from each replication and may result in deletion of a large number of outputs; and batching within a replication forces selection procedures to make elimination and selection decisions at long intervals. This is especially undesirable for fully sequential procedures where elimination occurs every basic observation. Thus, procedures that take the raw outputs within a single replication as basic observations are desirable for steady-state simulation.

A few procedures have been presented specifically for steady-state simulation that make a single replication from each system and take raw outputs rather than batch means as basic observations (e.g., Damerdji and Nakayama 1999; Goldsman et al. 2002; Kim and Nelson 2006a). One of the asymptotically valid procedures, called KN++ (Kim and Nelson 2006a), updates variance estimates as more observations are available. Variance update is shown to improve efficiency of the procedures greatly, although it may cause some technical and computational difficulties such as data storage and recalculation of estimates.

Slippage Configuration

Screening, the use of CRN when applicable, the use of raw outputs rather than within-replication averages or batch means in steady-state simulation, and variance update greatly improve the efficiency of IZ procedures. Unfortunately, the actual PCS of many IZ procedures tends to be close to 100%, i.e., overly conservative, for large k even with all these amendments, which implies that there is room for further improvement.

As discussed earlier, because IZ procedures are derived under the LFC (i.e., the SC), they are generally very conservative. The SC is essential in deriving statistically valid IZ procedures, because it frees the procedures from dependence on the unknown true differences among the means. A remedy to the SC is to replace the IZ parameter with estimated mean differences based on the first-stage samples, e.g., by adjusting the IZ parameter for system i to $\delta_i \equiv \max(\delta, \bar{X}_i(n_0) - \bar{X}_b(n_0))$, where b is the identity of a system with the largest first-stage sample

mean (Chen and Kelton 2005). Then use the adjusted IZ parameter to calculate the total number of observations N_i for each system i in a two-stage procedure such as Rinott's procedure. A similar technique can be used for fully sequential procedures by adjusting the IZ parameter between system i and j to $\delta_{ij} \equiv \max\left(\delta, |\bar{X}_i(n_0) - \bar{X}_j(n_0)|\right)$ (Healey 2010; Wang and Kim 2013). The adjusted IZ parameter δ_{ij} is used to calculate $R(r;\cdot)$ when comparing systems i and j. These modifications allow the procedures to make decisions quicker but at the cost of statistical validity and observed PCS.

R&S in Optimization

The procedures discussed in previous sections require keeping or running simulation models of all systems. When the number of alternative systems is extremely large, it may not be possible to keep simulation models of all systems, and thus those R&S procedures become inappropriate. Instead, a different class of methods, called optimization via simulation (OvS), is needed; for overviews, see Andradóttir (2006) and Fu (2006).

R&S procedures can assist OvS in two ways: clean-up at the end of the search or efficient and correct selection of the best neighbor.

Clean up: Commercial add-on products for simulation software packages employ a combination of heuristic optimization methods (genetic algorithms, tabu search, etc.) originally developed for a deterministic optimization problem. Each alternative is evaluated by a simulation model through a number of replications and sampling error is ignored. As results, heuristic algorithms give no statistical meaningful estimates and provide no information about how close the chosen system is to the true best. To add statistical confidence, a R&S procedure can be employed to ensure that the selected system is the best or near-best of all systems that the search actually did encounter. The number of systems to compare (all those encountered by the search) is large, and they may not have been simulated equally, so a procedure should be able to handle unequal sample sizes. Nelson et al. (2001) provide a revised version of the NGSG procedure, the Group-Screening procedure, in which one can avoid simulating all the systems simultaneously. Boesel et al. (2003) extend the Group-Screening procedure to account for unequal sample sizes to "clean up" after the search is done. Sequential Selection with Memory (SSM), an extension of the KN procedure that uses partial or complete information on systems previously visited, was developed by Pichitlamken et al. (2006), and is used in the commercial product OptQuest (OptTek Systems, Inc.).

Selection of the promising neighbor: Some OvS algorithms require the selection of the best neighbor from a finite number of alternatives. For example, the nested partitions (NP) method (Shi and Ólafsson 2000) and the convergent optimization via most-promising-area stochastic search (COMPASS) (Hong and Nelson 2006) repeat search iterations where a number of candidate solutions are sampled, their performances are evaluated through a number of replications, and the best neighbor is selected. In stochastic simulation, sampling error dramatically complicates selecting the best neighbor. To select the best neighbor confidently, a large number of replications for performance evaluation is needed. But then too much computational effort on the selection hinders the search to make much progress in the time available. Thus, the efficient and correct selection of the best neighbor is critical to the overall performance of an optimization algorithm in stochastic simulation. Pichitlamken and Nelson (2003) use SSM for selection of the best neighbor within their OvS algorithm and show that the use of SSM indeed enhances the overall performance of the algorithm.

Constrained R&S

Due to physical or managerial limits placed on a system, performance measures other than the primary performance measure often need to be considered. To handle multiple performance measures, one can formulate the problem as a multi-objective problem (e.g., Butler et al. 2001) or place constraints on secondary performance measures. The latter approach forms constrained R&S, for which a fully sequential IZ approach developed by Andradóttir and Kim (2010) is presented in this section, including both the feasibility check of multiple secondary performance measures and the comparison of primary performance measures.

Let $x_i = \mathrm{E}[X_{im}]$ and $y_{i\ell} = \mathrm{E}[Y_{i\ell m}]$ be the expected values of the primary and secondary constrained

performance measures for each system $i \in S$ and constraints $\ell = 1, \ldots, s$. The objective is to select a system with the best primary performance measure while satisfying all of the constraints:

$$\arg\max_{i \in S} x_i$$
$$\text{s.t. } y_{i\ell} \le q_\ell \text{ for all } \ell = 1, \ldots, s.$$

Similar to Assumption 1, the following assumption is needed.

Assumption 3. For each $i = 1,2,\ldots,k$,

$$\begin{bmatrix} X_{im} \\ Y_{i1m} \\ \vdots \\ Y_{ism} \end{bmatrix} \overset{iid}{\sim} \mathcal{MN}\left(\begin{bmatrix} x_i \\ y_{i1} \\ \vdots \\ y_{is} \end{bmatrix}, \Sigma_i \right), \; m = 1, 2, \ldots$$

where $\mathcal{MN}$ denotes the multivariate normal distribution and Σ_i is the $(s+1) \times (s+1)$ covariance matrix of the vector $(X_{im}, Y_{i1m}, \ldots, Y_{ism})$.

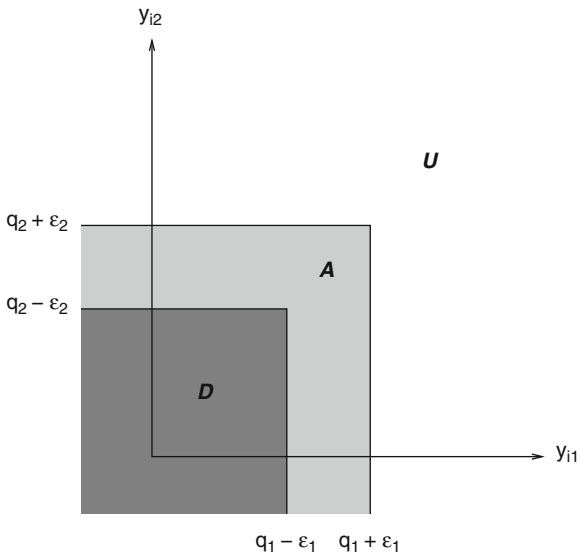
For the primary performance measure, δ still denotes the IZ parameter. The decision maker is essentially indifferent among the feasible systems whose primary performance measures are within δ of each other.

For the secondary performance measures, the smallest significant distance is ε_ℓ, called the tolerance level associated with the constraint ℓ. Systems fall into one of the following three categories for the constraints:

- Any system with $y_{i\ell} \le q_\ell - \varepsilon_\ell$ for all $\ell = 1, \ldots, s$ is considered desirable. The set of all desirable systems is denoted S_D.
- Systems that have at least one mean secondary performance measure greater than $q_\ell + \varepsilon_\ell$ (i.e., $y_{i\ell} \ge q_\ell + \varepsilon_\ell$ for some ℓ) are unacceptable and infeasible, placing them in the set S_U.
- Systems that fall within the tolerance level of q_ℓ for some ℓ, so that $q_\ell - \varepsilon_\ell < y_{i\ell} < q_\ell + \varepsilon_\ell$, and below the tolerance level for the remaining constraints are acceptable and they are placed in the set S_A.

Figure 1 shows the desirable (D), acceptable (A), and unacceptable (U) regions in terms of q_ℓ and ε_ℓ for $\ell = 1,2$ when there are two stochastic constraints.

Let $[b]$ be the index of the best desirable system. In constrained R&S, a CS event is defined as the event



Statistical Ranking and Selection, Fig. 1 D Desirable, A acceptable, and U unacceptable regions when there are two stochastic constraints

that a desirable or acceptable system is selected whose mean is greater than $x_{[b]} - \delta$. The procedures seek to guarantee

$$\Pr\left\{\text{select } i \in S_D \cup S_A \text{ with } x_i > x_{[b]} - \delta\right\} \ge 1 - \alpha. \tag{3}$$

Constrained R&S requires feasibility check of multiple secondary performance measures and selection of the best feasible system. First, some further notation must be introduced (where superscripted "T" denotes the vector transpose):

$$\mathbf{Y}_{im} = (Y_{i1m}, Y_{i2m}, \ldots, Y_{ism})^T;$$
$$\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_s)^T, \varepsilon_\ell \in \mathbb{R}^+ \text{for } \ell = 1, \ldots, s;$$
$$\mathbf{q} = (q_1, q_2, \ldots, q_s)^T, q_\ell \in \mathbb{R} \text{ for } \ell = 1, \ldots, s;$$
$$\boldsymbol{a} = (a_1, a_2, \ldots, a_s)^T, a_\ell \in \mathbb{R}^+ \text{for } \ell = 1, \ldots, s;$$
$$Y^a_{im} = \boldsymbol{a}^T \mathbf{Y}_{im};$$
$$\varepsilon^a = \boldsymbol{a}^T \boldsymbol{\varepsilon};$$
$$q^a = \boldsymbol{a}^T \mathbf{q};$$
$$S^2_{y_{i\ell}}(n) = \text{sample variance of} \{Y_{i\ell 1}, \ldots, Y_{i\ell n}\}$$
for the ℓth constraint of system i;
$$S^2_{y^a_i}(n) = \text{sample variance of} \left\{Y^a_{i1}, \ldots, Y^a_{in}\right\}.$$

Feasibility Check

To solve constrained R&S, feasibility check procedures are necessary that find a set F such that $S_D \subseteq F \subseteq (S_D \cup S_A)$ with a pre-specified probability, say $1 - \alpha_0$. Andradóttir and Kim (2010) developed an IZ procedure for feasibility check in the presence of one stochastic constraint, which was extended to multiple stochastic constraints by Batur and Kim (2010) using the Bonferroni inequality, which makes the procedure conservative for large k and s. To lessen the conservativeness, they introduce an accelerated feasibility check which features an artificial constraint in addition to original s constraints. The artificial constraint is obtained by aggregation (or linear combination) of all secondary performance measures $\mathbf{Y}_{im}^a$, the aggregated tolerance level ε^a, and the aggregated target value q^a.

This aggregate constraint adds some complexity, but can quickly eliminate systems that violate multiple constraints. The aggregate constraint should be used only for making infeasibility decisions and not for making a feasibility determination. The accelerated feasibility check procedure with aggregation (F_A) keeps M, the set of systems whose feasibility is yet to be determined, F, the set of systems found feasible, K_i, the set of constraints found feasible for system i, and A, the set of systems whose feasibility needs to be checked by the aggregate constraint. The F_A procedure continues samplings for any system in M and declares system i to be feasible if all constraints are found feasible (i.e., $|K_i| = s$).

Procedure F_A (Batur and Kim 2010)

1. *Setup*. Choose the overall desired confidence level $1 - \alpha_0$, vector of tolerance levels $\boldsymbol{\varepsilon}$, and first stage sample size $n_0 \geq 2$. Compute ε^a and q^a where $\boldsymbol{a} = [a_\ell]_{\ell=1,2,\ldots,s}$ such that $a_\ell = \prod_{v=1,v\neq\ell}^{s} \varepsilon_v$. Set $\eta = \frac{1}{2}\left[(2\beta)^{-2/(n_0-1)} - 1\right]$.
2. *Initialization*. Let $M = \{1, 2, \ldots, k\}$, $F = \emptyset$, $A = S$, and $K_i = \emptyset$, $i = 1,2,\ldots,k$. Set $h^2 = \eta(n_0 - 1)$. Obtain observations $\mathbf{Y}_{im}$ and compute Y_{im}^a, $m = 1,2,\ldots,n_0$, from each system i. For each system i and constraint $\ell = 1,2,\ldots,s$, compute the sample variance $S_{y_{i\ell}}^2(n_0)$ and $S_{y_i^a}^2(n_0)$. Set the number of observations $r = n_0$ and go to *Feasibility Check*.
3. *Feasibility Check*. For each $i \in M$ and any $\ell \notin K_i$, $\ell = 1,2,\ldots,s$, if

$$\sum_{m=1}^{r} (Y_{i\ell m} - q_\ell) \geq +R\left(r; \varepsilon_\ell, h^2, S_{y_{i\ell}}^2(n_0)\right),$$

then eliminate i from M; else if

$$\sum_{m=1}^{r} (Y_{i\ell m} - q_\ell) \leq -R\left(r; \varepsilon_\ell, h^2, S_{y_{i\ell}}^2(n_0)\right),$$

then add ℓ to K_i. For each $i \in M$, if $|K_i| = s$, then move i from M to F.
For each system $i \in M \cap A$, if

$$\sum_{m=1}^{r} \left(Y_{im}^a - q^a\right) \geq +R\left(r; \varepsilon^a, h^2, S_{y_i^a}^2(n_0)\right),$$

then eliminate i from M and A. For $i \in M \cap A$ with

$$\sum_{m=1}^{r} \left(Y_{im}^a - q^a\right) \leq -R\left(r; \varepsilon^a, h^2, S_{y_i^a}^2(n_0)\right),$$

remove i from A.
4. *Stopping Rule*. If $|M| = 0$, then return F as a set of feasible systems. Otherwise, take one additional observation $Y_{i,r+1}$ from each system $i \in M$ and compute $Y_{i,r+1}^a$. Set $r = r + 1$ and go to *Feasibility Check*.

F_A identifies all desirable and some acceptable systems with at least $1 - \alpha_0$ probability when $\beta = \alpha_0/(k(s+1))$ under Assumption 3. For practical use, the choice of $\beta = \alpha_0/ks$ is recommended. The values $a_\ell = \prod_{v=1,v\neq\ell}^{s} \varepsilon_v$, for $\ell = 1,2,\ldots,s$, are chosen to minimize the area where systems may be unacceptable for the original s constraints, but become acceptable for the aggregate constraint.

Finding the Best Feasible

A fully sequential, IZ framework for constrained R&S consisting of two phases, i.e., feasibility check and comparison of alternative systems, was introduced by Andradóttir and Kim (2010) for one stochastic constraint and extended to multiple stochastic

constraints by Healey (2010). These two phases can be performed either sequentially (the feasibility of each system is determined before comparison begins) or simultaneously (the feasibility check and comparison screening occur simultaneously after each additional sample). In either case, the overall error α needs to be split into α_0 for feasibility check and α_1 for comparison, to ensure the overall PCS $1 - \alpha$.

Sequentially running procedures apply a feasibility check procedure such as F_A first to all systems. Then a comparison procedure is applied to only those systems that are survived in the completed feasibility check phase. As feasibility check ends at a different number of observations for each system, the comparison phase needs a procedure that allows for unequal sample sizes across systems such as SSM. Sequentially running procedures are simple to implement but their statistical validity is hard to prove.

Simultaneously running procedures perform the following two steps after each stage of sampling. First, feasibility screening is performed for undetermined systems in M and eliminates systems that are found infeasible. Second, the procedure compares systems in contention. If a system i is found inferior to a feasible system, the inferior system i is eliminated. If a system is found inferior to a system in M, the procedure cannot eliminate the inferior system i until feasibility of the superior system is determined. Sampling from the inferior system continues until the inferior system is declared infeasible, its superior system is declared feasible, or the inferior system is either eliminated by another feasible system or selected as the best. Basically, simultaneously running procedures allow the elimination of a system only when it is declared infeasible or when it is found inferior to a system declared feasible.

Simultaneously running procedures are more complicated than sequentially running procedures but they are statistically valid, guaranteeing (3) whenever $x_{[b]} \geq x_i + \delta$ for all $i \in S_D \cup S_A \setminus \{[b]\}$. There is no uniform superiority between sequentially and simultaneously running procedures. However, under a difficult mean configuration such as the SC, the simultaneously running procedures usually perform slightly better.

Simultaneously running procedures can be further improved using the concept of dormancy (Healey et al. 2013), whereby a system may become dormant – halting sampling for that system – when it is found inferior to another system whose feasibility has not been determined yet, returning to contention only if its superior system is eliminated. If the superior system is found to be feasible, then the dormant system will be eliminated. The dormancy framework prevents procedures from collecting unnecessary observations from inferior systems. Other enhancements consider correlation across systems (allowing for the use of CRN) and procedures that minimize the number of switches (setup cost of starting and stopping simulations) between the simulated alternatives.

Other R&S Problems

Other classes of R&S problems include comparison with a standard, multinomial selection, and Bernoulli selection.

The goal of comparison with a standard is to find systems whose expected performance measures are larger (or smaller) than a standard and, if there are any, to find the one with the largest (or smallest) expected performance. In comparison with a standard, the standard is placed in a special status such as a guarantee that no alternative will be selected unless it beats the standard significantly, i.e., the standard is protected as long as an alternative system is not substantially better than the standard. However, if an alternative system does show a significant improvement, then it needs to be selected. For example, a decision maker will be reluctant to implement an alternative due to time and costs associated with replacement of the existing system (the standard) unless performance of an alternative in terms of some measures other than time and costs is significantly better than the standard. The standard is usually denoted as system 0 and there are k alternative systems. Then procedures for comparison with a standard should satisfy the following:

$$\Pr\{\text{select } 0 \mid x_0 \geq x_k\} \geq 1 - \alpha \text{ and}$$
$$\Pr\{\text{select } k \mid x_k - x_0 \geq \delta, x_k - x_{k-l} \geq \delta\} \geq 1 - \alpha.$$

In multinomial selection, the definition of best is the system that is most likely to be the best in a single trial, i.e., the system with the largest probability,

$p_i = \Pr\{X_{im} > X_{\ell m}, \forall \ell \neq i\}$. On the other hand, Bernoulli selection has basic output X_{im} taking either the value 1 (success) or 0 (failure), and the best system is the one with the largest probability of success, $p_i = \Pr\{X_{im} = 1\}$. When the performance measure is a probability, other types of IZ parameters can be considered. Some multinomial selection procedures guarantee the PCS whenever $p_k/p_{k-1} \geq \theta$, where $\theta > 1$ is the smallest p_k/p_{k-1} ratio (relative risk ratio) worth detecting. Bernoulli selection procedures consider at least three types of IZ parameters: the difference δ between probabilities, relative risk ratio, and odds ratio defined as $\frac{p_k/(1-p_k)}{p_{k-1}/(1-p_{k-1})}$; see Kim and Nelson (2006b).

See

- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Simulation Optimization
- ▶ Variance Reduction Techniques in Monte Carlo Methods

References

Andradóttir, S. (2006). An overview of simulation optimization via random search. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science* (pp. 575–616). Oxford, UK: Elsevier.

Andradóttir, S., & Kim, S.-H. (2010). Fully sequential procedures for comparing constrained systems via simulation. *Naval Research Logistics, 57*, 403–421.

Batur, D., & Kim, S.-H. (2006). Fully sequential selection procedures with parabolic boundary. *IIE Transactions, 38*, 749–764.

Batur, D., & Kim, S.-H. (2010). Finding feasible systems in the presence of constrains on multiple performance measures. *ACM Transactions on Modeling and Computer Simulation, 20*, Article No. 13.

Bechhofer, R. E. (1954). A single-sample multiple decision procedure for ranking means of normal populations with known variances. *Annals of Mathematical Statistics, 25*, 16–39.

Boesel, J., Nelson, B. L., & Kim, S.-H. (2003). Using ranking and selection to "clean up" after simulation optimization. *Operations Research, 51*, 814–825.

Butler, J., Morrice, D. J., & Mullarkey, P. W. (2001). A multiple attribute utility theory approach to ranking and selection. *Management Science, 47*, 800–816.

Chen, E. J., & Kelton, W. D. (2005). Sequential selection procedures: Using sample means to improve efficiency. *European Journal of Operational Research, 166*, 133–153.

Chick, S. (2006). Subjective probability and Bayesian methodology. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science* (pp. 225–258). Oxford, UK: Elsevier.

Damerdji, H., & Nakayama, M. K. (1999). Two-stage multiple-comparison procedures for steady-state simulation. *ACM: Transactions on Modeling and Computer Simulation, 9*, 1–30.

Fu, M. C. (2006). Gradient estimation. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science* (pp. 575–616). Oxford, UK: Elsevier.

Goldsman, D., Kim, S.-H., Marshall, W., & Nelson, B. L. (2002). Ranking and selection procedures for steady-state simulation: Perspectives and procedures. *INFORMS Journals on Computing, 14*, 2–19.

Goldsman, D., & Nelson, B. L. (1998). Comparing systems via simulation. In J. Banks (Ed.), *Handbook of simulation* (pp. 273–306). New York: John Wiley.

Gupta, S. S. (1965). On some multiple decision (ranking and selection) rules. *Technometrics, 7*, 225–245.

Healey, C. M. (2010). *Advances in ranking and selection: Variance estimation and constraints*. Doctoral dissertation, H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Healey, C. M., Andradóttir, S., & Kim, S.-H. (2013). Efficient comparison of constrained systems using dormancy. *European Journal of Operational Research, 224*, 340–352.

Hong, L., & Nelson, B. L. (2006). Discrete optimization via simulation using COMPASS. *Operations Research, 54*, 115–129.

Kim, S.-H., & Nelson, B. L. (2001). A fully sequential procedure for indifference-zone selection in simulation. *ACM Transactions on Modeling and Computer Simulation, 11*, 251–273.

Kim, S.-H., & Nelson, B. L. (2006a). On the asymptotic validity of fully sequential selection procedures for steady-state simulation. *Operations Research, 54*, 475–488.

Kim, S.-H., & Nelson, B. L. (2006b). Selecting the best: Theory and method. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in operations research and management science* (pp. 501–534). Oxford, UK: Elsevier.

Law, A., & Kelton, D. (2000). *Simulation modeling and analysis* (3rd ed.). New York: McGraw-Hill.

Nelson, B. L., & Matejcik, F. J. (1995). Using common random numbers for indifference-zone selection and multiple comparisons in simulation. *Management Science, 41*, 1935–1945.

Nelson, B. L., Swann, J., Goldsman, D., & Song, W. M. T. (2001). Simple procedures for selecting the best system when the number of alternatives is large. *Operations Research, 49*, 950–963.

Pichitlamken, J., & Nelson, B. L. (2003). A combined procedure for optimization via simulation. *ACM Transactions on Modeling and Computer Simulation, 13*, 155–179.

Pichitlamken, J., Nelson, B. L., & Hong, L. J. (2006). A sequential procedure for neighborhood selection-of-the-best in optimization via simulation. *European Journal of Operational Research, 173*, 283–298.

Rinott, Y. (1978). On two-stage selection procedures and related probability-inequalities. *Communications in Statistics – Theory and Methods, A7*, 799–811.

Shi, L., & Ólafsson, S. (2000). Nested partitions method for stochastic optimization. *Methodology and Computing in Applied Probability, 2*, 271–291.
Wang, H., & Kim, S.-H. (2013, forthcoming). Reducing the conservativeness of fully sequential indifference-zone procedures. *IEEE on Automatic Control*.

Steady State

A stochastic process is said to be in its steady state if its state probabilities have (essentially) become independent of initial conditions.

See

► Statistical Equilibrium

Steady-state Distribution

Another name for the limiting distribution of a stochastic process.

See

► Limiting Distribution
► Markov Chains
► Markov Processes
► Stationary Distribution
► Statistical Equilibrium

Steepest Descent Method

A fundamental procedure for minimizing a differentiable function of several variables. Central to the method is that the direction of steeptest descent, in moving from one intermediate solution point to another, is along the gradient of the function at the current intermediate solution point.

See

► Nonlinear Programming

Steiner Tree Problem

For a given subset of nodes S from a network with N nodes, the problem is to determine a minimum length (cost) tree that contains all the nodes of S and, optionally, some other nodes from the set N. The Steiner tree problem is often defined on the Euclidean plane where the problem is to find the minimum length (distance) tree that spans a given set of S nodes, where the tree can contain nodes (points) other than those in S.

See

► Minimum Spanning Tree Problem

Stepping-Stone Method

A procedure for solving a transportation problem based on a simplification of the simplex method as applied to the constraint structure that defines a transportation problem. It starts with an initial basic feasible solution and then evaluates, for every nonbasic variable, whether an improved solution can be obtained by introducing one of the nonbasic variables into the basis. The problem is structured into an m-origin by n-destination rectangular matrix of cells in which the cell location (i, j) corresponds to the variable x_{ij} that represents the amount to be shipped from origin i to destination j. The evaluation process for a nonbasic variable x_{ij} starts in cell (i, j) and finds a path (steps) to current basic variable cells so that if x_{ij} does come into the basis, a new feasible solution is generated. Such a path always exists, although degeneracy procedures may be needed to define the path if the current basic solution is degenerate. Associated with the path is a cost that indicates whether or not the new feasible solution would improve (decrease) the value of the objective function. Although useful for pedagogical purposes, the stepping-stone method is not efficient for computer solution. Most computer-based procedures for solving the transportation problem use the transportation (primal-dual) simplex method or special network algorithms.

See

- ▶ Revised Simplex Method
- ▶ Simplex Method (Algorithm)
- ▶ Transportation Problem

Stigler's Diet Problem

A problem formulated by the economist George Stigler in the early 1940s which had as its goal the determination of a minimum cost diet for an adult that met, for a full year, the recommended daily allowances of nutrients and calories, using 77 foods and 1939 prices. It was one of the first problems solved by the simplex method. Stigler's nonoptimal solution cost $39.93, with a diet consisting of wheat flour, evaporated milk, cabbage, spinach, and dried navy beans. The optimal, linear-programming solution cost $39.69 and included wheat flour, cabbage, spinach, beef liver, and dried navy beans.

See

- ▶ Diet Problem
- ▶ Linear Programming
- ▶ Simplex Method (Algorithm)

References

Garille, S. G., & Gass, S. I. (2001). Stigler's diet problem revisited. *Operations Research, 49*, 1–13.

Stigler, G. (1945). The cost of subsistence. *Journal of Farm Economics, 25*, 303–314.

Stochastic Approximation

David W. Hutchison[1] and James C. Spall[2]
[1]RAND Corporation, Santa Monica, CA, USA
[2]The Johns Hopkins University, Applied Physics Laboratory, Laurel, MD, USA

Introduction

Stochastic approximation is an iterative procedure which, under general conditions, employs noisy observations to estimate the root of a function. If this function is the gradient or an estimator for the gradient of a function of interest, the procedure enables the identification of optima.

The prototype application for stochastic approximation is root-finding. Consider a general function $g : \mathbb{R}^p \to \mathbb{R}^p$ defined for $\theta \in \Theta \subseteq \mathbb{R}^p, p > 0$. Then the root-finding problem is to find at least one $\theta = \theta^*$ such that

$$g(\theta) = 0. \tag{1}$$

The exact form of $g(\theta)$ is not known, and whatever observations exist of the function are obscured by noise.

An important special case of (1) is optimization. Consider a differentiable function $L : \mathbb{R}^p \to \mathbb{R}$ defined for $\theta \in \Theta \subseteq \mathbb{R}^p, p > 0$, and suppose g defined above is the gradient of L. Assume $L(\theta)$ is bounded from below and has a unique minimizer denoted by θ^*. The minimization problem is

$$\underset{\theta \in \Theta}{\operatorname{argmin}}\, L(\theta). \tag{2}$$

The exact form of $L(\theta)$ is not known and observations of L (and g, if it can be observed) are obscured by noise.

Let $\hat{\theta}_k$ be an estimate for θ^* at iteration k, a_k a step size at time k, and $G_k\left(\hat{\theta}_k\right) \in \mathbb{R}^p$ some information related to the gradient of the process, also at time k. Choose an initial estimate $\hat{\theta}_0$ and update the estimates following the iterative scheme

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k G_k(\hat{\theta}_k), \;\; k = 0, 1, 2, \cdots. \tag{3}$$

The process in (3), along with a set of conditions for convergence, is the general mathematical model of stochastic approximation.

There are two general methods in stochastic approximation, which differ in their use of gradient information embodied by $G_k\left(\hat{\theta}_k\right)$. Stochastic gradient methods, discussed first, use noisy observations of the gradient, whereas gradient estimation methods use observations of the loss function to estimate the gradient.

Convergence Results

It is of interest to know whether $\hat{\theta}_k$ converges (in a probabilistic sense) to θ^* as k gets large. The convergence of a stochastic approximation algorithm requires that conditions be placed on the objective function, the step size sequence, and the bias and variance of the observed or estimated gradient.

There is not a single "standard" set of conditions; rather, different references offer different sets of conditions that result in almost sure convergence. For the most part, the differences among these sets of conditions lie in weakening one or another condition, usually at the expense of adding elsewhere. These sets of conditions can be broadly categorized as having a statistical perspective (Spall 2003) or an engineering (ODE) perspective (Kushner and Yin 2003; Spall 2003).

The overwhelming majority of what is known about stochastic approximation comes from limit theorems. The robust convergence theory of stochastic approximation is a powerful result. In many cases, it is possible to establish formal convergence of an algorithm or procedure by demonstrating that the problem is equivalent to a form of stochastic approximation.

The conditions for convergence are, in general, global requirements, and thus very broad and difficult to satisfy (and to verify) except in simple cases. However, even if the conditions do not hold globally, there may be a smaller region (still of full dimension p) in which the conditions do hold. In practice, controllable parameters are selected carefully and convergence is assumed.

Practical applications frequently choose other than optimal parameters for algorithms based on (3)—even in violation of the regularity conditions—in order to move the estimate more quickly to the vicinity of θ^* at the sacrifice of more slowly reducing the variability in the estimate. It is often more important to "tweak" the stochastic approximation procedure appropriately for the application being considered in order to obtain satisfactory finite-sample performance. For any particular sequence of estimates, an individual observation of $\hat{\theta}_k$ for some (small) finite value of k could be a poor estimator for θ^*, even when $E\left[\hat{\theta}_k\right]$ is close to θ^*. This is because the variability of the estimate has not yet had a chance to die down.

Also of importance is the probability distribution of the iterate. Having knowledge of the distribution provides key insight into two main aspects of the algorithm: (1) error bounds and stopping procedures and (2) guidance in the choice of algorithm parameters to minimize the deviation of $\hat{\theta}_k$ from θ^*.

Stochastic Gradient Methods

Though analogous to the steepest descent algorithm, stochastic gradient methods are fundamentally different since the deterministic term $\partial L/\partial\theta$ does not equal the stochastic gradient $Y(\theta)$. However, there is an intuitive connection, since $E[Y(\theta)] = \partial L/\partial\theta$ under conventional conditions for convergence.

Robbins-Monro Stochastic Approximation (RMSA)

Denote observations of the gradient by $Y(\theta)$ and model these observations by $Y(\theta) = g(\theta) +$ noise. If the errors have mean zero, then $E[Y(\theta)] = g(\theta)$. Robbins and Monro (1951) studied the problem of finding the roots of an unknown function $g(\theta)$ based on noisy observations of $g(\hat{\theta}_k)$. If g is the gradient of L, the loss function in (2), then this procedure can be used to solve the corresponding minimization problem. After setting $G_k(\hat{\theta}_k) = Y_k(\hat{\theta}_k)$ in (3), the iteration formula for stochastic root-finding is

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k Y_k(\hat{\theta}_k). \tag{4}$$

Asymptotic Properties of RMSA. The earliest analytical results were by Robbins and Monro, who proved mean-square convergence of $\hat{\theta}_k$ to θ^* for their algorithm under mild conditions (thereby implying convergence in probability). A slight tightening of these conditions enabled Blum (1954) to prove almost sure convergence (see also Kushner and Yin 2003). Subsequent results have proved asymptotic normality of the estimate $\hat{\theta}_k$ (Fabian 1968), an asymptotic rate of convergence of $O(k^{-1/2})$ (Chen 1998), convergence probability bounds (Davisson 1970), and conditions that are necessary and sufficient for convergence (Kushner and Yin 2003).

With the conditions for convergence established, and with $\hat{\theta}_k$ generated according to (4), one can prove that $\hat{\theta}_k \xrightarrow{\text{a.s.}} \theta^*$ as $k \to \infty$ (see Kushner and Yin 2003).

Probably the best-known conditions for the convergence of RMSA are those on the gain sequence $\{a_k\}$. The conditions provide a balance between wanting to damp out the noise effects as $\hat{\theta}_k$ nears the solution θ^* $(a_k \to 0)$ and avoiding premature convergence of the algorithm $\left(\sum_{k=0}^{\infty} a_k = \infty\right)$. The scaled harmonic sequence $a/(k+1), a > 0$, is an example of a gain sequence that satisfies the gain conditions. Usually some numerical experimentation is required to choose the best value of the scale factor that appears in the decaying gain sequence. Other conditions important for convergence relate to the smoothness of $g(\theta)$, the relative magnitude of the noise, and the position of the initial condition (Spall 2003).

To obtain a non-degenerate limiting distribution, scale the error $\hat{\theta}_k - \theta^*$. If the step size function takes the form $a_k = a/(k+1)^{\beta}$ for $\frac{1}{2} < \beta \leq 1$ (and satisfies certain regularity conditions), one can show that the distribution of the scaled error is asymptotically normal:

$$k^{\beta/2}(\hat{\theta}_k - \theta^*) \xrightarrow{\text{dist}} N_p(0, \Sigma^*) \quad \text{as } k \to \infty,$$

where Σ^* is a covariance matrix determined by the coefficients of the sequence $\{a_k\}$ and by the Hessian of $L(\theta)$ at $\theta = \theta^*$ (Spall 2003, p. 112). One informal but natural interpretation of this fact is that $\hat{\theta}_k$ is approximately multivariate normal with mean θ^* and covariance Σ^*/k^{β}. Various special cases of this result dealing with the situation $\beta = 1$ can be found (Ljung et al. 1992).

Gradient Estimation Methods

Gradient estimation algorithms were first addressed by Kiefer and Wolfowitz (1952) by using finite differences to estimate the gradient of the function $L(\theta)$. These algorithms demonstrate the convergence properties of stochastic gradient algorithms using only measurements of the loss function.

Viewed asymptotically, when speed is measured by the number of iterations, gradient-based algorithms converge faster than those using gradient estimates (convergence being measured in terms of the deviation of the estimate from the true optimal parameter vector). The optimal rate of convergence for gradient-based algorithms is $O(k^{-1/2})$, while for algorithms based on gradient estimates the rate is $O(k^{-1/3})$, where k represents the number of iterations (Spall 2003).

One cannot say in general that a stochastic gradient algorithm is superior to a gradient estimation algorithm even though the stochastic gradient algorithm has a faster asymptotic rate of convergence. However, if direct gradient information is readily available, it is generally advantageous to use this information in the optimization process.

Denote the estimate of the gradient $\partial L/\partial\theta$ at $\hat{\theta}_k$ by $\hat{g}_k(\hat{\theta}_k)$. Then the general recursive procedure (3) with $G_k(\hat{\theta}_k) = \hat{g}_k(\hat{\theta}_k)$ is

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \hat{g}_k(\hat{\theta}_k). \tag{5}$$

Let $y(\theta)$ denote a measurement of $L(\theta)$ (i.e., $y(\theta) = L(\theta) +$ noise) and Δ a perturbation. One-sided gradient estimates involve the measurements $y(\hat{\theta}_k)$ and $y(\hat{\theta}_k + \Delta)$, while two-sided gradient estimates involve the measurements $y(\hat{\theta}_k \pm \Delta)$.

Under appropriate conditions, the iteration in (5) will converge to θ^* in some stochastic sense, usually a.s. (Kushner and Clark 1978). Typical convergence conditions are similar to those mentioned above for the RMSA algorithm.

Finite-Difference Stochastic Approximation (FDSA)

Each component of $\hat{\theta}_k$ is perturbed one at a time, and corresponding measurements y are obtained; each component of the gradient estimate is the difference in the y values divided by the difference interval. This is a standard approach to estimating gradient vectors and is motivated by the definition of the gradient as a vector of p partial derivatives. The ith component of $\hat{g}_k(\hat{\theta}_k), i = 1, 2, \ldots, p$, for a two-sided finite difference estimate is typically given by

$$\hat{g}_{ki}(\hat{\theta}_k) = \frac{y(\hat{\theta}_k + c_k e_i) - y(\hat{\theta}_k - c_k e_i)}{2c_k},$$

where e_i denotes the ith unit basis vector (Kiefer and Wolfowitz 1952).

Simultaneous Perturbation Stochastic Approximation (SPSA)

All components of $\hat{\theta}_k$ are randomly perturbed together (i.e., "simultaneously") to obtain two

measurements y, and each component of $\hat{g}_k(\hat{\theta}_k)$ is the difference in the y values divided by the difference interval. The ith component of $\hat{g}_k(\hat{\theta}_k), i = 1, 2, \ldots, p$, for a two-sided simultaneous perturbation is

$$\hat{g}_{ki}(\hat{\theta}_k) = \frac{y(\hat{\theta}_k + c_k\Delta_k) - y(\hat{\theta}_k - c_k\Delta_k)}{2c_k\Delta_{ki}},$$

where the random perturbations, $\Delta_k = [\Delta_{k1}, \Delta_{k2}, \ldots, \Delta_{kp}]^T$, satisfy the distributional conditions below.

Note that the number of loss function measurements y needed in each iteration of FDSA grows with p, while SPSA requires only two loss function measurements per iteration. Thus there is the potential for SPSA to achieve a savings over FDSA in the total number of measurements required to estimate θ when p is large (Spall 2003).

Asymptotic Properties of SPSA. The conditions for convergence of the SPSA algorithm are somewhat different from those of RMSA. Conditions must be imposed on the gain sequences $\{a_k\}$ and $\{c_k\}$, the distribution of Δ_k, and the statistical relationship of Δ_k to the measurements y (Spall 2003). The gain sequences $\{a_k\}$ and $\{c_k\}$ both must go to zero at rates neither too fast nor too slow, $L(\theta)$ should be sufficiently smooth near θ^* (several times differentiable), and the Δ_{ki} should be independent and symmetrically distributed about zero with finite inverse moments $E[|\Delta_{ki}|^{-1}]$ for all k and i. One distribution that satisfies these conditions is the symmetric Bernoulli distribution on $\{-1, 1\}$.

Under the conditions outlined in Spall, (2003, p. 204) one can show

$$k^{\beta/2}(\hat{\theta}_k - \theta^*) \xrightarrow{dist} N_p(\mu, \Sigma^*) \quad \text{as } k \to \infty.$$

In general, $\mu \neq 0$, in contrast to the asymptotic normality results for RMSA. The optimal rate of convergence for SPSA is $O(k^{-1/3})$, compared to $O(k^{-1/2})$ for the RMSA algorithm. There are exceptions to this result; see Kleinman, Spall, and Naiman (1999).

The efficiency of SPSA (relative to FDSA) depends on the shape of $L(\theta)$, the sequences $\{a_k\}$ and $\{c_k\}$, and the distributions of the Δ_{ki} and measurement noise terms. For most practical problems, SPSA is asymptotically more efficient than FDSA (Spall 2003). In particular, the total number of loss measurements y to achieve convergence in SPSA is proportional to $1/p$ the number needed in FDSA (Spall 2003, Chap. 7).

Extensions to Standard Approaches

Constrained Problems

A simple variation on the form in (3) includes a projection operator Π_Θ that maps solutions outside the constraint set Θ back to its nearest point in Θ. This approach for the Robbins-Monro algorithm is discussed in Kushner and Yin, (2003). In this case, (4) becomes

$$\hat{\theta}_{k+1} = \Pi_\Theta[\hat{\theta}_k - a_k Y(\hat{\theta}_k)].$$

There are implementations of SPSA for constrained optimization as well. See Sadegh (1997) for a projection approach, Wang and Spall (2008) for a penalty approach, and Bhatnagar, Hemachandra, and Mishra (2011) for a Lagrange multiplier approach.

Iterate Averaging

Averaging is an important development in stochastic approximation. The approach—sometimes referred to as Ruppert-Polyak stochastic approximation—was originally introduced as a means to improve the efficiency of the usual stochastic approximation process. Theory supporting the validity of this approach is found in Polyak and Juditsky (1992).

There are several variations, but the basic idea is to replace $\hat{\theta}_k$ as the current "best" estimate of θ^* after k iterations with the average

$$\bar{\theta}_k = \frac{1}{k+1}\sum_{j=0}^{k}\hat{\theta}_j.$$

A variation on this method is to compute a "sliding window" average based on the last m estimates:

$$\bar{\theta}_k = \frac{1}{m}\sum_{j=k-m+1}^{k}\hat{\theta}_j.$$

The advantage of a sliding window approach is that the averaging is focused on the later estimates which are presumably in a neighborhood of θ^*. Further variations on these ideas can be devised.

Recent work has shown some limitations and cautions regarding the method (see, for example, Maryak (1997) and Spall (2003, pp. 117–119)), and the method seems best suited for the class of problems where the estimates loiter approximately randomly in a neighborhood of θ^*.

A further modification to the averaging approach is to use $\bar{\theta}_k$ (together with $\hat{\theta}_k$) in a modified form of the RMSA iteration. This is referred to as the feedback approach (Kushner and Yang 1995), and occasionally yields improvements.

Adaptive Estimation

Kesten (1958) developed an adaptive algorithm for scalar θ that looks at the signs of the difference $\hat{\theta}_{k+1} - \hat{\theta}_k$. Frequent sign changes are seen as an indication that $\hat{\theta}_k$ is near θ^*, while if signs are not changing, it is an indication that $\hat{\theta}_k$ is far from θ^*. A larger step size a_k is used if there are no sign changes and a smaller a_k is used if the signs change frequently. A multivariate extension of Kesten's idea is described in Delyon and Juditsky (1993).

Second-Order Algorithms

There are stochastic analogs of the second-order Newton-Raphson search. The scalar gain a_k is replaced by a matrix H_k that approximates the unknown true inverse Hessian matrix corresponding to the current data point's contribution to the loss function. An accelerated form of SPSA extends the algorithm to include second-order (Hessian) effects. Recent results in adaptive SPSA are discussed in Spall (2003).

Ruppert (1985) describes an approach where the Hessian is estimated by taking finite differences of a gradient measurement. Spall (2000, 2009) gives a more efficient approach to general Hessian estimation based on simultaneous perturbations.

Joint Parameter and State Evolution

A generalization of the stochastic approximation process replaces $G_k(\hat{\theta}_k)$ by $G_k(\hat{\theta}_k, x_k)$, where x_k represents a state vector related to the system being optimized. It is typical to assume that x_k evolves according to Markov transition probabilities.

Time-Varying Loss Functions

The loss function $L(\theta)$ may itself be a function of k. It is assumed that, while $L_k(\theta)$ may change shape with k, the underlying minimum θ^* is either constant for all k or fixed in the limit as $k \to \infty$.

Stopping Rules

The need for a stopping rule for stochastic approximation was recognized by Kiefer and Wolfowitz (1952). Three broad categories of stopping methodologies include sequential methods, Monte Carlo methods, and relaxation methods. Sequential and Monte Carlo methods are discussed below. A review of relaxation methods can be found in Hutchison (2009).

Sequential Methods

Chow and Robbins (1965) developed a method to sequentially determine a bound on the mean of a continuous scalar random variable with unknown variance. They suggested the following rule: stop when the length of the confidence interval based on asymptotic normality of the sample means is smaller than 2δ for some $\delta > 0$. Since this initial work, much of the effort in stopping stochastic approximation has been on estimating the distribution of $\hat{\theta}_k$ in order to apply the Chow-Robbins criterion.

The idea is to fix a level of significance α, estimate the distribution of $\hat{\theta}_k$, and form a confidence region based on the estimated distribution and α. This is done successively as the algorithm steps through the iterations. As the sequence converges, the dispersion of the estimated distribution gets smaller, and the confidence region follows suit. The algorithm is stopped when the size of the $1 - \alpha$ confidence region is small enough.

Theory for the multi-dimensional case can be found in Pflug (1996). The general validity of the approach is demonstrated in Glynn and Whitt (1992).

The method requires knowledge of Σ^*, the covariance matrix of the limiting normal distribution of the scaled error. Using the step size sequence $a_k = a/(k+1)^\beta, \frac{1}{2} < \beta \leq 1$, the covariance matrix Σ^* is computed from one of the following matrix equations:

$$\begin{aligned} \left(aH(\theta^*) - \tfrac{1}{2}I\right)\Sigma^* + \Sigma^*\left(aH(\theta^*) - \tfrac{1}{2}I\right)^{\mathrm{T}} &= a^2 C(\theta^*), \\ &\text{for } \beta = 1; \\ H(\theta^*)\Sigma^* + \Sigma^* H(\theta^*) &= aC(\theta^*), \text{ for } \frac{1}{2} < \beta < 1. \end{aligned} \tag{6}$$

See, for example, Pflug (1996).

Approaches that use the asymptotic distribution as a proxy for the true distribution of $\hat{\theta}_k$ to stop a stochastic approximation may be unsatisfactory for small (finite) samples. Alternatively, one can attempt to directly estimate the true distribution of $\hat{\theta}_k$ or a simpler (but similar) surrogate distribution. Under certain conditions, surrogate-based probability calculations are close to the actual probabilities (Hutchison and Spall 2009).

Sequential procedures tend to perform well when the run lengths are relatively long (Glynn and Whitt 1992). The procedure is less reliable with small samples.

Monte Carlo Methods

A simple idea for stopping is to conduct multiple trials to generate the information necessary for stopping. There are two subcategories of this method: iterate sampling and sample path sampling.

Iterate sampling methods. Iterate sampling obtains repeated observations of $\hat{\theta}_k$ based on having arrived at $\hat{\theta}_{k-1}$. Iterate sampling is not directly useful for determining the distribution of $\hat{\theta}_k$, though it may be used to determine bounds on the next step of the current iteration, on an estimator of the loss, $\hat{L}(\hat{\theta}_k)$, or of the gradient, $\hat{g}(\hat{\theta}_k)$. The most direct approach is to sample at each iteration to obtain information that can be used as the basis for stopping.

Sample path methods. Sample path methods obtain repeated observations of $\hat{\theta}_k$ by running m independent applications of the stochastic approximation process.

Independent copies of the stochastic approximation process may become concentrated in a small region (nominally a ball). As each process converges to θ^*, the tails of the sequences of estimates get "close together." The iteration is stopped when the tails are "close enough."

More direct is to use sample path sampling to estimate the asymptotic distribution of $\hat{\theta}_k$, a hypothesized distribution, or the true distribution. For example, an improvement over using the asymptotic covariance structure in (6) is to use an estimator for the true covariance of $\hat{\theta}_k$, Σ_k. Hsieh and Glynn (2002) apply this approach by taking m sample paths of length k to estimate the covariance matrix Σ_k.

Concluding Remarks

Although stochastic approximation methods have the potential to treat a broader class of problems than many traditional deterministic techniques, their application can be a challenge. A problem common to all stochastic approximation techniques is that values must be specified for the algorithm's tunable coefficients. All stochastic approximation techniques have such coefficients. These coefficient values are typically problem dependent and can have a significant effect on the performance of an algorithm.

Stochastic approximation allows for the treatment of problems such as global optimization and noisy loss-function evaluations that arise frequently in areas such as network analysis, neural network training, image processing, nonlinear control, and simulation optimization. Stochastic approximation addresses a broader range of problems than possible with only standard deterministic methods.

See

- ▶ Neural Networks
- ▶ Perturbation Analysis
- ▶ Score Functions
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Simulation Optimization

References

Bhatnagar, S., Hemachandra, N., & Mishra, V. (2011). Stochastic approximation algorithms for constrained optimization via simulation. *ACM Transactions on Modeling and Computer Simulation*, *21*(3), Article 15 (22 pages).

Blum, J. R. (1954). Approximation methods which converge with probability one. *Annals of Mathematical Statistics, 25*(2), 382–386.

Chen, H.-F. (1998). Convergence rate of stochastic approximation algorithms in the degenerate case. *SIAM Journal on Control and Optimization, 36*(1), 100–114.

Chow, Y. S., & Robbins, H. (1965). On the asymptotic theory of fixed-width sequential confidence intervals for the mean. *Annals of Mathematical Statistics, 36*(2), 457–462.

Davisson, L. D. (1970). Convergence probability bounds for stochastic approximation. *IEEE Transactions on Information Theory, 16*(6), 680–685.

Delyon, B., & Juditsky, A. (1993). Accelerated stochastic approximation. *SIAM Journal on Optimization, 3*(4), 868–881.

Fabian, V. (1968). On asymptotic normality in stochastic approximation. *Annals of Mathematical Statistics, 39*(4), 1327–1332.

Glynn, P. W., & Whitt, W. (1992). The asymptotic validity of sequential stopping rules for stochastic simulations. *The Annals of Applied Probability, 2*(1), 180–198.

Hsieh, M.-H., & Glynn, P. W. (2002). Confidence regions for stochastic approximation algorithms. In E. Yücesan, C.-H. Chen, J. L. Snowdon, & J. M. Charnes (Eds.), *Proceedings of the 2002 Winter Simulation Conference* (Vol. 1, pp. 370–376). San Diego, CA: IEEE.
Hutchison, D. W. (2009). *Stopping times and confidence bounds for small-sample stochastic approximation algorithms*. Ph.d. dissertation, The Johns Hopkins University, Baltimore, MD.
Hutchison, D. W., & Spall, J. C. (2009). Stopping small-sample stochastic approximation. In *Proceedings of the 2009 American Control Conference (ACC09)* (pp. 26–31). St. Louis, MO: IEEE.
Kesten, H. (1958). Accelerated stochastic approximation. *Annals of Mathematical Statistics, 29*(1), 41–59.
Kiefer, J., & Wolfowitz, J. (1952). Stochastic estimation of the maximum of a regression function. *Annals of Mathematical Statistics, 23*(3), 462–466.
Kleinman, N. L., Spall, J. C., & Naiman, D. Q. (1999). Simulation-based optimization with stochastic approximation using common random numbers. *Management Science, 45*(11), 1570–1578.
Kushner, H. J., & Clark, D. S. (1978). *Stochastic approximation methods for constrained and unconstrained systems* (Vol. 26). New York: Springer-Verlag.
Kushner, H. J., & Yang, J. (1995). Stochastic approximation with averaging and feedback: Rapidly convergent on-line algorithms. *IEEE Transactions on Automatic Control, 40*(1), 24–34.
Kushner, H. J., & Yin, G. G. (2003). *Stochastic approximation and recursive algorithms and applications* (2d ed., No. 35). New York: Springer-Verlag.
Ljung, L., Pflug, G. C., & Walk, H. (1992). *Stochastic approximation and optimization of random systems*. Basel: Birkhäuser.
Maryak, J. L. (1997). Some guidelines for using iterate averaging in stochastic approximation. In *Proceedings of the 36th Conference on Decision and Control* (Vol. 3, pp. 2287–2290). San Diego, CA: IEEE.
Pflug, G. C. (1996). *Optimization of stochastic models: The interface between simulation and optimization*. Boston: Kluwer Academic.
Polyak, B. T., & Juditsky, A. B. (1992). Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization, 30*(4), 838–855.
Robbins, H., & Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics, 22*(3), 400–407.
Ruppert, D. (1985). A Newton-Raphson version of the multivariate Robbins-Monro procedure. *The Annals of Statistics, 13*(1), 236–245.
Sadegh, P. (1997). Constrained optimization via stochastic approximation with a simultaneous perturbation gradient approximation. *Automatica, 33*(5), 889–892.
Spall, J. C. (2000). Adaptive stochastic approximation by the simultaneous perturbation method. *IEEE Transactions on Automatic Control, 45*(10), 1839–1853.
Spall, J. C. (2003). *Introduction to stochastic search and optimization: Estimation, simulation, and control*. Hoboken, NJ: John Wiley and Sons.
Spall, J. C. (2009). Feedback and weighting mechanisms for improving Jacobian estimates in the adaptive simultaneous perturbation algorithm. *IEEE Transactions on Automatic Control, 54*(6), 1216–1229.
Wang, I.-J., & Spall, J. C. (2008). Stochastic optimisation with inequality constraints using simultaneous perturbations and penalty functions. *International Journal of Control, 81*(8), 1232–1238.

Stochastic Duel

A stochastic duel is a model of combat (originally between two individuals, expanded to two sides with finite numbers of individuals) which emphasizes the random nature of combat and finite attrition calculations.

See

► Battle Modeling

Stochastic Dynamic Programming

Dynamic programming setting in which the transitions and/or costs/rewards are stochastic. The resulting mathematical model is usually a Markov decision process.

See

► Approximate Dynamic Programming
► Dynamic Programming
► Markov Decision Processes

Stochastic Input Model Selection

Bahar Biller and Alp Akcay
Carnegie Mellon University, Pittsburgh, PA, USA

Introduction

Input modeling is the selection of a probability distribution to capture the uncertainty in the input

environment of a stochastic system. Example applications of input modeling include the representation of the randomness in the time to failure for a machining process, the time between arrivals of calls to a call center, and the demand received for a product of an inventory system. Building simulations of stochastic systems requires the development of input models that adequately represent the uncertainty in such random variables. Since there are an abundance of probability distributions that can be used for this purpose, a natural question to ask is how to identify the probability distribution that best represents the particular situation under study. For example, is the exponential distribution a reasonable choice to represent the time to failure for a machining process, or is it better to use an empirical distribution function obtained from the historical time-to-failure data? Recognizing the fact that there is no true input model waiting to be found, the goal of stochastic input modeling is to obtain an approximation that captures the key characteristics of the system inputs.

The development of a good input model requires the collection of as much information as possible about the relevant randomness in the system as well as the historical data consisting of the past realizations of the random variables of interest. In the presence of a data set, the input model can be identified by fitting a probability distribution to the historical data. However, it may be difficult and/or costly to collect data for the stochastic system under study; it can also be impossible to properly collect any data at all such as when the proposed system does not exist. In the absence of historical data, any relevant information (e.g., expert opinion and the conventional bounds suggested by the underlying physical situation) can be used for input modeling. This article addresses the key issues that arise in stochastic input modeling both in the presence and in the absence of historical data.

The first step in input modeling is to identify the sources of randomness in the input environment of the system under study. Many stochastic systems contain multiple sources of uncertainty, e.g., the completion time of an item on a particular machine, the potential breakdown of the machine, and the percentage of defective items produced by the machine might be among the sources of uncertainty in a manufacturing setting. Throughout, the random vector $\mathbf{X} = (X_1, X_2, \ldots, X_K)'$ is used to represent the collection of K different inputs of a stochastic system, where X_k is the random variable denoting the kth system input. The K components of this random vector might also be correlated with each other. Therefore, the stochastic properties of the random inputs X_k, $k = 1, 2, \ldots, K$, are captured in the joint probability distribution function of $\mathbf{X}$. Selecting a joint distribution function to capture the randomness in $\mathbf{X}$ is called multivariate input modeling. It might be the case that the machine breakdown probability is positively correlated with the job completion time on the machine. In this case, stochastic input modeling refers to the specification of a bivariate probability distribution function for the joint representation of the machine breakdown probability and the job completion time. It might also be the case that there is a single source of randomness ($K = 1$), or the random component X_k is independent of the remainder of the components, X_l, $l = 1, 2, \ldots, k-1, k+1, \ldots, K$. In both of these cases, the input-modeling problem reduces to univariate input modeling which selects a univariate distribution for the random component of interest. For example, if the percentage of the defective items is known to be independent of both the job completion time and the potential breakdown of the machine, then the defective-item percentage is represented with a univariate distribution. Furthermore, univariate input modeling, despite failing to capture the dependencies among different random components, is often a good start towards solving the multivariate input-modeling problem. Although the main focus here will be on univariate modeling, the key issues that arise in multivariate modeling will also be addressed.

Univariate Input Modeling

Assuming an independent and identically distributed (i.i.d.) input process, this section treats univariate input modeling both in the presence and in the absence of historical data. Additionally, an autocorrelated input process is considered, with focus on captuning the temporal dependence.

Input Modeling with Historical Data

The input-modeling problem of this section assumes the availability of i.i.d. historical data points x_i, $i = 1, 2, \ldots, n$ of length n, and describes how to use this data set for estimating $F(\cdot;\mathbf{\Psi})$, the underlying cumulative distribution function (c.d.f.) of the random variable X and the unknown parameter vector $\mathbf{\Psi}$.

Preliminary Analysis of the Historical Data

It is possible that the available historical data points are recorded imprecisely or grouped with the observations of the other random variables in the input environment of the stochastic system under study. It might also be the case that the data set is available in an order other than when the values were observed. Such data characteristics do not allow the simulation practitioner to check whether being i.i.d. is a reasonable assumption for the underlying input process; see Vincent (1998) for further examples that turn input modeling into a challenging problem. The input-modeling techniques of this section, on the other hand, assume that the historical data on hand are statistically i.i.d.; therefore, it is critical to perform a preliminary analysis of the historical data and verify the i.i.d. assumption before the implementation of the input-modeling techniques to be discussed shortly.

A graphical method that can be used for assessing the independence of the historical input data is the correlation plot. It shows the sample correlations for various lags; note that a lag ℓ correlation is the correlation between data points that are ℓ values apart. Vincent (1998) reports that lags of size one through ten are the most informative about a sample, whereas lags over 20 are non informational. For an i.i.d. historical data set, the sample correlations are expected to be small in magnitude and clustered around zero, with both negative and positive estimated values.

Another graphical method that can be used for the same purpose is the scatter diagram plotting the data pairs x_i and x_{i+1} for $i = 1, 2, \ldots, n-1$ on each of its axes. If the historical data points are independently distributed, then they are randomly scattered. If the data points lie along a diagonal line with a positive (negative) slope, then the scatter diagram suggests a positively (negatively) autocorrelated input process.

In addition to the assessment of the independence assumption, it should also be checked whether each data point comes from the same probability distribution; i.e., the (unconditional) probability distribution of the input process does not change with time. A way of doing this is to analyze the data for any discernible increasing or decreasing patterns over time; see Vincent (1998) for further discussion on assessing the stationarity of the underlying input process.

The next step in the preliminary analysis of the historical data is the calculation of the summary statistics such as the sample minimum $x_{(1)}$ and the sample maximum $x_{(n)}$ for the range estimation, the sample mean $\bar{x} = \sum_{i=1}^{n} x_i/n$ as the measure of the central tendency, the sample variance $s^2 = \sum_{i=1}^{n} (x_i - \bar{x})/(n-1)$ and the sample coefficient of variation $s/\bar{x}$ as the measures of variability, the sample coefficient of skewness $\sum_{i=1}^{n} (x_i - \bar{x})^3/s^3$ as the measure of asymmetry, and the sample coefficient of kurtosis $\sum_{i=1}^{n} (x_i - \bar{x})^4/s^4$ as the measure of peakedness and tail weight. In addition, the quantile summary and the histogram of the historical data show the level of asymmetry in the empirical density function. However, the use of the histogram for input modeling might lead to different conclusions based on the number of the bins used; therefore, the histograms should always be constructed for various bin sizes.

The summary statistics obtained from the preliminary analysis of the historical data can be used for identifying the form of the underlying probability density function. For example, a sample coefficient of variation close to 1, along with a histogram with an exponential decay, suggests exponential distribution as an input model. A sample coefficient of variation greater than 1 with a right-skewed histogram indicates that a lognormal distribution can be a potential input model, while a symmetric histogram or a sample coefficient of skewness close to zero indicates normal or Student's t distribution as an appropriate candidate for input modeling. Such a preliminary analysis allows the simulation practitioner to gain insights about the key characteristics of the historical input data. A good resource for further discussion on the initial analysis of the historical data for input modeling is Law (2007).

Model Fitting and Parameter Estimation

Standard families of distributions (e.g., beta, binomial, Erlang, exponential, gamma, lognormal, normal,

Poisson, triangular, uniform, or Weibull) are often the immediate choices in the selection of a distribution for the input data. These standard distributions can be classified as discrete and continuous, or with bounded and unbounded ranges. Detailed treatments of standard discrete and continuous distributions are available, respectively, in Johnson, Kemp and Kotz (2005) and Johnson, Kotz and Balakrishnan (1995).

The identification of the best fit (i.e., the probability distribution that provides the best representation for the underlying input process) often requires not only the use of historical data for model fitting but also a good understanding of the source of randomness in the system. Most probability distributions are invented to present a particular physical situation. If the physical basis for a distribution is well understood, that knowledge can be used to match it to the situation being modeled. For example, if the number of patient visits to an emergency room satisfies the assumptions of the Poisson process (i.e., the patients arrive one at a time, the number of arrivals in a time interval is independent of the number of arrivals in an earlier time interval and the times at which these arrivals occur, and the arrival rate of patients does not depend on the time of day), then the exponential distribution can be selected for modeling the interarrival times between patient arrivals. The lognormal distribution would be a potential input model to represent the rate of return on an investment when interest is compounded, because the rate of return in this particular case can be thought of as the product of a number of component processes. The use of the physical basis for distribution selection in this manner is especially important for input modeling in the absence of historical data.

In many practical situations, however, it is not easy to choose and justify an input distribution based on the source of randomness in the system. Moreover, the standard families of distributions might not adequately represent the probabilistic behavior of the input processes. In such cases, the use of flexible families of distributions for input modeling might allow the simulation practitioner to capture the key characteristics of the available data by fine-tuning the shape of the fitted input distribution. Some well-known flexible distribution systems include the curves proposed by Pearson (1895), the generalized lambda distribution (Ramberg and Schmeiser 1974), and the Schmeiser-Deutsch class of distributions (Schmeiser and Deutsch 1977). Other widely-used distribution systems for input modeling are the generalized beta family of distributions and the Bézier distribution family, along with the Johnson translation system; see Kuhl et al. (2010) for a review of these distribution systems with a focus on stochastic simulation.

Independent of whether a standard distribution or a flexible distribution is chosen, the next step in input modeling is to obtain $\hat{\boldsymbol{\Psi}}$, an estimate of the unknown parameter vector $\boldsymbol{\Psi}$, which minimizes the distance between the hypothesized distribution function $F_h(\cdot;\hat{\boldsymbol{\Psi}})$ and the empirical distribution function of the historical data. Assuming that the functional form of $F_h(\cdot;\hat{\boldsymbol{\Psi}})$ is known, the methods of maximum likelihood, least squares, and moment matching are the three widely-used estimation techniques for predicting a value for $\hat{\boldsymbol{\Psi}}$. Specifically, the maximum likelihood method estimates $\hat{\boldsymbol{\Psi}}$ by maximizing the likelihood function of the historical data, while the least-squares method minimizes the sum of squared residuals, each of which is a difference between an observed value and a fitted quantile. The method of moment matching, on the other hand, predicts $\hat{\boldsymbol{\Psi}}$ by matching the moments of the hypothesized distribution $F_h(\cdot;\hat{\boldsymbol{\Psi}})$ to the sample moments of the historical data set; a good resource for details on these estimation techniques is Rohatgi and Saleh (2001).

In situations where no parametric distribution provides a good fit for the historical data, the empirical distribution function can be chosen as the input model. Specifically, the empirical distribution function $\hat{F}$ is the c.d.f. constructed by assigning probability $1/n$ to each of the n data points. It is an unbiased, consistent, and asymptotically normal estimator of the c.d.f. F (Rohatgi and Saleh 2001). A limitation of the empirical distribution is that it ignores the possibility of a realization that does not appear in the historical data set. The common practice is to fill such gaps by linearly interpolating between the sorted values of the historical data. Also, the range of the empirical distribution is limited to that of the historical data set. However, this limitation can be overcome by extending one or both of the empirical-distribution tails with known distribution functions. A distribution that is often used for this purpose is the exponential distribution. A more detailed presentation of the empirical distribution function with a focus on simulation input modeling can be found in Vincent (1998).

Assessing the Goodness of the Fit

After the estimation of the distribution parameters, the next step is to assess how well the resulting fit captures the key distributional characteristics of the historical data. This can be accomplished by using the statistical goodness-of-fit tests and the graphical (heuristic) techniques.

Chi-square, Kolmogorov-Smirnov, and Anderson-Darling tests are the three of the statistical goodness-of-fit tests available in simulation software packages for input modeling. Specifically, the chi-square test provides a formal comparison between the histogram (or the line graph) of the historical data and the fitted density (or mass) function. The major drawback of this test is its sensitivity to the number of intervals (data groups) used for the construction of the histogram. The Kolmogorov-Smirnov and Anderson-Darling tests compare the empirical distribution function to the fitted distribution function. Neither of these tests require the grouping of the data, while the Anderson-Darling test with the higher power detects, in particular, the discrepancies in the distribution tails. It is important to note that each of these tests is unlikely to reject any distribution when there is little data, and is likely to reject every distribution when there is a lot of data. Keeping this caveat in mind, the goodness-of-fit test statistics should be interpreted as a recommendation to accept or reject an input model, but not as a definite rule to be strictly followed.

In addition to the use of goodness-of-fit tests for input modeling, graphical methods are recommended as advisory devices to examine the fits. The graphical tools that can be used for this purpose include the frequency comparison plot, the density-function-differences plot, the probability-probability (P-P) plot, and the quantile-quantile (Q-Q) plot. Specifically, the frequency comparison plot is a graphical comparison of the histogram of the historical data with the estimated density function, while the density-function-differences plot compares the empirical distribution function to the fitted distributed function by plotting their differences over the available data range. The P-P plot, on the other hand, plots the hypothesized (estimated) distribution function $F_h(x_{(i)}; \hat{\boldsymbol{\Psi}})$ against $(i - 0.5)/n$ with $x_{(i)}$ denoting the ith smallest historical data point, while $F_h^{-1}((i-0.5)/n; \hat{\boldsymbol{\Psi}})$ is plotted against $x_{(i)}$ in the Q-Q plot for $i = 1,2,\ldots,n$. Thus, the Q-Q plot (P-P plot) amplifies the differences between the tails (middles) of the hypothesized and sample distribution functions. Both the P-P plot and the Q-Q plot are expected to be approximately linear when the hypothesized distribution function $F_h(\cdot; \hat{\boldsymbol{\Psi}})$ is a good fit. Since it is important that the input model adequately captures the tail behavior of the historical input data, the use of the Q-Q plot is highly recommended for input-model building.

To summarize, identification of a good input model starts with the preliminary analysis of the historical input data, continues with the use of an estimation method for determining the distribution function and its parameters, and ends with the evaluation of the goodness of the resulting fit. The proprietary simulation software packages often include built-in modules that follow these steps to identify the input distribution that best fits the historical data; see Swain (2011) for a survey on such simulation software and their input-modeling capabilities; this survey is usually updated biennially in *OR/MS Today*. Nevertheless, the simulation practitioner should not just rely on the best fit identified by the software; if there is a strong physical basis for a particular distribution choice, then its selection as the input model should be seriously considered even if it is not the best fit.

Input Modeling in the Absence of Historical Data

The setting where no data are available to select a distribution or assess the fit is now considered. This might occur due to time and budget restrictions and/or the challenges in data collection. It might also be the case that the goal of the simulation study is to investigate the impact of design changes on the system performance, so that the (proposed) system does not exist yet. Therefore, the input model has to be developed by using any available information about the underlying input process. For example, if a distributor receives orders from a large number of independent retailers, then the Central Limit Theorem suggests the selection of the normal distribution as the input model for representing the total demand. Similarly, the number of defective items in a shipment of fixed size can be modeled by a binomial distribution when the probability of being defective is the same for each item. In the absence of historical data, the physical limitations of the

underlying process can also be useful for input modeling. For example, if the repair of a machine requires at least 6 hours to run a diagnostic check and the machine is replaced by a new one if it cannot be repaired in 3 days, then the input model for the machine repair time should assume values between 6 hours and 3 days.

The knowledge and experience of the experts (i.e., the people familiar with the system being studied) are called expert opinion. It is an important source of subjective information necessary for developing an input model in the absence of historical data. A way of extracting expert opinion is to use breakpoints; i.e., the numerical values that the input random variable can take and the chances of the random variable being higher or lower than the breakpoints. In this case, the input model is built using as many breakpoints as can be confidently obtained, especially near the extreme values of X. The breakpoints method is useful for modeling inputs with many possible outcomes. However, it is possible that the expert can only provide limited information about the properties of the input random variables such as the lower/upper bound, the most likely value, and the mean value. A way of building an input model in such a case is to consider the use of uniform, triangular, PERT, and beta distributions for incorporating expert opinion into the input-model development. Specifically, the uniform distribution can be used when the expert provides both a lower bound and an upper bound to the values the random variable can take. However, the expert can also provide a most likely value which can be incorporated into the input model as the mode of the triangular or PERT distribution. If the expert additionally provides an average value for the input, then all of the available expert opinion can be used for constructing a beta distribution. A drawback of using the beta distribution for input modeling is that it might be difficult for the expert to differentiate between the most likely and average values, but this difficulty can be overcome by modeling the input with a lognormal or Weibull distribution. The construction of these two distributions requires expert opinion about the location parameter, the most likely value, and the q-quantile of the input distribution. While the location parameter can be interpreted as the lower bound to the values the input random variable can take, the scale and shape parameters are functions of the mean and the q-quantile.

Autocorrelated Input Process

The focus switches now to an autocorrelated input process that exhibits temporal dependence, specifically to input modeling for the stationary univariate time series denoted by $\{X_t;\ t = 1, 2, \ldots\}$. An example of such an input process is the sequence of week-to-week quantities ordered by a distributor, when modeling the weekly orders as independent random variables misses the week-to-week dependence (autocorrelation) in the demand process. Autocorrelation in an input process can have a substantial effect on system performance and hence should not be ignored; see Livny, Melamed and Tsiolis (1993) for a well-known simulation study demonstrating the significant impact of interarrival and service-time autocorrelations on queueing system performance.

Input modeling for a univariate time series of order p refers to the selection of a probability distribution for the random variable X_t and the specification of the autocorrelation structure up to lag p; i.e., $\mathrm{Corr}[X_t, X_{t+l}] = (\mathrm{E}[X_t X_{t+l}] - \mu^2)/\sigma^2$ for $\ell = 1, 2, \ldots, p$, where μ and σ^2 are the mean and variance of X_t. An input model that is widely used for representing such an autocorrelated time series is the AutoRegressive Moving Average (ARMA) process. Specifically, the ARMA(p,q) process is represented by $X_t = \sum_{h=1}^{p} \alpha_h X_{t-h} + Y_t + \sum_{i=1}^{q} \beta_i Y_{t-i}$ for $t = p + 1, p + 2, \ldots$, where the α_h, $h = 1, 2, \ldots, p$ are fixed autoregressive coefficients and the β_i, $i = 1, 2, \ldots, q$ are fixed moving average coefficients that jointly determine the autocorrelation structure of the time series X_t, $t = 1, 2, \ldots$, while Y_t, $t = p + 1, p + 2, \ldots$ are independent and normally distributed random variables each of which has a mean of zero. A good resource for input modeling with ARMA is Box, Jenkins and Reinsel (1994). The major limitation of the linear ARMA model is the restriction of the marginal distribution of X_t to normal, limiting the use of the model for time series with arbitrary marginal distributions. Motivated by this drawback, there has been considerable research on modeling time series with marginals from non-normal families, such as exponential, gamma, geometric, or general discrete marginal distributions. For example, Lewis, McKenzie and Hugus (1989) relax the normal-distribution assumption of the ARMA model by constructing a time series with a gamma marginal distribution.

However, this model allows only limited control of the dependence structure, and a different model is required for each type of marginal distribution.

The Transform-Expand-Sample (TES) process introduced by Melamed (1991) differs from the previous time-series models by its ability to match an arbitrary marginal distribution and a lag-one correlation by applying the inverse-transformation method to a series of autocorrelated uniform random variables. However, the TES process does not guarantee the representation of the autocorrelation structure beyond the first lag, and extreme jumps may appear in its sample paths. In addition, the TES process is limited to the modeling of univariate time series; it cannot be extended to capture the joint distribution of correlated input random variables. The AutoRegressive-To-Anything (ARTA) process developed by Cario and Nelson (1996), on the other hand, allows the modeling of a time series with any marginal distribution and an autocorrelation structure specified up to lag p (≥ 1). Unlike TES, ARTA can represent autocorrelation structures beyond lag one with no extreme jumps in the sample paths. Furthermore, ARTA can be easily extended to model multivariate time series.

Specifically, the ARTA model builds on the autoregressive process of order p; i.e., the ARMA(p,q) process with $q = 0$, which serves as the base process with the following representation:

$$Z_t = \sum_{h=1}^{p} \alpha_p Z_{t-h} + Y_t, t = p+1, p+2, \ldots$$

The autocorrelation structure of the base process Z_t, $t = 1, 2, \ldots$ is uniquely determined by the autoregressive coefficients α_h, $h = 1, 2, \ldots, p$ and the variance of the random variables Y_t, $t = p+1, p+2, \ldots$ that ensure the standard normality of the base process. This allows one to obtain the input random variable X_t from the transformation $X_t = F^{-1}(\Phi(Z_t); \boldsymbol{\Psi})$, where $F^{-1}(\cdot;\boldsymbol{\Psi})$ is the inverse c.d.f. of the marginal distribution function $F(\cdot;\boldsymbol{\Psi})$, and Φ is the c.d.f. of the standard normal random variable. Cario and Nelson (1998) develop the ARTAFACTS software that specifies the autocorrelation structure of the base process Z_t, $t = 1, 2, \ldots$ to obtain the autocorrelation structure of the input process X_t, $t = 1, 2, \ldots$. Building on the input-modeling techniques described earlier, Biller and Nelson (2005) introduce an automated and statistically valid algorithm called ARTAFIT to fit an ARTA process with a marginal distribution from the Johnson translation system to historical data of limited length. It is reported that the ARTAFIT algorithm, which jointly estimates the marginal distribution and the autocorrelation structure of a time series, provides better fits than those obtained under the assumption of an independent input process; it also performs better than the separate estimation of the marginal distribution and the autocorrelation structure. The extension of this univariate time-series input model to multivariate time series is discussed later.

Multivariate Input Modeling

The objective of this section is to describe how to select a multivariate input model for a random vector composed of K correlated random variables. Examples of such a random vector include the demands of K different items in a retailer's product line, the processing times of a product at K different machines, and the financial defaults of K different suppliers in a supply-chain network. First, it is assumed that the K − dimensional random vectors $\mathbf{X}_t = (X_{1,t}, X_{2,t}, \ldots, X_{K,t})'$, $t \geq 1$ are independent over time to focus on capturing the dependence among the K component random variables $X_{k,t}$, $k = 1, 2, \ldots, K$. Then the temporal independence assumption is relaxed to consider multivariate time-series models that additionally account for the autocorrelations within the time series $\mathbf{X}_t$, $t \geq 1$.

Capturing the Joint Distribution of Correlated Inputs

Due to the analytical tractability in parameter estimation and the ease in random-vector generation, multivariate normal distribution has been a widely-used input model for correlated random variables. This model, however, assumes a normal marginal distribution for each input random variable, limiting its use for multivariate input modeling. This has motivated the multivariate extension of various standard distributions for input modeling; see Johnson, Kotz and Balakrishnan (1997) and Kotz, Balakrishnan and Johnson (2000) for a presentation of the resulting multivariate distributions.

The focus of recent multivariate input-modeling research has been on developing flexible input

models with arbitrary marginal distributions and positive definite correlation matrices. More specifically, the goal has been to construct a K-dimensional random vector $\mathbf{X} = (X_1, X_2, \ldots, X_K)'$ with marginal distribution functions $F_k(\cdot;\boldsymbol{\Psi}_k)$, $k = 1, 2, \ldots, K$ and input correlation matrix $\sum_{\mathbf{X}} \equiv [\rho(i, j); i, j = 1, 2, \ldots, K]$, where $\rho(i,j) = \mathrm{Corr}\left[X_i, X_j\right]$ for $i = 1, 2, \ldots, K$ and $j = 1, 2, \ldots, K$. The joint distribution function of the random vector $\mathbf{X}$ is given by

$$H(x_1, x_2, \ldots, x_k) = \Pr(X_1 \le x_1, X_2 \le x_2, \ldots, X_K \le x_K).$$

The application of the probability-integral transformation $F_k(\cdot;\boldsymbol{\Psi}_k)$ to the random variable X_k results in the uniform random variable U_k in (0, 1); i.e., $F_k(X_k;\boldsymbol{\Psi}_k) \equiv U_k$, and the standard normal random variable Z_k is obtained by applying the inverse c.d.f. Φ^{-1} of the standard normal random variable to U_k, i.e., $Z_k = \Phi^{-1}(F_k(X_k;\boldsymbol{\Psi}_k))$. Therefore, the joint distribution function H of the random vector $\mathbf{X}$ can be alternatively written as follows:

$$\begin{aligned}
&H(x_1, x_2, \ldots, x_K)\\
&= \Pr(F_k(X_k;\boldsymbol{\Psi}_k) \le F_k(x_k;\boldsymbol{\Psi}_k); k = 1, 2, \ldots, K)\\
&= \Pr(U_k \le u_k; k = 1, 2, \ldots, K)\\
&= \Pr\left(\Phi^{-1}(U_k) \le \Phi^{-1}(u_k); k = 1, 2, \ldots, K\right)\\
&= \Pr\left(Z_k \le \Phi^{-1}(F_k(x_k;\boldsymbol{\Psi}_k)); k = 1, 2, \ldots, K\right)\\
&= \Phi_{\Sigma_{\mathbf{Z}}}\left(\Phi^{-1}(F_1(x_1;\boldsymbol{\Psi}_1)), \Phi^{-1}(F_2(x_2;\boldsymbol{\Psi}_2)), \ldots, \Phi^{-1}(F_K(x_K;\boldsymbol{\Psi}_K))\right)
\end{aligned}$$

In this representation, $\Phi_{\Sigma_{\mathbf{Z}}}$ is the joint c.d.f. of the base random vector $\mathbf{Z} = (Z_1, Z_2, \ldots, Z_K)'$ with the correlation matrix $\Sigma_{\mathbf{Z}} \equiv [\rho_{\mathbf{Z}}(i,j); i, j = 1, 2, \ldots, K]$, where $\rho_Z(i,j) = \mathrm{Corr}\left[Z_i, Z_j\right]$ for $i = 1, 2, \ldots, K$ and $j = 1, 2, \ldots, K$. $\Phi_{\Sigma_{\mathbf{Z}}}$ allows the joint distribution function of $\mathbf{X} = (X_1, X_2, \ldots, X_K)$ to be written as a function of the marginal c.d.f.'s $F_k(X_k;\boldsymbol{\Psi}_k)$, $k = 1, 2, \ldots, K$. This function is known as the K-dimensional normal copula, which can be considered as a multivariate function that couples the arbitrary marginal c.d.f.'s $F_k(X_k;\boldsymbol{\Psi}_k)$, $k = 1, 2, \ldots, K$ with the correlation matrix Σ_Z to obtain the joint distribution function H. This joint distribution is also known as the Normal-To-Anything (NORTA) distribution in the stochastic input-modeling literature (Cario and Nelson 1997). Therefore, the dependence structure of the K – dimensional NORTA distribution is captured in a K − dimensional normal copula. Since the transformation $F_k^{-1}\left(\Phi^{-1}(Z_k);\boldsymbol{\Psi}_k\right)$ ensures that the input random variable X_k has the marginal distribution $F_k(\cdot;\boldsymbol{\Psi}_k)$, the main challenge in the construction of the NORTA distribution with the input correlation matrix $\Sigma_{\mathbf{X}}$ is to determine the base correlation matrix $\Sigma_{\mathbf{Z}}$ that matches the input correlation matrix $\Sigma_{\mathbf{X}}$. The identification of the base correlation matrix requires the solution of $K(K-1)/2$ individual correlation-matching problems of the form

$$\begin{aligned}
\rho_X(i,j) &= \mathrm{Corr}\left(F_i^{-1}(\Phi(Z_i);\boldsymbol{\Psi}_i), F_j^{-1}\left(\Phi(Z_j);\boldsymbol{\Psi}_j\right)\right)\\
&= \left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_i^{-1}(\Phi(Z_i);\boldsymbol{\Psi}_i) F_j^{-1}\left(\Phi(Z_j);\boldsymbol{\Psi}_j\right)\right.\\
&\qquad \left.\vartheta_{\rho_Z(i,j)}(z_i, z_j)\,dz_i\,dz_j - \mu_i\mu_j\right) \Big/ (\sigma_i\sigma_j)\\
&= c_{ij}(\rho_Z(i,j))
\end{aligned}$$

for $\rho_{\mathbf{Z}}(i, j)$, where $\vartheta_{\rho_{\mathbf{Z}}(i,j)}$ is the standard bivariate normal probability density function with correlation $\rho_{\mathbf{Z}}(i, j)$, and μ_i and σ_i^2 are the mean and variance of X_i. The function $c_{ij}(\rho_{\mathbf{Z}}(i, j))$ is nondecreasing, lies on the origin for $\rho_{\mathbf{Z}}(i, j) = 0$, and satisfies $|c_{ij}(\rho_{\mathbf{Z}}(i, j))| \le |\rho_{\mathbf{Z}}(i, j)|$ for $\rho_{\mathbf{Z}}(i, j) \in [-1, 1]$. Furthermore, the function $c_{ij}(\rho_{\mathbf{Z}}(i, j))$ is continuous under mild conditions on the marginal distribution functions $F_i(\cdot;\boldsymbol{\Psi}_i)$ and $F_j(\cdot;\boldsymbol{\Psi}_j)$. These properties of the function $c_{ij}(\rho_{\mathbf{Z}}(i, j))$ allow the simulation practitioner to perform a numerical search to find the base correlation $\rho_{\mathbf{Z}}(i, j)$ within a predetermined precision; see Cario and Nelson (1997) for further details on the function $c_{ij}(\rho_{\mathbf{Z}}(i, j))$ and the solution of the correlation-matching problem. However, the solution of the correlation-matching problems might lead to a base correlation matrix that is not positive definite. Therefore, there exist sets of marginal distributions with a feasible input correlation matrix that are not representable by the NORTA transformation. If, after solving the correlation-matching problems, the base correlation matrix is not positive definite, then the procedures introduced by Lurie and Goldberg (1998) and Ghosh and Henderson (2002) can be used for obtaining a symmetric, positive definite approximation of the base correlation matrix. Nevertheless, the failure of the transformation-based method is relatively rare in moderate dimensions of random vectors, and the method fails when the correlations are on the

boundary or within close proximity to the correlation values achievable for the specified marginals of the input process.

Next, consider the problem of estimating the unknown marginal distribution parameters $\boldsymbol{\Psi}_k$, $k = 1, 2, \ldots, K$ and the base correlation matrix Σ_Z from the historical data set $\{x_{k,t}; k = 1, 2, \ldots, K, t = 1, 2, \ldots, n\}$ of length n. The marginal-copula representation of the distribution function $H(x_1, x_2, \ldots, x_K)$ allows the joint density function to be written in terms of the normal copula density function ϕ_{Σ_Z} (i.e., the Kth-order derivative of the normal copula Φ_{Σ_Z} with respect to $F_k(X_k;\boldsymbol{\Psi}_k)$, $k = 1, 2, \ldots, K$) and the marginal density functions $f_k(x_k;\boldsymbol{\Psi}_k)$, $k = 1, 2, \ldots, K$:

$$\phi_{\Sigma_Z}(F_1(x_1;\boldsymbol{\Psi}_1), F_2(x_2;\boldsymbol{\Psi}_2), \ldots, F_K(x_K;\boldsymbol{\Psi}_K)) \times \prod_{k=1}^{K} f_k(x_k;\boldsymbol{\Psi}_k).$$

Therefore, the log-likelihood function of the historical data can be written as the sum of the K log-likelihood functions for the marginal distribution functions $F_k(\cdot;\boldsymbol{\Psi}_k)$, $k = 1, 2, \ldots, K$, i.e.,

$$L_k(\boldsymbol{\Psi}_k) = \sum_{t=1}^{n} \log f_k\left(x_{k,t};\boldsymbol{\Psi}_k\right), k = 1, 2, \ldots, K,$$

and the log-likelihood function for the copula density function; i.e.,

$$L_c(\Sigma_{\mathbf{Z}}) = \sum_{t=1}^{n} \log \phi_{\Sigma_{\mathbf{Z}}}\left(F_1\left(x_{1,t};\boldsymbol{\Psi}_1\right), F_2\left(x_{2,t};\boldsymbol{\Psi}_2\right) \ldots, F_K\left(x_{K,t};\boldsymbol{\Psi}_K\right)\right).$$

The structure of the log-likelihood function allows the use of a multi-stage estimation method known as the Inference For Margins (IFM). Specifically, the IFM method obtains the parameter estimate $\hat{\boldsymbol{\Psi}}_k$ by maximizing the log-likelihood function $L_k(\boldsymbol{\Psi}_k)$ for $k = 1, 2, \ldots, K$, followed by the maximization of the log-likelihood function $L_c(\Sigma_{\mathbf{Z}})$ to estimate the correlation matrix $\hat{\Sigma}_{\mathbf{Z}}$ using $\hat{\boldsymbol{\Psi}}_k$, $k = 1, 2, \ldots, K$ obtained in the first K stages. This procedure is computationally simpler than the maximum likelihood estimation which estimates all parameters $\boldsymbol{\Psi}_k$, $k = 1, 2, \ldots, K$, and $\Sigma_{\mathbf{Z}}$ from the full log-likelihood function $L_c(\Sigma_{\mathbf{Z}}) + \sum_{k=1}^{K} L_k(\boldsymbol{\Psi}_k)$. The IFM estimators are different from the maximum likelihood estimators unless the marginal distributions are normal. The IFM estimators are also less efficient than the maximum likelihood estimators, but they are strongly consistent and asymptotically normal under certain regularity conditions (Joe 1997).

The common feature of the multivariate input models presented in this section is the use of correlation as the measure of dependence. Correlation is also the most widely used dependence measure in multivariate input modeling. However, it is not the only dependence measure available for input modeling. A measure of dependence, which has been of particular interest in recent years, is tail dependence; i.e., the amount of dependence in the lower-quadrant tail or upper-quadrant tail of a bivariate distribution. The multivariate input models of this section can be easily extended to work for dependence structures with positive tail dependencies by simply replacing the normal copula with the appropriate multivariate copula; see Biller (2009) for the use of copula theory to extend the transformation-based methods to represent positive tail dependencies. An excellent review of the alternative dependence measures together with the copula theory is available in Joe (1997).

Autocorrelation

In settings with multiple sources of randomness, a correlation might exist not only between the input random variables of the system but also over time. An input model that can be used for representing a stationary multivariate time-series process is the Vector-AutoRegressive-To-Anything (VARTA) model (Biller and Nelson 2003). The measure of dependence assumed in this model is also correlation whose limitations are further inherited by VARTA. Biller (2009) shows that this multivariate time-series model fails to capture the dependencies in the tails of the joint distributions of its K component series, and generalizes VARTA to work for dependence structures with positive tail dependencies by using appropriate families of multivariate copulas.

Specifically, VARTA pulls together the theory behind the ARTA and NORTA input models and extends it to the K – dimensional time series

$$\left\{\mathbf{X}_t = \left(X_{1,t}, X_{2,t}, \ldots, X_{K,t}\right)'; t = 1, 2, \ldots\right\}$$

by taking the inverse transformation of the following vector autoregressive process of order p:

$$\mathbf{Z}_t = \sum_{h=1}^{p} \alpha_h \mathbf{Z}_{t-h} + \mathbf{Y}_t, t = p+1, p+2, \ldots.$$

In this representation, the α_h, $h = 1, 2, \ldots, p$ are fixed K-dimensional autoregressive coefficient matrices and $\{\mathbf{Y}_t; t = p + 1, p + 2, \ldots\}$ is a sequence of K-dimensional normal random vectors with zero mean and the covariance matrix that ensures that each component of $\mathbf{Z}_t = (Z_{1,t}, Z_{2,t}, \ldots, Z_{K,t})'$ is marginally standard normal. Notice that if $K = 1$, then the VARTA process reduces to an ARTA process; and if $K > 1$ and $p = 0$, the VARTA process corresponds to a NORTA vector. The major challenge in the construction of a VARTA process (and hence, for the ARTA and NORTA processes) is to specify the autocorrelation structure of the autoregressive base process $\{\mathbf{Z}_t; t = 1, 2, \ldots\}$ so that the input process $\{\mathbf{X}_t; t = 1, 2, \ldots\}$ exhibits a prespecified autocorrelation structure. Specifically, the base correlation $\rho_{\mathbf{Z}}(i,j,h) = \mathrm{Corr}\left(Z_{i,t}, Z_{j,t+h}\right)$ depends only on the input correlation $\rho_{\mathbf{X}}(i,j,h) = \mathrm{Corr}\left(X_{i,t}, X_{j,t+h}\right)$ for $i, j = 1, 2, \ldots, K$ and $h = 0, 1, 2, \ldots, p$, and the determination of the autocorrelation structure for the base process is equivalent to solving $pK^2 + K(K-1)/2$ individual correlation-matching problems; see Biller and Nelson (2003) for a discussion of solving correlation-matching problems in a multivariate time-series setting.

Concluding Remarks

Practical and theoretical issues for developing input models in both univariate and multivariate settings have been presented. Also considered was the situation where univariate input models fall short of capturing the correlations among different sources of randomness in the input environment of the stochastic system under study. Much research in the last decade has focused on multivariate input modeling to develop flexible input models in correlated settings. Most of the recent developments in input modeling is reported in the annual *Proceedings of the Winter Simulation Conference*, which are publicly available on the World Wide Web.

See

- ▶ Distribution Selection for Stochastic Modeling
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Time Series Analysis

References

Biller, B. (2009). Copula-based multivariate input models for stochastic simulation. *Operations Research, 57*(4), 878–892.

Biller, B., & Nelson, B. (2005). Fitting time-series input processes for simulation. *Operations Research, 53*(3), 549–559.

Biller, B., & Nelson, B. L. (2003). Modeling and generating multivariate time-series input processes using a vector autoregressive technique. *ACM Transactions on Modeling and Computer Simulation, 13*(3), 211–237.

Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (1994). *Time series analysis: Forecasting and control* (3rd ed.). Englewood Cliffs, NJ: Prentice Hall.

Cario, M. C., & Nelson, B. L. (1996). Autoregressive to anything: Time-series input processes for simulation. *Operations Research Letters, 19*(2), 51–58.

Cario, M. C., & Nelson, B. L. (1997). *Modeling and generating random vectors with arbitrary marginal distributions and correlation matrix*. Working paper, Department of Industrial Engineering and Management Science, Northwestern University, Evanston, IL.

Cario, M. C., & Nelson, B. L. (1998). Numerical methods for fitting and simulating autoregressive-to-anything processes. *INFORMS Journal on Computing, 10*, 72–81.

Ghosh, S., & Henderson, S. G. (2002). Chessboard distributions and random vectors with specified marginals and covariance matrix. *Operations Research, 50*(5), 820–834.

Joe, H. (1997). *Multivariate Models and Dependence Concepts*. London: Chapman and Hall.

Johnson, N. L., Kemp, A. W, & Kotz, S. (2005). *Discrete Univariate distributions* (3rd ed.). John Wiley & Sons, Inc., Hoboken, New Jersey.

Johnson, N. L., Kotz, S., & Balakrishnan, N. (1995). *Continuous Univariate distributions* (Vols. 1 and 2, 2nd ed.). John Wiley & Sons, Inc., United States of Amercia.

Johnson, N. L., Kotz, S., & Balakrishnan, N. (1997). *Discrete multivariate distributions*. John Wiley & Sons, Inc., United States of Amercia.

Kotz, S., Balakrishnan, N., & Johnson, N. L. (2000). *Continuous multivariate distributions*. Vol. 1: Models and applications (2nd ed.). John Wiley & Sons, Inc., United States of Amercia.

Kuhl, M. E., Ivy, J. S., Lada, E. K., Steiger, N. M., Wagner, M. A. F., & Wilson, J. R. (2010). Univariate input models for stochastic simulation. *Journal of Simulation, 4*, 81–97.

Law, A. M. (2007). *Simulation modelling and analysis* (4th ed.). McGraw-Hill.

Lewis, P. A. W., McKenzie, E., & Hugus, D. K. (1989). Gamma processes. *Stochastic Models, 5*(1), 1–30.

Livny, M., Melamed, B., & Tsiolis, A. K. (1993). The impact of autocorrelation on queueing systems. *Management Science, 39*(3), 322–339.

Lurie, P. M., & Goldberg, M. S. (1998). An approximate method for sampling correlated random variables from partially-specified distributions. *Management Science, 44*, 203–218.
Melamed, B. (1991). TES: A class of methods for generating autocorrelated uniform variates. *ORSA Journal on Computing, 3*, 317–329.
Pearson, K. (1895). Skew variations in homogeneous material. *Philosophical Transactions of the Royal Society, 186*, 343–414.
Ramberg, J., & Schmeiser, B. (1974). An approximate method for generating asymmetric random variables. *Communications of the ACM, 17*(2), 78–87.
Rohatgi, V. K., & Saleh, E. (2001). *An introduction to probability and statistics* (2nd ed.). John Wiley & Sons, Inc., United States of Amercia.
Schmeiser, B. W., & Deutsch, S. J. (1977). A versatile four parameter family of probability distributions suitable for simulation. *IIE Transactions, 9*, 176–181.
Swain, J. J. (2011). To boldly go . . .discrete event simulation software tools. *OR/MS Today, 38*(5), 58–71.
Vincent, S. (1998). Input data analysis. In J. Banks (Ed.), *Handbook of Simulation* (pp. 55–91). New York: Wiley.

Stochastic Model

A mathematical model in which some data and parameters are random variables.

See

- ▶ Deterministic Model
- ▶ Mathematical Model

Stochastic Optimization

Optimization in which the objective function and/or constraint functions are "noisy," i.e., involve random variables (e.g., expected values) that cannot be evaluated analytically and thus require estimation, such as through simulation of a stochastic system. Sometimes the term is also used to refer to deterministic optimization problems that introduce randomness in the search process, i.e., the resulting procedures are randomized algorithms for optimization.

See

- ▶ Simulation Optimization

Stochastic Process

A set of random variables indexed over a parameter set that is either discrete or continuous and often represents some concept of time.

See

- ▶ Inventory Modeling
- ▶ Markov Chains
- ▶ Markov Processes
- ▶ Point Stochastic Processes
- ▶ Queueing Theory
- ▶ Random Field
- ▶ Reliability
- ▶ Renewal Process
- ▶ Simulation of Stochastic Discrete-Event Systems

Stochastic Programming

Suvrajeet Sen
The University of Arizona, Tucson, AZ, USA
University of Southern California, Los Angeles, CA, USA

Introduction

Stochastic programming (SP) deals with a class of optimization models and algorithms in which some of the data may be subject to significant uncertainty. Such models are appropriate when data evolve over time, and decisions need to be made prior to observing the entire data stream. For instance, investment decisions in portfolio planning must be implemented before stock performance can be observed. Similarly, utilities must plan power generation before the demand for electricity is realized. Such inherent uncertainty is amplified by technological innovation and market forces.

As an example, consider electric power supply. Most states in the U.S. have adopted Renewable Portfolio Standards, which mandate far greater use of renewable resources in the future. However, renewable sources of energy (e.g., wind and solar) are intermittent

resources, because they are governed by highly variable forces of nature. Including such generators into a deterministic generation planning model would require accurate predictions of natural phenomena (i.e., wind and sunshine). Of course, requiring accurate predictions is tantamount to having a crystal ball into the future – an untenable assumption. SP provides a formal approach in which events of the future can be modeled using random variables (or stochastic processes) modeling the future. Because of the wide variety of applications where data uncertainty plays a critical role, this paradigm has also attracted researchers from a variety of academic domains, as well as government and industry. For instance, the area of supply chain management (e.g., Fisher et al. 1997) was an early adopter of SP. More recently, the 2011 report of the Defense Science Board has strongly recommended the use of SP for long-term trade studies carried out for military planning. Indeed, the volume by Wallace and Ziemba (2005) provides an entire collection of papers that are based on applications ranging from airline revenue management, homeland security, network planning and many more. Nevertheless, SP models remain some of the more challenging optimization problems.

While SP grew out of the need to incorporate uncertainty in linear and other optimization models (see Birge and Louveaux 1997), it also has close connections with other paradigms for decision-making under uncertainty. For instance, decision theory, dynamic programming and simulation-optimization, all share some common themes with SP. To explore these connections recall that one of the main strengths of decision theory is its emphasis on decision-maker's preferences, and ideas such as stochastic dominance emerged from the decision-theoretic segment of the OR/MS literature. It turns out that ideas such as stochastic dominance have made significant in-roads into the SP literature (Dentcheva and Ruszczyński 2003). Similarly, there are growing connections between dynamic programming (DP) and SP through new thrusts such as Approximate DP (Powell 2010). SP has a long tradition of creating approximations that provide asymptotic results and many SP methods provide bounds on deviations from optimality when terminated in finite time. The section on Computational Issues provides some examples. Finally, consider the connections between SP and simulation-optimization. The latter category supports models that are reasonably realistic in their description of the system, because they inherit the expressiveness of computer programming within the modeling framework. However, simulation-optimization models are relatively difficult to optimize because of the generality associated with non-convex noisy "Black Box" functions. Accordingly, simulation-optimization often attempts to combine statistical tools with global optimization. The connections between SP and simulation-optimization however are clearest in the context of sampling-based algorithms for SP. The latter draw upon several concepts from simulation-optimization (e.g., importance sampling, Latin hypercube sampling). The interplay between the different paradigms presents fascinating possibilities.

SP provides a general framework for modeling stochastic processes within constrained optimization models. Furthermore, it permits uncountably many states and actions, together with constraints, time lags, etc. One of the important distinctions that should be highlighted is that SP separates the model formulation activity from the solution algorithm. One advantage of this separation is that users need not be intimately familiar with SP algorithms in order to use them. This separation also promotes algorithmic developments that are based on specialized structures (e.g., linear, integer etc.), which may help algorithms to scale up by using special structures. On the downside of the ledger, SP formulations can lead to very large-scale problems, and methods based on approximation and decomposition become paramount. This article provides a road map for these methods and points to fruitful research directions along the way.

Mathematical Models and Properties

General Purpose SP Models

Consider a model in which the design/decision associated with a system is specified via vector x_1. Under uncertainty, the system operates in an environment in which there are uncontrollable parameters that are modeled using random variables. Consequently, the performance of such a system can also be viewed as a random variable. Accordingly, SP models provide a framework in which designs (x_1) can be chosen to optimize some measure of

performance (random variable). It is therefore natural to consider measures such as the worst-case performance, expectation and other moments of performance, or even the probability of attaining a predetermined performance goal. Furthermore, measures of performance must reflect the decision maker's attitudes towards risk. For example in the finance literature, it is common to model risk aversion through maximizing expected utility. However, alternative risk models have also become common in the SP literature. These include measures like conditional value at risk (Rockafellar and Uryasev 2000), semi-deviation (Ogryczak and Ruszczyński 2001), and others. The mathematical structure presented below begins with a traditional SP formulation, and a discussion of risk is revisited subsequently.

The following mathematical model represents a general SP formulation in which the design/decision variable x_1 is restricted to the set X_1 and $\tilde{\omega}_1$ denotes a multi-dimensional random variable:

$$\min_{x_1 \in X_1} \quad f_1(x_1) + E[h_2(x_1, \tilde{\omega}_1)] \tag{1a}$$

$$\text{s.t. } \Pr\{g_1(x_2, \tilde{\omega}_1) \geq 0\} \geq p_1 \tag{1b}$$

Here $E[]$ denotes the expectation with respect to $\tilde{\omega}_1$ and Pr[] is the probability of the event $g_1(x_1, \tilde{\omega}_1) \geq 0$.. The function g_1 is often modeled as a linear form and h_2 is the value function of another optimization problem stated as follows:

$$h_2(x_1, \omega_1) = \min_{x_2 \in X_2(x_1, \omega_1)} f_2(x_2; x_1, \omega_1)$$

In this notation, the subscripts are intended to designate the stage in which the relevant calculations are carried out. Thus f_1 reflects the initial cost in stage 1, whereas, h_2 reflects the cost-to-go in stage 2. In the SP literature, the function h_2 is used to reflect costs associated with adapting to information revealed through an outcome ω_1. For instance, in financial applications, this function may reflect the utility associated with costs of rebalancing the portfolio. Because the function $E[h_2]$ is associated with a recourse action x_2, it is referred to as the recourse function, although the use of the term value function is not uncommon either. Constraint (1b) is called a probabilistic (or chance) constraint. Such a constraint may be used to model system reliability. Note that formulation (1) is somewhat more general than one usually finds in the SP literature. Historically, the probabilistic constraint (1b) is treated separately from models using the recourse functions (1a). However, including both types of functions within one model allows a more cohesive statement of SP problems.

Multi-stage Stochastic Programming — While model (1) appears somewhat static, it is not difficult to glean a dynamic element in the formulation: note that the function h_2 is realized only after the design x_1 is in place. This sequential nature is an essential element of decision making under uncertainty. Indeed, if h_2 is defined recursively, problem (1) may be looked upon as the first-stage problem of a more extensive multi-stage formulation. To present this generalization of (1), consider an N-stage problem. Let the boundary conditions be given by $h_{N+1} \equiv 0$, and let ω_0 denote a degenerate random variable reflecting the deterministic information available prior to decisions of stage 1. For $t = 1,\ldots, N$, let ξ^t denote the history prior to stage t [i.e., $\xi^t = (\omega_0, \ldots, \omega_{(t-1)})$]. Note that the decision variables in stage t depend on the history of the data process. Hence these variables are functions of random variables, and will be denoted $x_t(\xi^t)$. The entire history of decisions until stage t will then be represented as a superscripted vector $x^t(\xi^t) = (x_1(\xi^1), x_2(\xi^2), \ldots, x_t(\xi^t))$, or simply x^t. For $t = 2,\ldots, N$, define the value functions

$$\begin{aligned} h_t(x^{t-1}, \xi^t) = \min_{x_t \in X_t(x^{t-1}, \xi^t)} & f_t(x_t; x^{t-1}, \xi^t) \\ & + E\left[h_{t+1}\left(x^t, \tilde{\xi}^{t+1} | \xi^t\right)\right] \\ \text{s.t } \Pr\{g_t(x_t, \xi^{t+1} | \xi^t) \geq 0\} & \geq p_1 \end{aligned} \tag{2}$$

where $h_{N+1} \equiv 0$, and E and Pr[] denote the conditional expectation and the conditional probability (respectively) associated with the appropriate evolutionary state of the random variables. Using these recursively defined functions in (1) yields a multi-stage SP formulation.

While (1) and (2) present a DP-type recursion to state the SP problem, it is important to note that all random variables are path dependent, and furthermore,

the statement of the model does not constitute the algorithm. In fact, alternative statements of the multi-stage problem are also possible. Consider a formulation in which the decisions are allowed to depend on the entire realization ξ^N. Let $\mathrm{x}^N\left(\tilde{\xi}^N\right)$ denote a sequence of random vectors $\left(x_1\left(\tilde{\xi}^N\right), x_2\left(\tilde{\xi}^N\right), \ldots, x_N\left(\tilde{\xi}^N\right)\right)$. It is important to note that such a policy cannot be implemented since decisions in stage t require the knowledge of the entire realization! Hence, the plans $x^N\left(\tilde{\xi}^N\right)$ cannot be feasible, unless the decisions are such that x_t depends only on data available until stage $t-1$. As shown below, such information constraints can be incorporated explicitly.

Let $\omega^t = (\omega_t, \ldots, \omega_N)$. Since any outcome $\xi^N = (\xi^t, \omega^t)$ for any t, the decisions in stage t can be represented as a random vector denoted $x_t(\tilde{\omega}^t|\xi^t)$. Then the information constraints (also called the nonanticipativity constraints) may be stated as $x_t(\tilde{\omega}^t|\xi^t) - E[x_t(\tilde{\omega}^t|\xi^t)] = 0$, almost surely.

Since an objective function that is non-separable by stage can be written as $\mathrm{E}\left[\mathrm{f}\left(\mathrm{x}^{\mathrm{N}}\left(\tilde{\xi}^{\mathrm{N}}\right), \tilde{\xi}^{\mathrm{N}}\right)\right]$, the inclusion of information (nonanticipativity) constraints provides a legitimate multi-stage model which does not appeal to either separability or recursion. However, multi-stage stochastic programming algorithms are far less advanced than two-stage models.

The formulations presented above impose very few restrictions. Perhaps the most important restriction imposed in an SP formulation arises from the assumption that randomness is exogenous and cannot be affected by decisions. In certain design problems, such an assumption may not be valid, and in these cases, the models outlined above may not be adequate. However, in cases where the impact of decisions on the probability distribution can be reflected via a binary switching variable, then one may be able to introduce binary variables to represent the evolution of the distribution based on prior decisions. Naturally, such a model will lead to a Stochastic Mixed-Integer Program (SMIP) for which there has been considerable interest since 2000. Nevertheless, note that there is a large class of applications where randomness is exogenous (e.g., weather, loads, prices of financial instruments, market demands, etc.), and SP models provide an attractive approach, especially when faced with a continuum of choices in the presence of constraints.

The main challenge in designing algorithms for SP problems arises from the need to calculate conditional expectation and/or conditional probability associated with multi-dimensional random variables. For all but the smallest of problems, one resorts to approximations. The study of SP algorithms has therefore led to alternative ways of approximating problems, some of which satisfy certain asymptotic properties. This reliance on approximations has prompted researchers to study conditions for the convergence of approximations, and/or the convergence of solutions of approximate problems (to a solution of the original). Of course, conditions ensuring the former imply the latter, but the converse does not hold. Issues related to convergence of approximations can be addressed through the theory of epi-convergence, whereas issues pertaining to convergence of solutions of approximations (to a solution of the original) can be addressed through the notion of epigraphical nesting (see Rockafellar and Wets 1998).

Properties of Specific SP Models

Computational challenges associated with SP problems vary a great deal with the class of problems being addressed. As with any large-scale optimization problem, exploiting properties and the structure of problems provides the key to effective algorithms. This subsection presents properties associated with some important classes of SP problems, and related computational issues will be presented in the following section.

Some Properties of Stochastic Linear Programs with Recourse — For this class of problems, all functions and constraints are defined by linear/affine functions, and the probabilistic constraints are absent. This remains one of the more widely studied models, and most of the applications reported in the literature belong to this category (including the applications mentioned earlier). Problems of this type can be shown to be convex optimization problems, and the full power of convex analysis can be brought to bear on such problems. Notwithstanding such mathematical attractiveness, SLP problems lack one of the more desirable numerical properties, namely, smoothness.

Only under very special circumstances (absolute continuity of random variables) can one expect (1a) to be differentiable. Accordingly, many of the more successful algorithms for these problems draw upon non-differentiable optimization methods such as Regularized Decomposition (Ruszczyński 1986), as well as Regularized SD (Higle and Sen 1994), both of which are closely tied to bundle-trust algorithms.

Some Properties of Stochastic Mixed Integer Programs — As in the previous paragraph, suppose that probabilistic constraints are set aside. In a stochastic mixed integer linear program (SMIP), if only the first-stage decisions include integer restrictions, then the remaining problem inherits the properties of a SLP. This class of problems (with first-stage integer variables) is similar to the problems originally envisioned by Benders (1962). In general, though (i.e., when integer variables appear in future stages), SMIP is much more challenging. For such problems, convexity of the objective function is far too much structure to expect. Indeed, the objective function (1a) can be discontinuous. However, by assuming that any setting of decision variables yields a finite objective value (i.e., complete recourse), and assuming a weak covariance condition (Schultz 1993), the objective function can be shown to be lower semicontinuous. As with continuous SP algorithms, scalability is a key requirement, and decomposition-coordination methods remain the basis for effective algorithms.

Some Properties of Probabilistically-Constrained Problems — These models are widely used to reflect grade of service constraints (e.g., Medova 1998). One of the simpler probabilistically-constrained problems arises in cases where the function g_1 used in (1b) assumes values in $\Re$ and is separable (i.e., $g_1(x_1, \omega_1) = \theta_1(x_1) + \omega_1$). In this case a deterministic constraint requiring $\theta_1(x_1)$ to exceed the quantile (level p_1) is equivalent to the chance constraint. There are a few other cases that are easily handled. Prékopa (1971) showed that a much larger class of random variables yield the convexity property; he showed that if the function g (see (1b)) is linear/affine in x and randomness only appears additively, and the random variable has a log-concave probability density function, then the resulting feasible region is convex. However, for discrete random variables this is no longer true, and in this case, the set of feasible solutions can be represented as a disjunctive set. Algorithmic work on probabilistically constrained models also allows MIP models to be extended with probabilistic constraints (see following section).

Risk Modeling in Stochastic Programming — Since 2000, the SP approach has grown enormously in its ability to incorporate risk. One approach, namely the notion of Conditional Value at Risk (CVaR), is discussed next. Recall from (1) that both the objective function and constraints are defined in terms of the first-stage decision x_1, as well as random variables $\tilde{\omega}_1$. For the sake of brevity, the sub/super scripts are dropped and consider any one of the functions, $g(x, \tilde{\omega})$, say. In order to model risk, one typically wishes mitigate the negative impacts of variability (referred to as risk). The real-valued function g may be looked upon as a loss (random variable), and let $\psi(x, \zeta) = \Pr[g(x, \tilde{\omega}) \leq \zeta]$. Given $p \in (0,1)$, define $\zeta_p(x) := \min\{\zeta | \psi(x, \zeta) \geq p\}$. This quantity is known as Value at Risk (VaR_p), a term that is popular in the financial world. Thus, for a given $p \in (0,1)$, VaR_p ensures that a decision x satisfies (1b). However, this constraint does not measure the consequences of outcomes beyond $\zeta_p(x)$. Thus when tail losses are very high, VaR_p is unable to distinguish between decisions that are more risky at the same confidence level p. In addition, as indicated in the subsection on Probabilistic Constraints, these models can be mathematically difficult due to non-convex feasible sets. Due to these difficulties, Rockafellar and Uryasev (2000) suggested an alternative to VaR_p, known as CVaR_p, or Conditional Value at Risk. CVaR_p is defined as the conditional mean of the loss random variable in the tail to the right of $\zeta_p(x)$ in case of continuous random variables. In other words, if $\psi_p(x,\zeta)$ denotes the distribution of the random variable $\mathrm{Max}\{g(x, \tilde{\omega}) - \zeta_p(x), 0\}$, then the expectation of this random variable gives the mean *excess* loss. For cases in which $g(x, \tilde{\omega})$ is a discrete random variable, the definition of CVaR_p can be shown to be a convex combination of VaR_p and CVaR_p^+, which is the conditional mean (of the loss random variable) strictly to the right of $\zeta_p(x)$. One important feature of CVaR_p is its convexity, and is therefore computationally tractability too. For further attractive properties of CVaR_p and other coherent measures of risk, see Rockafellar (2007).

Robust Optimization — The term robust optimization has been used for several different

classes of models, all of which share the goal of providing decisions that are feasible and reasonably good for an entire set of model parameters, rather than one specific instance of parameters. This approach to decision-making is different from other stochastic optimization approaches in at least two substantive ways: (a) the goal of robust optimization is to immunize decisions in case of imperfect information (or data), and (b) the decisions themselves do not evolve as better information becomes available. Such models are useful in applications such as engineering design. For example, in designing a truss, an engineer assumes nominal values for properties of each member of the truss. However, due to manufacturing and material variability, some of the properties may vary from those nominal values. Of course, the design should be such that the truss is able to perform so long as all members satisfy the required properties within some acceptable tolerance. Moreover, in such applications, the design (i.e., the decision) cannot change with the specific realization of parameters, although the objective as well as constraint values (i.e., performance) will vary. This is reflected in items (a) and (b) above.

Robust optimization models, first proposed by Soyster (1973), are distribution-free formulations in which an uncertainty set replaces the notion of a probability space. For the sake of definiteness, consider the approach of Bertsimas and Sim (2004). In their paper the authors start with a nominal deterministic LP, and augment it by incorporating certain protection functions that represent extreme values of constraint coefficients which are allowed to belong to given intervals of uncertainty. The cross product of these intervals forms the uncertainty set. By using LP duality, these protection functions can be shown to have a very simple structure, and the resulting Robust LP is simply another larger LP whose size is only modestly larger than the nominal LP. Similar results have also been obtained for IP formulations (Bertsimas and Sim 2003), and they show that when uncertainty only affects cost coefficients, and the nominal problem is polynomially solvable (e.g., shortest path), then the Robust IP also inherits polynomially solvability. Other approaches to robust optimization, (e.g., using ellipsoidal uncertainty sets leading to conic quadratic programs) have been proposed by Ben-Tal and Nemirovski (1998).

Computational Issues

The main computational challenges for stochastic programming may be attributed to the fact that uncertainty must be quantified every step of the way: input data and model development, algorithmic methods for optimization, and finally, output analysis. It has been suggested that such effort (especially knowledge of distributions), may be far too demanding, thus leading to slower adoption of SP methodology within the modeling and optimization community. Advances in statistical computing, machine learning etc. are making such activities (e.g., estimating/approximating distributions) much less onerous than in the past. Since many tools in OR/MS, e.g., stochastic discrete-event simulation and decision analysis, are deeply steeped in the use of probabilistic knowledge, the need to provide probabilistic description should not be a bottleneck for SP. The challenge is to provide an end-to-end software environment in which decisions under uncertainty can be modeled, processed and analyzed in a manner that distills uncertainty down to statistically quantified reports to support decision-making. In the remainder of this section current approaches towards statistical quantification for SP are summarized, but not before the more traditional deterministic decomposition algorithms.

Deterministic Decomposition Algorithms for SP

Deterministic Algorithms for Stochastic Linear Programs (SLP) — The design of algorithms for SLP is intimately tied to the universe of outcomes reflected in the model. For instances in which the future is encapsulated using only a few outcomes/scenarios, the deterministic equivalent formulation (DEF) can be solved using deterministic decomposition methods that are extensions of Benders' decomposition (e.g., Regularized Decomposition of Ruszczyński 1986). Others (e.g., Linderoth and Wright 2003) are more direct extensions of bundle-trust methods of non-smooth optimization (e.g., Kiwiel 1990). Another class of deterministic decomposition algorithms is based on relaxing the information/nonanticipativity constraints (Rockafellar and Wets 1991). This approach is particularly promising for parallelizing algorithms for multi-stage problems. An overview of all of these methods can be found in the text by Birge and Louveaux (1997). In any event, since

the entire collection of DEF formulations use deterministic methods, the need for statistical quantification of outputs does not apply to them. However, in cases where an instance is created by sampling the original problem, statistical analysis of output is essential.

Algorithms for Stochastic MIPs — One of the major thrusts in the SMIP literature calls for algorithms that go beyond Benders' decomposition so that one can address models in which the second-stage (recourse) decision variables are also restricted to be integer. In cases where the first stage has binary variables, and the second-stage has general mixed-integer variables, Laporte and Louveaux (1993) have proposed an extension of the L-shaped method. Unfortunately, it requires that the second stage problem (possibly a mixed-integer linear program (MIP)) is solved to optimality for all scenarios. It is not difficult to see that solving many MIPs in each iteration can become a major bottleneck. To overcome this, several algorithms have been proposed so that each iteration only requires the solution of an LP relaxation of the scenario problems. In order to achieve asymptotic convergence of such schemes, it becomes necessary to generate cutting planes in a sequential manner so that the approximations become stronger as the algorithm proceeds, and obtains an optimal solution (or a near-optimal solution) in the limit. Since 2000, several algorithms of this genre have been proposed. Some of these are based on parametric Disjunctive Programming where cuts depend parametrically on binary first-stage decisions (e.g., Sen and Higle 2005; Sen and Sherali 2006; Sherali and Fraticelli 2003). A similar approach based on parametric Gomory cuts is presented in Gade et al. (2012) Computational results for parametric disjunctive cuts are given in Yuan and Sen (2009). For a more complete exposition of these algorithms and computations, see Sen (2010).

Successive Approximation Algorithms for SP

Unlike the deterministic decomposition methods mentioned above, there are many realistic instances for which the sample space is so large that enumerating every outcome may be impossible. In such cases, one resorts to successive approximation methods. In the following the focus is on alternative ways to construct approximations.

Bound-based Approximations for SLP Problems — For two-stage models, there are essentially two major approaches to generating approximations. One is based on aggregating data points, and another based on selecting data points. Algorithms in the former class of lead to successive approximation methods in which finer discretizations of the sample space are created based on the solution of an aggregated stochastic program. These successive approximation schemes (e.g., Frauendorfer 1992; Edirisinghe and Ziemba 1996) are able to provide bounds on the optimality gap, thus providing the decision-maker some guarantees. Similar bounds on the gap are also possible for multi-stage models as suggested in Casey and Sen (2005). Unfortunately the scalability of these methods has remained unresolved, and the literature has moved steadily towards sample-based methods, which are presented next.

Sampling-based Approximations for SLP Problems — Sampling in SP includes both Monte-Carlo as well as quasi-Monte Carlo methods. The latter draws upon the numerical integration literature, and its application to SP appears in Pennanen and Koivu (2005). However, one of the main distinctions to bear in mind for SP is that the expectation (integrand) depends on the decisions. For the most part, SP algorithms tend to generate a sample prior to optimization, so that the approximation does not typically adapt to the decision. This is also true for most Monte-Carlo approaches such as Sample Average Approximation (Shapiro and Homem-de-Mello 1998). The separation between generation of the sample, and the application of an optimization algorithm tends to preclude inexact optimization, which has been a very powerful concept in many areas of optimization (e.g., inexact Newton's method). The concept of inexact optimization between samples is highlighted in the Stochastic Decomposition (SD) algorithm for two-stage SLP (Higle and Sen 1996). Such inexact optimization is also adopted in a robust stochastic approximation (RSA) algorithm (Nemirovski et al. 2009), where the adaptation of stochastic approximation generates excellent solutions with a fraction of the computational effort required by SAA. In defense of SAA, however, one is able to obtain relatively low variance lower bound estimates at the end of an experiment with a few runs using a relatively large sample approximation (Linderoth et al. 2006).

Such lower bound estimates are difficult to obtain for RSA, and further post-processing becomes necessary. Moreover, RSA computations presented in Nemirovski et al. (2009) do not include computational times for post-processing. It is interesting to note that SD is, in some ways, a compromise between RSA and SAA: as with RSA, SD works with inexact subgradients that are observed dynamically as the algorithm proceeds, and as with SAA, it provides a capability for estimating sampled lower bound as well as its variability. In addition, both SAA and SD are also capable of providing reasonable estimates of the first-stage dual multipliers, which may be useful in certain pricing applications of SP (Higle 2007).

Sampling-based Approximations in SMIP Problems — For cases where the integer variables only appear in the first stage, algorithms based on Benders' decomposition are generally sufficient, although for cases in which a large number of scenarios arise, one may have to combine ideas from Benders' decomposition with those from sampling as discussed above. Such a method has been presented in Norkin et al. (1996). Sampling has also been part of the motivation for MIP with chance constraints, where Leudtke and Ahmed (2008) provide performance guarantees by setting up a convex approximation by sampling. In related work on jointly chance constrained MIPs, Kucukyavuz (2012) presents strong valid inequalities by using mixing sets with cardinality constraints (for the sampled case), as well as a knapsack constraint for the more general case. Compact extended formulations are also suggested for obtaining strong relaxations for solving chance constrained MIPs.

Multi-stage SP: Scenario Generation and Sampling — Two early approaches to scenario tree generation were Hoyland and Wallace (2001) and Pflug (2001). The former is designed to match certain known moments of a stochastic process using optimization, where as the latter generates a scenario tree using a simulation model, coupled with a stochastic approximation algorithm. Other approaches combining Monte Carlo sampling and clustering have also been proposed in Gulpinar et al. (2004). These methods are independent of the solution algorithm used to solve the multi-stage SP, and it is likely that the scenario tree generated by any one exceeds the capability of most solution algorithms. In order to allow users to formulate a smaller SP model, Dupacova et al. (2003) (see also Heitsch and Romisch 2007) present a scenario reduction scheme based on the nearest discretization for a prescribed number of scenarios. This approach, which also incorporates some heuristics to enhance scalability, is now available through GAMS.

As for combining both sampling in solution algorithms for multi-stage SLP, the idea of stochastic dual decomposition, originating with Pereira and Pinto (1991), has continued to attract attention. Asymptotic results are provided by Philpott and Guan (2008). In Shapiro (2011), the author suggests that in many previously reported papers, the quality of solutions reported for multi-stage models were unclear. In order to give the reader a sense of what the future of multi-stage SP may hold, the special case of statistical quantification for two-stage SLP is discussed below.

Statistical Quantification of SP Output: Two-Stage Models

Sampling methods for two-stage SLPs (i.e., SAA as well as SD) have been tested on several test instances of varying sizes. Unlike deterministic optimization where the size of an optimization model is unambiguous (because the size of the input is well defined), this is not necessarily the case for SP. To appreciate this, note that instances that are defined by continuous random variables should be looked upon as infinite-dimensional problems, although the first-stage (here-and-now) decisions may be finite dimensional. As a result, an exact representation of an SP instance may be difficult, and in fact, an exact solution as well as the exact optimal value may also be elusive. For these reasons, it is important to report the degree of accuracy of both the objective function estimate, as well as any lower bound estimate that may provide a sense of the quality of one or more feasible solutions. Simply reporting the objective function value of an approximation (using a few scenarios) without providing an estimate of errors can be misleading!

Bayraksan and Morton (2009) discuss a framework which highlights the need for reporting variability of objective value estimates by replicating the optimization process using independent samples. The estimated sample mean obtained after multiple runs then provides an estimated lower bound on the optimal value of the original (unsampled) model (Mak et al. 1999). It is not difficult to see that the

Stochastic Programming, Table 1 SP test instances

Problem name	No. of first stage variables	No. of second stage variables	No. of random variables	Universe of scenarios
LandS	4	12	3	$O(10^6)$
20TERM	63	764	40	$O(10^{12})$
SSN	89	706	86	$O(10^{70})$
STORM	121	1,259	117	$O(10^{81})$

estimated sample mean increases with sample size and asymptotically approaches the optimal value of the SP problem. As for estimated upper bounds, the obvious strategy is to fix a first-stage decision, and use i.i.d. sampling to estimate an upper bound. However, the reader should bear in mind that a low variance upper bound estimate is often computationally as demanding as solving an approximation that yields a lower bound. Striking a reasonable balance between these is important for overall efficiency.

Table 1 presents some characteristics of a few test instances that have been used for computational experiments in the literature. These instances are listed in increasing order of random variables, and are useful in illustrating statistically quantifiable bounds for SP problems whose sample space may contain so many outcomes that they are best described as being infinite dimensional. Again, if one uses sampling to provide approximate solutions for any of these instances, it is important to report errors with respect to the original SP instance.

The SAA estimates of Table 2 appear in Linderoth et al. (2006), and these are compared with computational results from Regularized SD (Higle and Sen 1994). In addition to the four instances above, Linderoth et al. (2006) also include a test instance named "gbd". This instance has not been included here because it is a simple-recourse model with independent random variables; for such instances, bounds-based approximation (see the previous subsection) provides a more accurate and scalable approach.

For the SAA experiment of Table 2, Linderoth et al. (2006) report solving SAA experiments with several different sample sizes. In the interest of brevity, as well as best accuracy in estimates, their results are summarized for a sample size of 5,000 using Latin hypercube sampling. In the SAA-(Average Values) column of Table 2, the entries in rows OBJ-LB for each instance correspond to the average value from solving 7-10 SAA replications in which each replication contained 5,000 sampled scenarios. The authors used a grid-enabled bundle-trust algorithm presented in Linderoth and Wright (2003) to solve each replication. The computational grid was managed by Condor, and consisted of hundreds of Linux PCs at several locations around the U.S. However, the runs apparently used only 100 PCs at any given time. One might recall that Pentium IV processors, with average clock speeds of 2.0–2.4 Ghz were average PC processors in 2005. In any event, the wall clock time for an instance like SSN was reported to be about 30-45 min per SAA instance, suggesting a total wall clock time of about 3.5 h for seven replications (@30 min/replication) and 7.5 h for ten replications (@ 45 min/replication). Once the solutions to these 7–10 instances were obtained, they were each evaluated to some degree of accuracy using a sample size of 20,000 for each SAA solution. From this preliminary estimate, the authors chose the solution with the lowest (preliminary) estimate of the objective value, and sampled further 50 batches, with each batch consisting of 20,000 outcomes. The OBJ-UB estimates reported in the SAA-(Average Values) column is the estimate obtained from this upper bounding exercise. It is important to note that with each upper bounding entry, one associates exactly one primal solution – a solution that is recommended for decision-making.

In Table 2, the set of columns adjacent to the SAA columns is data from the SD experiment. For the SD-(Average Values) column, the OBJ-LB entries correspond to the average value from solving 20 SD replications of the Regularized SD algorithm (Higle and Sen 1994). The reader might observe that there is no sample size reported for SD, because it samples until a non-parametric stopping rule (based on bootstrapping) terminates a replication of the algorithm. In any event, these calculations were carried out on a laptop-grade platform: MacBook Air with 1.8 GHz Intel Core i5 processor, 4 GB of 1,600 MHz DDR3 Memory.

The last column of Table 2 summarizes the differences in average values obtained from these two experiments. For the most part, the differences in average bounds are less than a small fraction of 1%. The only bound whose difference (with SAA) is close to 1% is the lower bound for SSN. Considering that the

Stochastic Programming, Table 2 Statistical quantification with SAA and SD

Instance name	Upper (UB) and lower bounds (LB)	SAA estimates using a computational grid		SD estimates using a laptop		Percentage difference in Average values
		Average values	95% CI's	Average values	95% CI's	
LandS	OBJ-UB	225.624	±0.005	225.54	±0.64	0.037
	OBJ-LB	225.62	±0.02	225.24	±0.64	0.168
20TERM	OBJ-UB	254311.55	±5.56	254476.87	±1005.86	0.065
	OBJ-LB	254298.57	±38.74	253905.44	±162.49	0.154
SSN	OBJ-UB	9.913	±0.022	9.91	±0.05	0.03
	OBJ-LB	9.84	±0.10	9.76	±0.16	0.813
STORM	OBJ-UB	15498739.41	±19.11	15498624.37	±48176.76	0.0007
	OBJ-LB	15498657.8	±73.9	15496619.98	±4615.85	0.013

upper bound for SSN is only 0.03% different from the average reported in Linderoth et al. (2006), it is fair to suggest that the average bounds produced by SAA with 5,000 samples and SD are comparable. However it is difficult to make precise comparisons of computational effort between algorithms implemented and tested using software and hardware from different eras. To give the reader a sense of the computational time for SD, note that all 20 SD replications for SSN were completed within 50 min. Suffice to say that using SD on computing platforms of today provides a widely accessible approach for realistic stochastic programming models (e.g., SSN – see Sen et al. 1994).

Software for SP

Since 2000, there have been significant advances in the development of modeling environments for SP. Watson, Woodruff, and Hart (2010) have developed a Python-based modeling environment for Stochastic Programming (PySP). In addition to its modeling capabilities, PySP also provides an implementation of the scenario aggregation (progressive hedging) algorithm for SP. Other notable developments include the implementation of the scenario reduction framework within GAMS. This has the potential to move multi-stage SP into a higher level of usage, and perhaps, there will be reports of accuracy of multi-stage models in the same manner that Table 2 presents confidence intervals for both upper as well as lower bounds. In 2012, Frontline Systems released their platform (Risk Solver Platform) for Robust Optimization as well as two-stage Stochastic Decomposition. Similarly, a product named Portfolio Safeguard (by AORDA) uses SP for portfolio optimization. Two-stage SLP is at the cusp of breaking into a new segment of the modeling world that brings together risk analysis and optimization. In other words, the time for two-stage SLP as a practical modeling tool has arrived.

Concluding Remarks

Several prevailing trends in SP are expected to continue. For instance, it is expected that there will be continued growth in modeling risk; greater emphasis on multi-stage models and methods; greater focus on scalable (e.g., decomposition) methods for SMIP models; greater visibility for strategic models under uncertainty (e.g., stochastic games and stochastic variational inequalities). Nevertheless, it is also important to recognize some important challenges: (a) As pointed out in the introduction, there are several areas of OR/MS that address decision-making under uncertainty. In this sense, there is a relatively large community of researchers from DP, Decision Theory, and Simulation-Optimization with significant overlaps with SP. It is a challenge to put these areas on a common unified footing. Such a development would allow greater cross-fertilization between the areas. (b) Most approaches to multi-stage SP (see (2)) are unable to address models driven by continuous stochastic processes. A statistically quantifiable approach to such models would be very satisfying; however, such a capability is not available yet. By emphasizing statistical quantification for similar two-stage models, it is hoped that this article provided a sense of the types of outputs that are necessary for multi-stage models. Finally, SP computations are not difficult to perform on today's hardware environment. What is holding SP

back is the lack of end-to-end software support. Given the new breed of SP software that is becoming popular, it is expected that there will be a full-fledged SP environment before too long.

See

- ▶ Approximate Dynamic Programming
- ▶ Benders Decomposition Method
- ▶ Chance-Constrained Programming
- ▶ Decision Analysis
- ▶ Dynamic Programming
- ▶ Linear Programming
- ▶ Portfolio Theory: Mean-Variance Model
- ▶ Risk Assessment
- ▶ Sample Average Approximation
- ▶ Simulation Optimization
- ▶ Variance Reduction Techniques in Monte Carlo Methods

References

Bayraksan, G., & Morton, D. P. (2009). Assessing solution quality in stochastic programming via sampling. *Tutorials in Operations Research, 5*, 102–122.

Benders, J. F. (1962). Partitioning procedures for solving mixed variables programming problems. *Numerische Mathematik, 4*, 238–252.

Ben-Tal, A., & Nemirovski, A. (1998). Robust convex optimization. *Mathematics of Operations Research, 23*, 769–805.

Bertsimas, D., & Sim, M. (2003). Robust discrete optimization and network flows. *Mathematical Programming, 98*, 49–71.

Bertsimas, D., & Sim, M. (2004). The price of robustness. *Operations Research, 52*, 35–53.

Birge, J. R., & Louveaux, F. V. (1997). *Introduction to stochastic programming*. New York: Springer.

Casey, M., & Sen, S. (2005). The scenario generation algorithm for multi-stage stochastic linear programming. *Mathematics of Operations Research, 30*, 615–631.

Defense Science Board Report. (2011). *Enhancing adaptability of U.S. military forces* (p. 31).

Dentcheva, D., & Ruszczyński, A. (2003). Optimization with stochastic dominance constraints. *SIAM Journal on Optimization, 14*, 548–566.

Dupacova, J., Growe-Kuska, N., & Romisch, W. (2003). Scenario reduction in stochastic programming: An approach using probability metrics. *Mathematical Programming, 95*, 493–511.

Edirisinghe, N. C. P., & Ziemba, W. T. (1996). Implementing bounds-based approximations in convex-concave two stage programming. *Mathematical Programming, 19*, 314–340.

Fisher, M., Hammond, J., Obermeyer, W., & Raman, A. (1997). Configuring a supply chain to reduce the cost of demand uncertainty. *Production and Operations Management, 6*, 211–225.

Frauendorfer, K. (1992). *Stochastic two-stage programming* (Lecture notes in economics and mathematical systems, Vol. 392). Berlin: Springer.

Gade, D., Kucukyavuz, S., & Sen, S. (2012, to appear). Decomposition algorithms with parametric Gomory cuts for two-stage stochastic integer programs. *Mathematical Programming*.

Gulpinar, N., Berc, R., & Settergren, R. (2004). Simulation and optimization approaches to scenario tree generation. *Journal of Economic Dynamics and Control, 28*, 1291–1315.

Heitsch, H., & Romisch, W. (2007). Scenario tree modeling for multistage stochastic programs. *Mathematical Programming, 118*, 371–406.

Higle, J. L. (2007). Bid-price control with origin-destination demand: A stochastic programming approach. *Journal of Revenue and Pricing Management, 5*, 291–304.

Higle, J. L., & Sen, S. (1994). Finite master programs in stochastic decomposition. *Mathematical Programming, 67*, 143–168.

Higle, J. L., & Sen, S. (1996). *Stochastic decomposition: A statistical method for large scale stochastic linear programming*. Dordrecht: Kluwer.

Higle, J. L., & Sen, S. (1999). Statistical approximations for stochastic linear programming problems. *Annals of Operations Research, 85*, 173–192.

Hoyland, K., & Wallace, S. W. (2001). Generating scenario trees for multistage problems. *Management Science, 47*, 295–307.

Kiwiel, K. C. (1990). Proximity control in bundle methods for convex non-differentiable minimization. *Mathematical Programming, 46*, 105–122.

Kucukyavuz, S. (2012). On mixing sets arising in chance constraint programming. *Mathematical Programming, 132*, 31–56.

Laporte, G., & Louveaux, F. V. (1993). The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters, 13*, 133–142.

Linderoth, J., Shapiro, A., & Wright, S. J. (2006). The empirical behavior of sampling methods for stochastic programming. *Annals of Operations Research, 142*, 215–241.

Linderoth, J. T., & Wright, S. J. (2003). Decomposition algorithms for stochastic programming on a computational grid. *Computational Optimization and Applications, 24*, 207–250.

Luedtke, J., & Ahmed, S. (2008). A sample approximation approach for optimization with probabilistic constraints. *SIAM Journal on Optimization, 19*, 674–699.

Mak, W. K., Morton, D. P., & Wood, R. K. (1999). Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters, 24*, 47–56.

Medova, E. (1998). Chance constrained stochastic programming for integrated services network management. *Annals of Operations Research, 81*, 213–229.

Nemirovski, A., Juditsky, A., Lan, G., & Shapiro, A. (2009). Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization, 19*, 1574–1609.

Ogryczak, W., & Ruszczyński, A. (2001). On the consistency of stochastic dominance and mean-semideviation models. *Mathematical Programming, 89*, 217–232.
Pennanen, T., & Koivu, M. (2005). Epi-convergent discretizations of stochastic programs via integration quadratures. *Numerische Mathematik, 100*, 141–163.
Pereira, M. V., & Pinto, L. M. (1991). Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming, 52*, 359–375.
Pflug, G. C. (2001). Scenario tree generation for multiperiod financial optimization by optimal discretization. *Mathematical Programming, 89*, 251–271.
Philpott, A. B., & Guan, Z. (2008). On the convergence of stochastic dual dynamic programming and related methods. *Operations Research Letters, 36*, 450–455.
Powell, W. B. (2010). *Approximate dynamic programming*. Hoboken: Wiley.
Prékopa, A. (1971). Logarithmic concave measures with application to stochastic programming. *Acta Scientiarium Mathematiarum(Szeged), 32*, 301–316. Strategic choice 789.
Rockafellar, R. T. (2007). Coherent approached to risk in optimization under uncertainty. In *Tutorials in operations research* (pp. 38–61). Hanover: INFORMS.
Rockafellar, R. T., & Uryasev, S. (2000). Optimization of conditional value at risk. *Journal of Risk, 2*, 21–41.
Rockafellar, R. T., & Wets, R. J.-B. (1991). Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research, 16*, 119–147.
Rockafellar, R. T., & Wets, R. J.-B. (1998). *Variational analysis*. Berlin: Springer.
Ruszczyński, A. (1986). A regularized decomposition method for minimizing a sum of polyhedral functions. *Mathematical Programming, 35*, 309–333.
Schultz, R. (1993). Continuity properties of expectation functions in stochastic integer programming. *Mathematics of Operations Research, 18*, 578–589.
Sen, S. (2010). Stochastic integer programming algorithms: Beyond benders' decomposition. In J. J. Cochran (Editor-in-Chief), *Wiley encyclopedia of operations research and management science*. Hoboken: Wiley.
Sen, S., Doverspike, R. D., & Cosares, S. (1994). Network planning with random demand. *Telecommunication Systems, 3*, 11–30.
Sen, S., & Higle, J. L. (2005). The C^3 theorem and a D^2 algorithm for large scale stochastic integer programming. *Mathematical Programming, 104*, 1–20.
Sen, S., & Sherali, H. D. (2006). Decomposition with branch-and-cut approaches for two-stage stochastic integer programming. *Mathematical Programming, 106*, 203–223.
Shapiro, A. (2011). Analysis of stochastic dual dynamic programming method. *European Journal of Operational Research, 209*, 63–72.
Shapiro, A., & Homem-de-Mello, T. (1998). A simulation-based approach to stochastic programming with recourse. *Mathematical Programming, 81*, 301–325.
Sherali, H. D., & Fraticelli, B. M. P. (2002). A modification of benders' decomposition algorithm for discrete subproblems: An approach for stochastic programs with integer recourse. *Journal of Global Optimization, 22*, 319–342.
Soyster, A. L. (1973). Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research, 21*, 1154–1157.
Wallace, S. W., & Ziemba, W. T. (2005). *Applications of stochastic programming*. Philadelphia: SIAM and MPS Publication.
Watson, J.-P., Woodruff, D., & Hart, W. (2010, to appear). PySP: Modeling and solving stochastic programs in python. *Mathematical Programming Computations*.
Yuan, Y., & Sen, S. (2009). Enhanced cut generation methods for decomposition-based branch-and-cut algorithms for two-stage stochastic mixed-integer programs. *INFORMS Journal on Computing, 21*, 480–487.

Strategic Assumption Surfacing and Testing (SAST)

A problem structuring method for use in situations where decisive action is obstructed by internal disagreements. Coherent sub-groups are formed with the purpose of advocating differing strategies, and each identifies the significant assumption on which its preferred strategy depends. The sub-groups are then reunited to debate the differences in assumptions, with the aim of achieving a compromise on assumptions so that a consensus strategy can be derived.

See

▶ Problem Structuring Methods

Strategic Choice Approach (SCA)

Strategic Choice Approach (SCA) is a problem structuring method centered on the management of uncertainty and commitment in strategic situations, where strategic refers to the advisability of considering particular decisions in the context of others. Strategic occasions can occur at any level. For fuller descriptions see Friend (2001), Friend and Hickling (2004).

The structure of the planning situation is elicited from stakeholders in a workshop format. This structure is built up in a participatory manner, with the aid of facilitators. SCA is a member of the Problem Structuring Methods family; within that

group it is notable for the variety of tools and techniques available to make progress with the problem. It has been widely used in diverse areas of public planning.

There are four modes of analysis within SCA. Switching between modes, which may be recursive, is guided by the facilitator. The modes are:

- Shaping – in which the stakeholder group identifies relevant areas for choice, and the linkages between them. They select a subset of these as a problem focus by reference to their urgency, importance and inter-connectedness
- Designing – here the options for action within each of the selected decision areas are identified, plus any incompatibilities between options in different decision areas. The feasible decision schemes (consisting of one option choice within each decision area) are derived using the AIDA algorithm
- Comparing – criteria for choice, often non-quantitative, are agreed by the group. A small number of the decision schemes are short-listed, and pairwise comparisons are made between short-listed schemes. The group agrees the relative advantage on each criterion between the two schemes, commonly revealing significant uncertainties.
- Choosing – bearing in mind the surfaced uncertainties, a 'progress package' is agreed consisting of partial commitments, explorations to reduce key uncertainties areas, contingency plans, and a timetable for later choices.

The progress package summarizes the outcomes of a SCA application. Other outputs include improved understanding of the problem area and better working relations among group members.

See

▶ Problem Structuring Methods

References

Friend, J. (2001). The strategic choice approach. In J. Rosenhead & J. Mingers (Eds.), *Rational analysis for a problematic world revisited* (pp. 115–149). Chichester: Wiley.

Friend, J., & Hickling, A. (2004). *Planning under pressure: The strategic choice approach* (3rd ed.). Oxford: Elsevier.

Strategic Options Development and Analysis (SODA)

A problem structuring method for group decision making. Individual cognitive maps are elicited for participants, and then merged into a strategic map which is used in workshop mode to facilitate discussion and commitment.

See

▶ Problem Structuring Methods

References

Ackerman, F., & Eden, C. (2010). Strategic options development and analysis. In M. Reynolds & S. Holwell (Eds.), *Systems approaches to managing change: A practical guide* (pp. 135–190). New York: Springer. Chapter 4.

Strictly Quasi-Concave Function

A function $f(x)$ is strictly quasi-concave over a convex set S if for any two points $x_1 \neq x_2$ in S and for any $0 < \alpha < 1$, $f(x_2) \geq f(x_1)$ implies that $f(\alpha x_1 + (1 - \alpha)\, x_2) > f(x_1)$.

See

▶ Concave Function
▶ Convex Function
▶ Quasi-Concave Function
▶ Quasi-Convex Function

Strictly Quasi-Convex Function

A function $f(x)$ is strictly quasi-convex over a convex set S if for any two points $x_1 \neq x_2$ in S and for any $0 < \alpha < 1$, $-f(x_2) \geq -f(x_1)$ implies that $-f(\alpha x_1 + (1 - \alpha)\, x_2) > -f(x_1)$.

See

- ▶ Concave Function
- ▶ Convex Function
- ▶ Quasi-Concave Function
- ▶ Quasi-Convex Function

Strong Duality Theorem

Consider the following primal linear-programming problem and its dual problem:

$$\begin{aligned} &\textit{Dual} \\ &\text{Maximize} \quad \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} \\ &\text{subject } \textit{to} \quad \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} \leq \boldsymbol{c} \\ &\qquad\qquad \boldsymbol{y} \geq \boldsymbol{0} \\ &\textit{Primal} \\ &\text{Minimize} \quad \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \\ &\text{subject } \textit{to} \quad \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \\ &\qquad\qquad \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

The strong duality theorem is usually stated as follows: If either the primal or the dual has a finite optimal solution, then the other problem has a finite optimal solution, and the optimal values of their objective functions are equal, i.e.,

$$\text{minimum } \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} = \text{maximum } \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y}$$

The weak duality theorem basically relaxes the equality result to a bound by removing the optimizing operator as in the following statement: If $\boldsymbol{x}$ is a feasible solution to the primal problem and $\boldsymbol{y}$ is a feasible solution to the dual problem, then $\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} \leq \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$.

Strongly NP-Complete (NP-Hard)

- ▶ Computational Complexity

Strongly Polynomial-time Algorithm

An algorithm whose running time is bounded polynomially by a function only of the inherent dimensions of the problem and is independent of the sizes of the numerical data of the instance.

See

- ▶ Computational Complexity

Structural Variables

The original variables of a linear-programming problem as differentiated from slack, surplus and artificial variables. Structural variables are usually the variables of interest and have a physical interpretation such as production or shipments. They appear in the original defining inequalities or equations prior to the conversion of the problem to all equations.

See

- ▶ Linear Inequality
- ▶ Linear Programming
- ▶ Logical Variables
- ▶ Slack Variable
- ▶ Surplus Variable

Structure Function

- ▶ System Reliability

Structured Modeling

Arthur M. Geoffrion
University of California, Los Angeles, CA, USA

Introduction

Structured modeling was developed as a comprehensive response to perceived shortcomings of modeling systems available in the 1980s. It is

a systematic way of thinking about models and their implementations, based on the idea that every model can be viewed as a collection of distinct elements, each of which has a definition that is either primitive or based on the definition of other elements in the model. Elements are categorized into five types (so-called primitive entity, compound entity, attribute, function, and test), grouped by similarity into any number of classes called genera, and organized hierarchically as a rooted tree of modules so as to reflect the model's high-level structure. It is natural to diagram the definitional dependencies among elements as arcs in a directed acyclic graph. Moreover, this dependency graph can be computationally active because every function and test element has an associated mathematical expression for computing its value.

Using a model for any specific purpose involves subjective intentions. Structured modeling makes a sharp distinction between the resulting user-defined problems or tasks associated with a model, and the relatively objective model per se. A typical problem or task has to do with ad hoc query, drawing inferences, evaluating model behavior with specified inputs, determining a constrained solution, or optimization, and requires applying a computerized model manipulation tool (solver). For certain recurring kinds of problems and tasks, these tools are highly developed and readily available for incorporation into a structured modeling software system.

The theoretical foundation of structured modeling is formalized in Geoffrion (1989), which presents a rigorous semantic framework that deliberately avoids committing to a representational formalism. The framework is semantic, because it casts every model as a system of definitions styled to capture semantic content. Ordinary mathematics, in contrast, typically leaves more of the meaning implicit. Twenty-eight definitions and eight propositions establish the notion of model structure at three levels of detail (so-called elemental, generic, and modular structure), the essential distinction between model class and model instance, certain related concepts and constructs, and basic theoretical properties. This framework has points in common with certain ideas found in the computer science literature on knowledge representation, programming language design, and semantic data modeling, but is designed specifically for modeling as practiced in OR/MS and related fields (Geoffrion 1987; Section 4).

Structured Modeling Languages

An executable model description language called SML (Structured Modeling Language) fully supports structured modeling's semantic framework (Geoffrion 1992). Other languages for (at least parts of) structured modeling also exist, including ones that are graph-based, logic-based, SQL-oriented, subscript- free, or object-oriented. SML can be viewed in terms of four upwardly compatible levels of increasing expressive power. The first level encompasses simple definitional systems and directed graph models such as those found in Harary et al. (1965) The second level covers more complex extensions of these, spreadsheet models, numeric formulas, and propositional calculus models. The third level encompasses mathematical programming and predicate calculus models with simple indexing over sets and Cartesian products. Finally, the fourth level covers sparse versions of the above plus relational and semantic database models.

Exhibits A and B, taken from Geoffrion (1987), show an SML schema (third level) specifying the general structure of the classical feedmix model, and sample SML elemental detail tables specifying model elements. The latter, together with the schema, yield a specific feedmix model instance.

Space does not permit a proper description of SML's syntax, but a few hints are as follows. Schemas are organized as a tree of paragraphs whose leaves are the genera and whose interior nodes are the modules. The boldfaced part of each paragraph is the formal definition of the genus or module, as the case may be, and the rest consists of documentary comments about the formal part that are informal except for conventions about the use of underlining and upper case. The formal definition of a genus paragraph begins with the name of the genus, a parenthetical statement of definitional dependencies (if any), a slash-delimited statement of genus type, a colon-announced statement of data type if an attribute genus, and a semicolon-announced mathematical expression called a generic rule if a function or test genus. The formal definition of a module paragraph consists only of its name. Note that a schema is always specified independently

Structured Modeling, Exhibit A SML schema for the classical feedmix model

&NUT_DATA NUTRIENT DATA

NUTRi /pe/ There is a list of NUTRIENTS.

MIN (NUTRi) /a/ : Real + For each NUTRIENT there is a MINIMUM DAILY REQUIRMENT (units per day per animal).

&MATERIALS MATERIALS DATA

MATERIALm /pe/ There is a list of MATERIALS that can be used for feed.

UCOST (MATERIALm) /a/ Each MATERIAL has a UNIT COST ($ per pound of material).

ANALYSIS (NUTRi, MATERIALm) /a/ : Real + For each NUTRIENT-MATERIAL combination, there is an ANALYSIS (units of nutrient per pound of material).

Q (MATERIALm) /va/ : Real + The QUANTITY (pounds per day per animal) of each MATERIAL is to be chosen.

NLEVEL (ANALYSISi., Q) /f/ ; **@SUMm (ANALYSISim * Qm)** Once the QUANTITIES are chosen, there is a NUTRITON LEVLE (units per day per animal) for each NUTRIENT calculable from the ANALYSIS.

T:NLEVEL (NLEVELi, MINi) /t/ ; NLEVELi > = MINi For each NUTRIENT there is a NUTRITION TEST to determine wherther the NUTRITION LEVEL is at least as large as the MINIMUM DAILY REQUIREMENT.

TOTCOST (UCOST, Q) /f/ ; @SUMm (UCOSTm * Qm) There is a TOTAL COST (dollars per day per animal) associated with the chosen QUANTITIES.

Structured Modeling, Exhibit B Sample elemental detail tables for the feedmix schema

NUTR

MATERIAL

NUTR	INTERP	MIN	MATERIAL	INTERP	UCOST
P	Protein	16	std	Standard Feed	1.20
C	Calcium	4	add	Additive	3.00

ANALYSIS

NUTR	MATERIAL	ANALYSIS
P	std	4.00
P	add	14.00
C	std	2.00
C	add	1.00

Q

MATERIAL	Q
std	2.00
add	0.50

NLEVEL

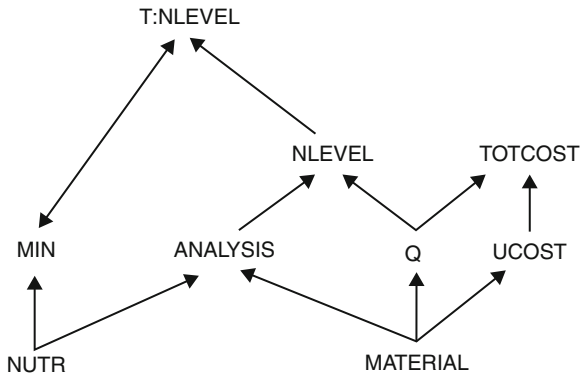
TOTCOST

NUTR	NLEVEL	T:NLEVEL	TOTCOST
P	15.00	FALSE	3.90
C	4.50	TRUE	

of any problem or task that might be posed on it. A common problem associated with the above schema is to find values for all Q elements such that all T:NLEVEL elements evaluate to true and the value of the TOTCOST element is minimal.

The structure and sequence of the following elemental detail tables are determined procedurally from the schema. Each table is named, has column names that usually coincide with genus names, and has a row for each element of the corresponding genus.

Figure 1 shows the so-called genus graph associated with the above schema. It represents definitional dependencies at the level of genera.



Structured Modeling, Fig. 1 Genus graph for the feedmix schema

Structured Modeling Systems

Owing to the design of the underlying semantic framework and of SML itself, SML-based modeling systems can have certain features often lacking in modeling systems of more conventional design, including:

- Error checking, of the formal specification of general model structure, that is exhaustive with respect to the underlying semantic framework
- Detailed semantic connections among model parts, a feature that facilitates maintaining, enhancing, and integrating models, and that enables automatic generation of several kinds of model reference documents useful for communication, debugging, model maintenance and evolution, and other essential activities;
- The ability, owing to the generality of structured modeling's view of models as definitional systems, for a single modeling system to accommodate a wide variety of modeling paradigms, which leads to easier model integration and many of the benefits of standardization;
- Browsable definitional dependency graphs at three levels of abstraction, constructs useful for visualizing and communicating the general structure of any model;
- The use of hierarchical organization as an approach to managing model complexity, and also as a visual device for model navigation;
- Automatic generation of relational data table designs for model instance data, a feature that facilitates exploiting relational database tools for data management;
- Partial consistency checking of SML's informal sublanguage for documenting the formal model specification, and also partial consistency and completeness checking of formal specifications by reference to this documentation;
- Complete independence between the general structure of a class of models and instantiating data, a feature that promotes the reuse of each of these, conciseness, efficient communication, and dimensional flexibility;
- Complete independence between models and solvers, a feature that promotes using multiple solvers with a single model, multiple models with a single solver, and conceptual clarity.

A research prototype implementation exhibiting the above features is described in Geoffrion (1991) and Neustadter et al. (1992). The first paper references several other research prototypes for structured modeling with different emphases, including: graph- based modeling, hybrid information/mathematical modeling systems, model management with a SQL database server in a networked environment, optimization-based applications, statistical analysis, and syntax-directed model editing. Other implementations include those of Chari and Sen (1998), Hamacher (1995), Iyer et al. (2005), Makowski (2005), Maturana et al. (2004), and Wright et al. (1997).

Concluding Remarks

An ample foundation has been laid for the development of software intended for commercial application. Experimental studies and a few real applications have taken place in the consumer appliance, food, industrial gases, oil, steel, and tire industries, and there are ongoing applications to environmental policy-making at IIASA.

Promising topics for future work include discrete-event simulation (Lenard 1993), graph-based modeling (Jones 1992), language-directed editors (Vicuña 1990), object-oriented systems (Muhanna 1993), model integration (Dolk and Kottemann 1993; Gagliardi and Spera 1995), distributed and semantic-web-based service-oriented architectures for model management (El-Gayar and Deokar 2008; Deokar et al. 2010), improved languages for model

definition and manipulation, applications to early and late modeling life-cycle phases not supported by conventional modeling systems, and structured modeling-based enhancements and usage disciplines for other modeling approaches and systems. See Geoffrion (1999) for an extensive annotated bibliography on structured modeling, Krishnan and Chari (2000) for a broad survey of the literature and research opportunities of model management that is explicitly in accord with structured modeling's modeling-lifecycle worldview, and Dolk (2010) for a thoughtful retrospective on structured modeling.

See

- ▶ Algebraic Modeling Languages for Optimization
- ▶ Mathematical Model
- ▶ Model Management

References

Chari, K., & Sen, T. (1998). An implementation of a Graph-Based Modeling System for Structured Modeling (GBMS/SM). *Decision Support Systems, 22*, 103–120.

Deokar, A. V., El-Gayar, O. F., Aljafari, R. (2010). Developing a semantic web-based distributed model management system: Experiences and lessons learned. In *Proceedings of the 43rd annual Hawaii international conference on system sciences* (pp. 1–10) [CD-ROM]. Honolulu: Computer Society Press.

Dolk, D. (2010). Structured modeling and model management. In M. S. Sodhi & C. S. Tang (Eds.), *A long view of research and practice in operations research and management science: The past and the future. Vol. 148: International series in operations research & management science* (Chap. 5, pp. 63–88). New York: Springer.

Dolk, D., & Kottemann, J. (1993). Model integration and a theory of models. *Decision Support Systems, 9*, 51–63.

El-Gayar, O. F., & Deokar, A. V. (2008). Distributed model management: Current status and future directions. In F. Adam & P. Humphreys (Eds.), *Encyclopedia of decision making and decision support technologies* (Vol. 1, pp. 272–277). Hershey: IGI Global.

Gagliardi, M., & Spera, C. (1995). Toward a formal theory of model integration. *Annals of Operations Research, 58*, 405–440.

Geoffrion, A. M. (1987). An introduction to structured modeling. *Management Science, 33*, 547–588.

Geoffrion, A. M. (1989). The formal aspects of structured modeling. *Operations Research, 37*, 30–51.

Geoffrion, A. M. (1991). FW/SM: A prototype structured modeling environment. *Management Science, 37*, 1513–1538.

Geoffrion, A. M. (1992). The SML language for structured modeling. *Operations Research, 40*, 38–75.

Geoffrion, A. M. (1999). Structured modeling: Survey and future research directions. *Interactive Transactions of ORMS* [Online], *1*(3). http://itorms.pubs.informs.org

Hamacher, S. (1995). *Modeling systems for operations research problems: Study and applications* (Ph.D. dissertation). Paris: Industrial Engineering, Ecole Paris Centrale. 235 p (in French).

Harary, F., Norman, R., & Cartwright, D. (1965). *Structural models: An introduction to the theory of directed graphs*. New York: Wiley.

Iyer, B., Shankaranarayanan, G., & Lenard, M. (2005). Model management decision environment: A web service prototype for spreadsheet models. *Decision Support Systems, 40*, 283–304.

Jones, C. V. (1992). Attributed graphs, graph-gram-mars, and structured modeling. *Annals of Operations Research, 38*, 281–324 (Special volume on *Model management in operations research* edited by B. Shetty, H. Bhargava, and R. Krishnan).

Krishnan, R., & Chari, K. (2000). Model management: Survey, future directions and a bibliography. *Interactive Transactions of ORMS* [Online], *3*(1). http://itorms.pubs.informs.org

Lenard, M. L. (1993). A prototype implementation of a model management system for discrete-event simulation models. In *Proceedings of the 1993 winter simulation conference* (pp. 560–568). Piscataway: IEEE.

Makowski, M. (2005). A structured modeling technology. *European Journal of Operational Research, 166*, 615–648.

Maturana, S., Ferrer, J. C., & Barañao, F. (2004). Design and implementation of an optimization-based decision support system generator. *European Journal of Operational Research, 154*, 170–183.

Muhanna, W. (1993). An object-oriented framework for model management and DSS development. *Decision Support Systems, 9*, 217–229.

Neustadter, L., Geoffrion, A., Maturana, S., Tsai, Y., & Vicuña, F. (1992). The design and implementation of a prototype structured modeling environment. *Annals of Operations Research, 38*, 453–484. Special volume on *Model management in operations research* edited by B. Shetty, H. Bhargava, and R. Krishnan.

Vicuña, F. (1990). *Semantic formalization in mathematical modeling languages* (Ph.D. dissertation). Computer Science Department, UCLA.

Wright, G., Worobetz, N. D., Kang, M., Mookerjee, R., & Chandrasekharan, R. (1997). OR/SM: A prototype integrated modeling environment based on structured modeling. *INFORMS Journal on Computing, 9*, 134–153.

SUB Problem

- ▶ Simple Upper-bounded Problem (SUB)

Subderivative

For a real-valued convex function f defined on an open interval of the real line, a subderivative at a point x^0 is any real number y that satisfies

$$f(x) - f(x^0) \geq y(x - x^0) \text{ for all } x$$

See

- ▶ Subdifferential
- ▶ Subgradient

Subdifferential

Set of all subderivatives or subgradients at a point.

See

- ▶ Convex Optimization
- ▶ Lagrangian Relaxation
- ▶ Subderivative
- ▶ Subgradient

Subgradient

For a real-valued convex function f defined on a convex open set in R^n, a vector y is a subgradient at a point $\boldsymbol{x^0} \in R^n$ if for all $\boldsymbol{x} \in R^n$

$$f(\boldsymbol{x}) - f(\boldsymbol{x^0}) \geq \boldsymbol{y} \cdot (\boldsymbol{x} - \boldsymbol{x^0}),$$

where the '$\cdot$' operator denotes inner product.

See

- ▶ Convex Optimization
- ▶ Lagrangian Relaxation
- ▶ Subdifferential

Subjective Probability

- ▶ Bayesian Decision Theory, Subjective Probability, and Utility
- ▶ Decision Analysis

Suboptimization

The finding of a solution to an optimization problem by a procedure that does not guarantee that the solution will be optimal. The procedure usually includes heuristic rules that help eliminate the geneation of poor solutions.

See

- ▶ Heuristics

Super-Sparsity

In most large-scale mathematical-programming problems, especially linear-programming problems, the number of nonzero elements in the problem matrix is quite small. Such problems are said to have a low density. Further, it has been noted that the number of distinct numerical values in the problem matrix is usually much smaller than the number of nonzero coefficients. This characteristic is known as super-sparsity. Computational savings in storage and processing time can be achieved by taking advantage of super-sparsity, as follows. Each distinct numerical value is recorded once in a value table stored in main memory. Each nonzero coefficient is recorded in an index array by means of a number triple: row index, column index, and a pointer. The pointer locates the coefficient's numerical value in the value table.

See

- ▶ Density
- ▶ Large-Scale Systems
- ▶ Sparse Matrix

Supplemental Variables

An analysis technique that introduces additional variables in the process state definition to allow non-Markovian systems to be made Markovian.

See

- ▶ Markov Processes
- ▶ Queueing Theory

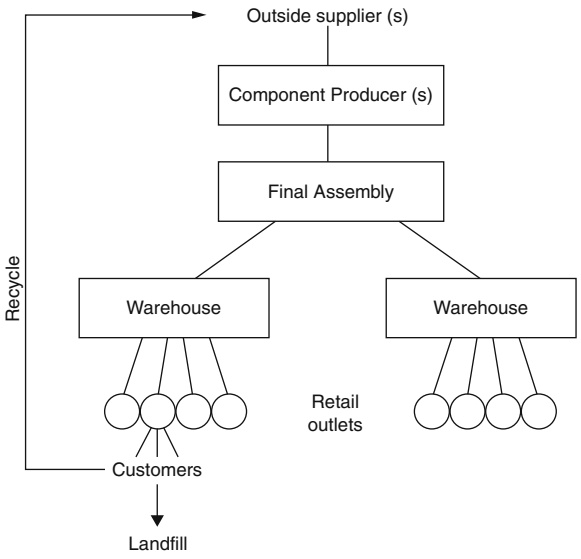
Supply Chain Management

M. Eric Johnson[1] and David F. Pyke[2]
[1]Dartmouth College, Hanover, NH, USA
[2]University of San Diego, San Diego, CA, USA

Introduction

Supply chain management (SCM) is the term used to describe the operational management of the flow of materials, information, and funds across the entire supply chain, from suppliers to component producers to final assemblers to distribution (warehouses and retailers), and ultimately to the consumer. In fact, it often includes after-sales service and returns or recycling. Figure 1 is a schematic of a supply chain. In contrast to multiechelon inventory management, that coordinates inventories at multiple locations, SCM typically involves coordination of information and materials among multiple firms. Much of the material in this article is based on Chapter 12 of Silver et al. (1998) and Johnson and Pyke (2000a).

Supply chain management has generated much interest in the OR/MS community in recent years for a number of reasons. Many managers now realize that actions taken by one member of the chain can influence the profitability of all others in the chain. Certain industries use other terms in place of SCM. For example, many grocery industry executives are pursuing efficient consumer response (ECR), the equivalent of just-in-time distribution or "continuous replenishment." Initiatives such as ECR are within the purview of supply chain management.



Supply Chain Management, Fig. 1 A schematic of a supply chain

Many firms now think of competition as pitting their supply chain against other supply chains, rather than their firm against other individual firms. Also, as firms successfully streamline their own operations, the next opportunity for improvement is through better coordination with their suppliers and customers. The costs of poor coordination can be extremely high. In the Italian pasta industry, consumer demand is quite steady throughout the year. However, because of trade promotions, volume discounts, long lead times, full-truckload discounts, and end-of-quarter sales incentives, the orders seen at the manufacturers are highly variable (Hammond 1994). In fact, the variability increases in moving up the supply chain from consumer to grocery store to distribution center to central warehouse to factory, a phenomenon that is often called the bullwhip effect (Fig. 2). The bullwhip effect has been experienced by many students playing the "Beer Distribution Game" (Sterman 1989; Sterman 1992; Chen and Samroengraja 2000; Jacobs 2000). The costs of this variability are high — inefficient use of production and warehouse resources, high transportation costs, and high inventory costs, to name a few. Acer America, Inc. sacrificed $20 million in profits by paying $10 million for air freight to keep up with surging demand, and then paying $10 million more later when that inventory became obsolete (Business Week 1996, p. 72; Towill and Vecchio 1994; Berry et al. 1995; Buzzell and Ortmeyer 1995).

Supply Chain Management, Fig. 2 An illustration of the bullwhip effect

It seems that integration, long the dream of management gurus, has finally been sinking into the minds of managers. Some would argue that managers have long been interested in integration, but the lack of information technology made it impossible to implement a more "systems-oriented" approach. Clearly industrial dynamics researchers dating back to the 1950s (Forrester 1958; Forrester 1961) have maintained that supply chains should be viewed as an integrated system. With the recent explosion of inexpensive information technology, it seems only natural that business would become more supply chain focused. However, while technology is clearly an enabler of integration, it alone can not explain the radical organizational changes in both individual firms and whole industries. Changes in both technology and management theory set the stage for integrated supply chain management. One reason for the change in management theory is the power shift from manufacturers to retailers. Wal-Mart, for instance, has forced many manufacturers to improve their management of inventories, and even to manage inventories of their products at Wal-Mart.

While integration, information technology and retail power may be key catalysts in the surge of interest surrounding supply chains, electronic-based business — eBusiness — is fueling even stronger excitement. eBusiness facilitates the virtual supply chain, and as companies manage these virtual networks, the importance of integration is magnified. Firms like Amazon are superb at managing the flow of information and funds, via the Internet and electronic funds transfer. Now, the challenge is to efficiently manage the flow of products.

Some would argue that the language and metaphors are wrong. "Chains" evoke images of linear, unchanging, and powerless. "Supply" feels pushy and reeks of mass production rather than mass customization. Better names, like "demand networks" or "customer driven webs" have been proposed. Yet, for now, the name "supply chain" seems to have stuck. And under any name, the future of supply chain management appears bright.

Key Components of Supply Chain Management

Supply chain management is an enormous topic covering multiple disciplines and employing many quantitative and qualitative tools. Within the last few years, several textbooks on supply chain have arrived on the market providing both managerial overviews and detailed technical treatments. For examples of managerial introductions to supply chain, seeby Copacino (1997), by Fine (1998), and Handfield and Nichols (1998), and for logistics texts, see Lambert et al. (1997) and by Ballou (1998). For more technical, model-based treatments, see Silver et al. (1998) and Simchi-Levi et al. (1998). Tayur et al. (1999) is an extensive collection of research papers, while Johnson and Pyke (2000b) is a collection papers on teaching supply chain management. Also, there are several casebooks that give emphasis to global management issues, includingby Taylor (1997),

by Flaherty (1996), and Dornier et al. (1998). Introductory articles include Cooper et al. (1997b), by Davis (1993), Johnson (1998a), and Lee and Billington (1992).

To help order the discussion, the supply chain management is divided into twelve areas [see Johnson and Pyke (2000a) for a list of teaching cases and popular press articles that fit within each area].

Each area represents a supply chain issue facing the firm. For any particular problem or issue, managers may apply analysis or decision-support tools. For each of the twelve areas, a brief description of the basic content is provided, along with a few relevant research papers OR/MS-based tools that may aid analysis and decision support are also mentioned. For a more detailed review of recent research and teaching in supply chain management, see Ganeshan et al. (1999) and Johnson and Pyke (2000a).

The twelve categories defined are:

- location
- transportation and logistics
- inventory and forecasting
- marketing and channel restructuring
- sourcing and supplier management
- information and electronic mediated environments
- product design and new product introduction
- service and after sales support
- reverse logistics and green issues
- outsourcing and strategic alliances
- metrics and incentives
- global issues.

Location pertains to both qualitative and quantitative aspects of facility location decisions. This includes models of facility location, geographic information systems (GIS), country differences, taxes and duties, transportation costs associated with certain locations, and government incentives (Hammond and Kelly 1990). Exchange rate issues fall in this category, as do economies and diseconomies of scale and scope. Decisions at this level set the physical structure of the supply chain and therefore establish constraints for more tactical decisions. Binary integer programming models play a role here, as do simple spreadsheet models and qualitative analyses. There are many advanced texts specially dedicated to the modeling aspects of location (Drezner 1996) and most books on logistics also cover the subject. Simchi-Levi et al. (1998) presented a substantial treatment of GIS, while Dornier et al. (1998) dedicated a chapter to issues of taxes, duties, exchange rates, and other global location issues (Brush et al. 1999). Ballou and Masters (1999) examined several software products that provide optimization tools for solving industrial location problems.

The transportation and logistics category encompasses all issues related to the flow of goods through the supply chain, including transportation, warehousing, and material handling. This category includes many of the current trends in transportation management including vehicle routing (Bodin 1990; Gendreau et al. 1996; Anily and Bramel 1999), dynamic fleet management with global positioning systems, and merge-in-transit. Also included are topics in warehousing and distribution such as cross docking (Kopczak et al. 1995) and materials handling technologies for sorting, storing, and retrieving products (Johnson and Brandeau 1999; Johnson 1998b).

Because of globalization and the spread of out-sourced logistics, this category has received much attention in recent years. However, a separate category will examine issues specifically related to outsourcing and logistics alliances. Both deterministic (such as linear programming and the traveling salesman problem) and stochastic optimization models (stochastic routing and transportation models with queueing) often are used here, as are spreadsheet models and qualitative analysis. Recent management literature has examined the changes within the logistics functions of many firms as the result of functional integration (Greis and Kasarda 1997) and the role of logistics in gaining competitive advantage (Fuller et al. 1993).

Inventory and forecasting includes traditional inventory and forecasting models. Inventory costs are some of the easiest to identify and reduce when at-tacking supply chain problems. Simple stochastic inventory models can identify the potential cost savings from, for example, sharing information with supply chain partners (Lee and Nahmias 1993), but more complex models are required to coordinate multiple locations. A few years ago, multiechelon inventory theory captured most of the research in this area that would apply to supply chains. However, in nearly every case, multiechelon inventory models assume a single decision-maker. Supply chains, unfortunately, confront the problem of multiple firms, each with its own decision-maker and objectives.

Of course, there are many full texts on the subject, such as Silver et al. (1998) and Graves et al. (1993). Useful managerial articles focusing on inventory and forecasting includeby Davis (1993) and Fisher et al. (1994).

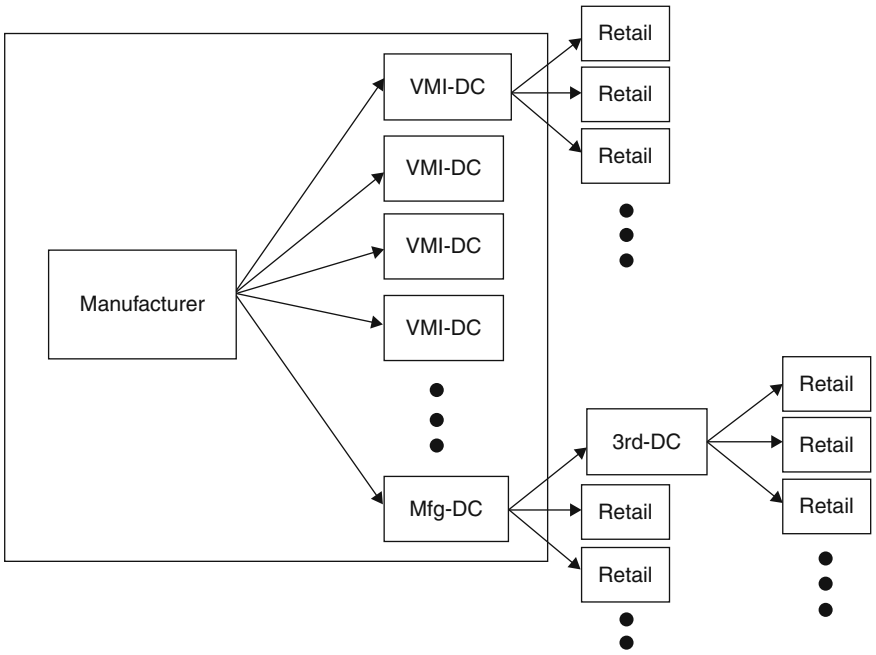
Clark and Scarf (1960) performed one of the earliest studies in serial systems with probabilistic demand. They introduced the concept of an imputed penalty cost, wherein a shortage at a higher echelon generates an additional cost. This cost decomposes the multiechelon system into a series of stages so that, assuming centralized control and the availability of global information, the ordering policies can be optimized. Lee and Whang (1999a) and Chen (1996) both proposed performance measurement schemes for individual managers that allow for decentralized control (so that each manager makes decisions independently), and in certain instances, local information only. The result is a solution that achieves the same optimal solution as if centralized control and global information were assumed.

Marketing and channel restructuring includes fundamental thinking on supply chain structure (Fisher 1997) and covers the interface with marketing that emerges from having to deal with down-stream customers (Narus and Anderson 1996). While the inventory category addresses the quantitative side of these relationships, this category covers relationship management, negotiations and even the legal dimension. Most importantly, it examines the role of channel management (Anderson et al. 1997) and supply chain structure in light of the well-studied phenomena of the bullwhip effect that was noted in the introduction.

The bullwhip effect has received enormous attention in the research literature. Many authors have noted that central warehouses are designed to buffer the factory from variability in retail orders. The inventory held in these warehouses should allow factories to smooth production while meeting variable customer demand. However, empirical data suggest that exactly the opposite happens (e.g., see Blinder 1981, and Baganha and Cohen 1998). Orders seen at the higher levels of the supply chain exhibit more variability than those at levels closer to the customer. In other words, the bullwhip effect is real. Typically, causes include those noted in the introduction, as well as the fact that retailers and distributors often over-react to shortages by ordering more than they need. Lee et al. (1997) showed how four rational factors help to create the bullwhip effect: demand signal processing (if demand increases, firms order more in anticipation of further increases, thereby communicating an artificially high level of demand); the rationing game (there is, or might be, a shortage so a firm orders more than the actual forecast in the hope of receiving a larger share of the items in short supply); order batching (fixed costs at one location lead to batching of orders); and manufacturer price variations (which encourage bulk orders). The latter two factors generate large orders that are followed by small orders, which implies increased variability at upstream locations.

Some recent innovations, such as increased communication about consumer demand, via electronic data interchange (EDI) and the Internet, and everyday low pricing (EDLP) (to eliminate forward buying of bulk orders), can mitigate the bullwhip effect. See Bell et al. (1998a, 1998b) for a discussion of EDLP versus High-Low pricing. They showed with a simple model that High-Low pricers can charge a higher average price without risking the loss of rational customers. In addition, Baganha and Cohen (1998) noted that, if locations that are designed to buffer the factory from variability in retail orders follow the optimal policy, the variance can in fact be reduced. In particular, these locations should account for autocorrelation in the demand process; that is, if a retailer orders today, it is unlikely that it will order in the next few days (Bourland et al. 1996; Srinivasan et al. 1994; Lee et al. 1999).

In fact, the number of firms ordering, and receiving orders, via EDI and the Internet is exploding. The information available to supply chain partners, and the speed with which it is available, has the potential to radically reduce inventories and increase customer service. For example, see Moinzadeh and Aggarwal (1997) and Lee and Whang (1999b). Milgrom and Roberts (1988) noted that inventory and information are substitutes. Other initiatives can also mitigate the bullwhip effect. For example, changes in pricing and trade promotions (Buzzell et al. 1990) and channel initiatives, such as vendor-managed inventory (VMI), coordinated forecasting and replenishment (CFAR), and continuous replenishment (Fites 1996; Verity 1996; Waller et al. 1999), can significantly reduce demand variance. VMI is one of the most widely discussed partnering initiatives for improving multi-firm supply chain efficiency. Popularized in the late 1980s by Wal-Mart and Procter and Gamble,



Supply Chain Management, Fig. 3 Typical VMI implementation (adapted from Waller et al. 1999)

VMI became one of the key programs in the grocery industry's pursuit of efficient consumer response and the garment industry's quick response. Successful VMI initiatives have been trumpeted by other companies in the United States, including Campbell Soup and Johnson and Johnson, and by European firms like the pasta manufacturer Barilla.

In a VMI partnership, the supplier — usually the manufacturer but sometimes a reseller or distributor — makes the main inventory replenishment decisions for the consuming organization. This means the supplier monitors the buyer's inventory levels (physically or via electronic messaging) and makes periodic resupply decisions regarding order quantities, shipping, and timing. Transactions customarily initiated by the buyer (like purchase orders) are initiated by the supplier instead. Indeed, the purchase order acknowledgment from the supplier may be the first indication that a transaction is taking place; an advance shipping notice informs the buyer of materials in transit. Thus the manufacturer is responsible for both its own inventory and the inventory stored at is customers' distribution centers (Fig. 3).

Because many of these initiatives involve channel partnerships and distribution agreements, this category also contains important information on pricing, along with anti-trust and other legal issues (Train 1998). These innovations require interfirm, and often intrafirm, cooperation and coordination that can be difficult to achieve.

While marketing focuses downstream in the supply chain, sourcing and supplier management looks upstream to suppliers. Make/buy decisions (Venkatesan 1992; Carroll 1993; Christensen 1994; Quinn and Hilmer 1994; Kelley 1995; Robertson and Langlois 1995) fall into this category, as does global sourcing (Little 1995; Pyke 1994). The location category addresses the location of a firm's own facilities, while this category pertains to the location of the firm's suppliers. Supplier relationship management falls into this category as well (McMillan 1990; Womack et al. 1991). Some firms are putting part specifications on the Web so that dozens of suppliers can bid on jobs. GE, for instance, has developed a trading process network that allows many more suppliers to bid than was possible before. The automotive assemblers have developed a similar capability. Independent Internet firms, such as Digital Market, are providing services focused on certain product categories. Other firms are moving in the opposite direction by reducing the number of suppliers, in some cases to a sole source (Helper and Sako 1995; Cusumano and Takeishi 1991). Determining the number of suppliers and the best

way to structure supplier relationships is becoming an important topic in supply chains (Cohen and Agrawal 1996; Dyer 1996; Magretta 1998; Pyke (1998).

Much of the research in this area makes use of game theory to understand supplier relationships, contracts, and performance metrics. See, for instance, Cachon and Lariviere (1999), by Cachon (1997), and Tsay et al. (1999).

The information and electronic mediated environments category addresses long-standing applications of information technology to reduce inventory (Woolley 1997) and the rapidly expanding area of electronic commerce (Benjamin and Wigand 1997; Schonfeld 1998). Often this subject may take a more systems orientation, examining the role of systems science and information within a supply chain (Senge 1990). Such a discussion naturally focuses attention on integrative ERP software such as SAP (Whang et al. 1995), Baan and Oracle, as well as supply chain offerings such as i2's Rhythm and Peoplesoft's Red Pepper. The many supply chain changes wrought by electronic commerce are particularly interesting to examine, including both the highly publicized retail channel changes (like Amazon.com) and the more substantial business to business innovations (like the GE trading process network). It is here that OR/MS interfaces most directly with information technology and strategy, which again creates opportunities for cross-functional integration (Lee and Whang 1999b).

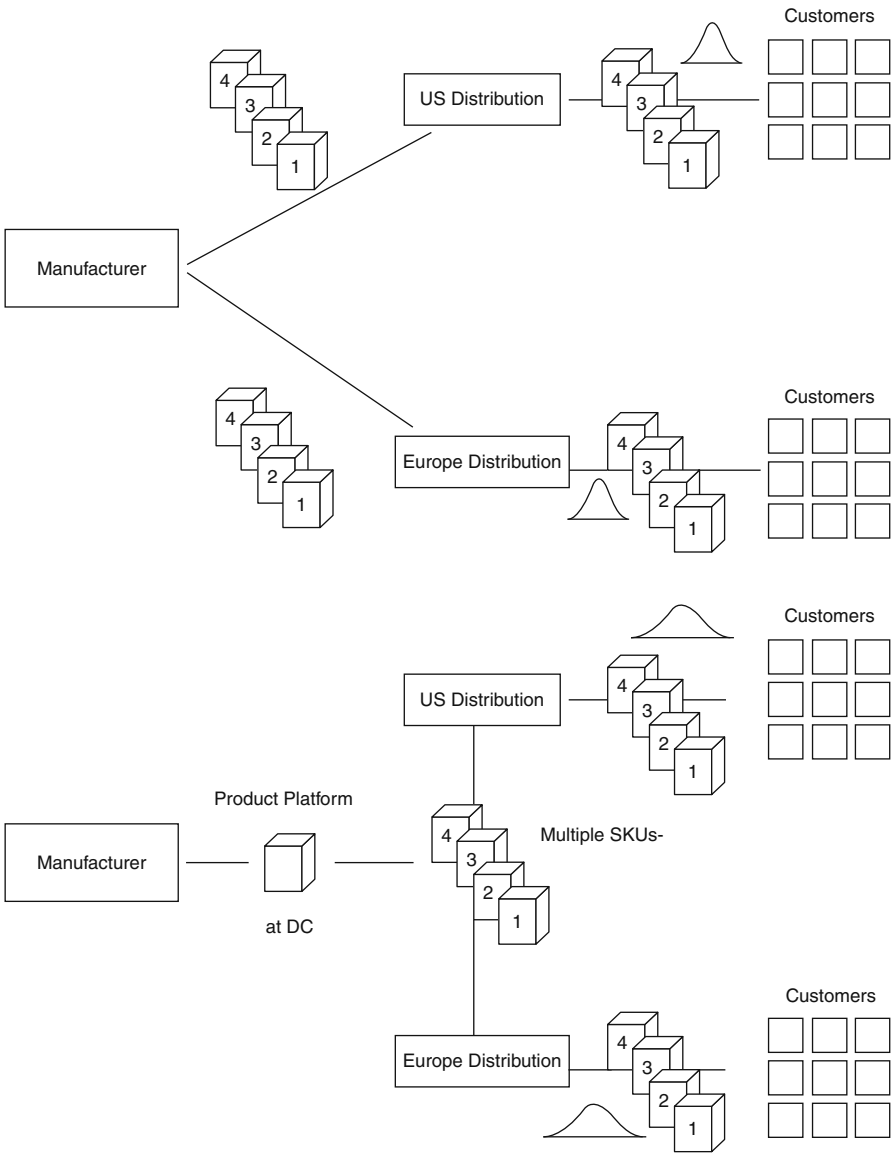
Product design and new product introduction deals with design issues for mass customization, delayed differentiation, modularity and other issues for new product introduction. With the increasing supply chain demands of product variety (Gilmore and Pine 1997) and customization (McCutcheon et al. 1994), there is an increasing body of research available. One of the most exciting applications of supply chain thinking is the increased use of postponed product differentiation (Feitzinger and Lee 1997). Traditionally, products destined for world markets would be customized at the factory to suit local market tastes. While a customized product is desirable, managing worldwide inventory is often a nightmare. Using postponement, the product is redesigned so that it can be customized for local tastes in the distribution channel. The same generic product is produced at the factory and held through-out the world (Fig. 4). Thus, if the French version selling well, but the German version is not, the German product can be quickly shipped to France and customized for the French market.

For these problems, there is an interface with engineering and development, with clear implications for product cost and inventory savings. Stochastic inventory models are often used to identify some of the benefits of these initiatives (Lee et al. 1993). Also important are issues related to product design (Ulrich and Ellison 1999; Robertson and Ulrich 1998), managing product variety (Fisher et al. 1999) and managing new product introduction and product rollover (Billington et al. 1998).

The service and after sales support category addresses the critical, but often overlooked, problem of providing service and service parts (Cohen and Lee 1990). Some leading firms, such as Saturn and Caterpillar, build their reputations on their ability in this area, and this capability generates significant sales (Cohen et al. 1997). Stochastic inventory models for slow-moving items fall into this category, and there are many papers on this topic related to inventory management (Williams 1984; Cohen et al. 1986) and forecasting (Johnston and Boylan 1996). While industry practice still shows much room for improvement (Cohen et al. 1997), several well-known firms have shown how spare parts can be managed more effectively (Cohen et al. 1990; Cohen et al. 1992; Cohen et al. 1999).

Reverse logistics and green issues are emerging dimensions of supply chain management (Marien 1998). This area examines both environmental issues (Herzlinger 1994) and the reverse logistics issues of product returns (Padmanabhan and Png 1995; Clendenin 1997; Rudi and Pyke 1999). Because of legislation and consumer pressure, the growing importance of these issues is evident to most managers. Managers are being compelled to consider the most efficient and environmentally friendly way to deal with product recovery and researchers have begun significant effort in modeling these systems.

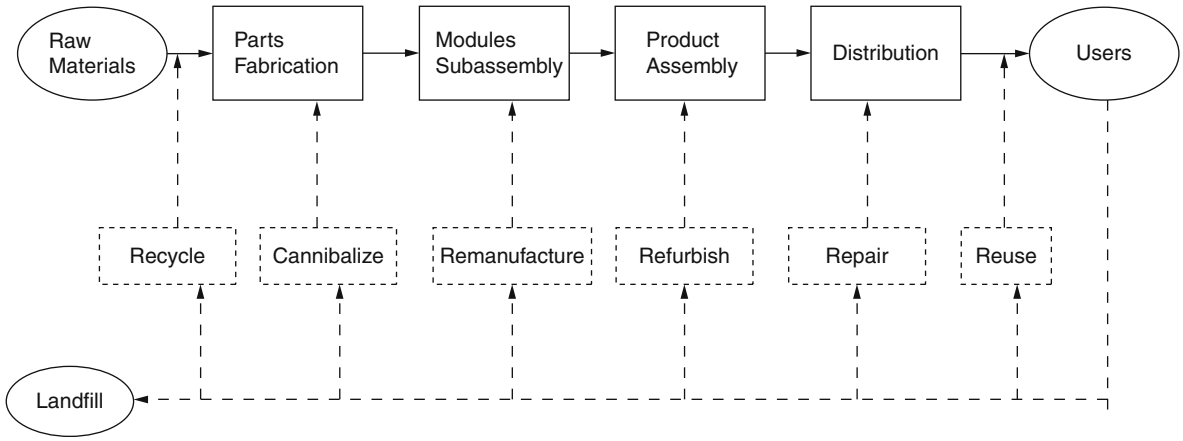
The term product recovery encompasses the handling of all used and discarded products, components and materials. Thierry et al. (1995) noted that product recovery management attempts to recover as much economic value as possible, while reducing the total amount of waste. They also provided a framework and a set of definitions that can help managers think about the issues in an organized way (Fig. 5).



Supply Chain Management, Fig. 4 Using postponement a product destined for both US and Europe markets is redesigned so that local content can be added to a common platform within distribution (adapted from Johnson and Anderson 1999)

These authors examine the differences among various product recovery options including repair, refurbishing, remanufacturing, cannibalization, and recycling. The whole process of manufacturing begins, of course, with product design. Firms are beginning to consider design for the environment (DFE) and design for disassembly (DFD) in their product development processes. Unfortunately, AT&T discovered that designing products for reuse can result in more materials and complexity, thereby violating other environmental goals. (See Frankel 1996, who also reports on product takeback and recycling initiatives in numerous countries).

The analysis of the recovery situation is considerably more complicated than that of consumables. Normally, in a recovery situation some items cannot be recovered, so the number of units demanded is not balanced completely by the return of reusable units. Thus, in addition to recovered units, a firm must also purchase some new units from time to time. Consequently, even at a single location,



Supply Chain Management, Fig. 5 Product recovery options (adapted from Thierry et al. 1995)

there are five decision variables: (1) how often to review the stock status, (2) when to recover returned units, (3) how many to recover at a time, (4) when to order new units, and (5) how many to order. When there are multiple locations, the firm must decide how many good units to deploy to a central warehouse and how many to deploy to each retailer or field stocking location.

With consumable items, the lead time to the retailers is a transportation time from the warehouse plus a random component, depending on whether the warehouse has stock. With recoverable items, the lead time is the transportation time plus the time to recovery, if the warehouse does not have stock. So in some cases, the two systems can be treated in almost the same way. However, if the recovery facility has limited capacity or if the number of items in the system is small, the systems will differ significantly. For example, if many items have failed and are now in recovery, they cannot be in the field generating failures. Therefore, the demand rate at the ware-house will decline. In a consumable system, it is usually assumed that the demand rate does not depend on how many items have been consumed.

Most of the research in this area concerns products and packaging after manufacturing has been completed. For example, a large U. S. chemical company gained significant market share in water treatment chemicals by delivering its products in reusable containers. The customers (hospitals and other large institutions, for example) need never touch the chemicals or deal with the disposal of used containers. This problem has been addressed by Goh and Varaprasad (1986), Kelle and Silver (1989), and Castillo and Cochran (1996).

Some products that are not reused "as is" can be disassembled so that some of the parts can be used in remanufactured products. Muckstadt and Isaac (1981) reported on a model developed in connection with a manufacturer of reprographic equipment. There is a single location with two types of inventory: serviceable and repairable. Demands for serviceable units and returns of repairable units occur probabilistically, specifically, according to independent Poisson processes with rates D and fD, respectively (where f is a fraction). In addition, repairs are done on a continuous, first come-first served basis (e.g., at a local machine shop). Any demands for serviceable units, when none is available, are back-ordered at a cost per unit short per unit time. Purchases of new stock from outside involve a known lead time. With respect to purchase decisions, a continuous review (s,Q) system is used; specifically, when the inventory position drops to s or lower, a quantity Q is purchased.

Inderfurth (1997) extended the Muckstadt and Isaac model to a remanufacturing problem in which there are two decisions each period: how many returned products to remanufacture (the remainder will be disposed of), and how many new parts to procure. In this system, returned products arrive probabilistically and are either remanufactured or thrown away. (In other words, there is no stock of returned products). Newly procured products are stored with

remanufactured products in a finished goods inventory that serves demand that arrives probabilistically. There are per-unit costs to procure, remanufacture, and dispose, and holding costs are charged against ending inventory each period. For the case of equal lead times to remanufacture and to procure, by Inderfurth (1997) showed that the structure of the optimal policy is based on two parameters, L_t and U_t, in each period t. To describe the policy, define the following:

d_t=	the number of units to be disposed of in period t;
p_t =	the number of units to procure in period t; and
IP_t=	the inventory position at the beginning of period t = stock on hand (which includes products returned this period and finished goods inventory) + procurement orders outstanding − remanufacturing orders out-standing − backordered demand

The optimal policy is then:

p_t =	$L_t - IP_t$ and $d_t = 0$ for a $IP_t < L_t$
p_t =	0 and $d_t = 0$ for $L_t > IP_t \leq U_t$
p_t =	0 and $d_t - IP_t - U_t$ for $IP_t > U_t$

In words, if the inventory position is lower than the lower limit, L_t, order-up-to L_t and do not dispose of any units. If the inventory position is higher than the upper limit, U_t, dispose "down to" U_t and do not procure any units. Otherwise, do not buy or dispose. (Again, all returned units, not disposed of, are remanufactured). by Inderfurth (1997) pointed out that when one permits a stock of returned units waiting for disposal or remanufacturing, or when the lead times to procure and remanufacture are different, the policy is similar but more complex.

van der Laan et al. (1996) proposed a policy for a continuous review version of this problem. Thierry et al. (1995) looked at the strategic issues related to product recovery. Also seeby Heyman (1977) and Penev and de Ron (1996), who studied the disassembly process; and van der Laan et al. (1996, 1997, 1999), by Ferrer (1995), by Richter (1996), Guide and Spencer (1997), and Taleb et al. (1997), who studied other aspects of the remanufacturing process. Other reverse logistics issues were also examined by Carter and Ellram (1998), while Fleischmann et al. (1997) provided a review of quantitative models for reverse logistics.

Outsourcing and strategic alliances examines the supply chain impact of outsourcing logistics services. With the rapid growth in third party logistics providers, there is a large and expanding group of technologies and services to be examined. These include initiatives such as supplier hubs managed by third parties. The rush to create strategic relationships with logistics providers and the many well-published failures have raised questions about the future of such relationships (Bowersox 1990). In any case, outsourcing continues to raise many interesting issues (Cooper et al. 1997a).

Metrics and incentives examines measurement and other organizational and economic issues. This category includes both measurement within the supply chain (Meyer 1997) and industry benchmarking (Council of Logistics Management Consortium 1994; Pittiglio, Rabin, Todd, and McGrath 1997). Because metrics are fundamental to business management, there are many reading materials outside of the supply chain literature, accounting texts, for instance. Several articles concentrate on the link between performance measurement and supply chain improvement (O'Laughlin 1997; Johnson and Davis 1998).

Finally, global issues examines how all of the above categories are affected when companies operate in multiple countries. This category goes beyond country specific issues to encompass issues related to crossboarder distribution and sourcing (Kouvelis 1999). For example, currency exchange rates, duties and taxes, freight forwarding, customs issues, government regulation, and country comparisons are all included. Note that the location category, when applied in a global context, also addresses some of these issues (Cohen and Huchzermeier 1999; Huchzermeier and Cohen 1996; Arntzen et al. 1995). There are several texts devoted to global management. Many articles also examine challenges in specific regions of the world (e.g., Asia: Lee and Kopczak 1997; Europe: Sharman 1997).

Concluding Remarks

Supply chain management is an expanding field, both in research and in practice. Major international consulting firms have developed large practices in the supply chain field and the number of research papers in the field is growing rapidly. The discussion covered twelve areas often seen in supply chain research and practice.

These areas appear to be somewhat disparate, but they are all linked by the integrated nature of the problems at hand. Firms operate in global environments, deal with multiple suppliers and customers, are required to manage inventories in new and innovative ways, and are faced with possible channel restructuring. The field promises to continue growing as the research advances and as firms continue to apply new knowledge in their global networks. Finally, as the Internet changes fundamental assumptions about business, firms operating in supply chains will be required to understand this new phenomenon and respond accordingly.

See

- ▶ Closed-Loop Supply Chains
- ▶ Electronic Commerce
- ▶ Facility Location
- ▶ Forecasting
- ▶ Game Theory
- ▶ Geographic Information Systems
- ▶ Inventory Modeling
- ▶ Linear Programming
- ▶ Logistics and Supply Chain Management
- ▶ Material Handling
- ▶ Network Optimization
- ▶ Operations Management
- ▶ Production Management
- ▶ Queueing Theory
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Spreadsheets
- ▶ Transportation Problem
- ▶ Traveling Salesman Problem

References

Anderson, E., Day, G. S., & Rangan, V. K. (1997). Strategic channel design. *Sloan Management Review, 38*(4), 59–69.

Anily, S., & Bramel, J. (1999). Vehicle routing and the supply chain. In S. Tayur, M. Magazine, & R. Ganeshan (Eds.), *Quantitative models for supply chain management, 147–196*. Norwell, MA: Kluwer Academic Publishers.

Arntzen, B. C., Brown, G. G., Harrison, T. P., & Trafton, L. L. (1995). Global supply chain management at digital equipment corporation. *Interfaces, 25*(1), 69–93.

Baganha, M. P., & Cohen, M. A. (1998). The stabilizing effect of inventory in supply chains. *Operations Research, 46*, S72–S83.

Ballou, R. H. (1998). *Business logistics management: Planning, organizing, and controlling the supply chain* (4th ed.). Englewood Cliffs, NJ: Prentice Hall.

Ballou, R. H., & Masters, J. M. (1999). Facility location commercial software survey. *Journal of Business Logistics, 20*, 215–232.

Bell, D. R., Ho, T., & Tang, C. (1998a). Determining where to shop: Fixed and variable costs of shopping. *Journal of Marketing Research, 35*, 352–369.

Bell, D. R., Ho, T., & Tang, C. (1998b). Rational shopping behavior and the option value of variable pricing. *Management Science, 44*, S145–S160.

Benjamin, R., & Wigand, R. (1997). Electronic markets and virtual value chain on the info super highway. *Sloan Management Review, 36*(2), 62–72.

Berry, D., Naim, M., & Towill, D. (1995). Business process re-engineering an electronic products supply chain, Paper presented at the Manufacturing Engineering Conference.

Billington, C., Lee, H. L., & Tang, C. S. (1998). Successful strategies for product rollovers. *Sloan Management Review, 39*(3), 23–30.

Blinder, A. S. (1981). Retail inventory investment and business fluctuations. *Brookings Papers on Economic Activity, 2*, 443–505.

Bodin, L. D. (1990). Twenty years of routing and scheduling. *Operations Research, 39*, 571–579.

Bourland, K., Powell, S., & Pyke, D. (1996). Exploiting timely demand information to reduce inventories. *European Journal of Operational Research, 92*, 239–253.

Bowersox, D. J. (1990). The strategic benefits of logistics alliances. *Harvard Business Review, 68*(4), 36–45.

Brush, T. H., Maritan, C. A., & Karnani, A. (1999). The plant location decision in multinational manufacturing firms: An empirical analysis of international business and manufacturing strategy perspectives. *Production and Operations Management, 8*(2), 109–132.

Business Week (1996). September 30, page 72.

Buzzell, R., & Ortmeyer, G. (1995). Channel partnerships streamline distribution. *Sloan Management Review, 36*, 85–96.

Buzzell, R., Quelch, J. A., & Salmon, W. J. (1990). The costly bargain of trade promotion. *Harvard Business Review, 68*(2), 141–149.

Cachon, G. P. (1997). "Stock wars: Inventory competition in a two echelon supply chain with multiple retailers," Unpublished Working Paper, Duke University, Durham, New Hampshire.

Cachon, G. P., & Lariviere, M. A. (1999). Capacity choice and allocation: Strategic behavior and supply chain performance. *Management Science, 45*(8), 1091–1108. forthcoming.

Carroll, P. (1993). *Big blues*. New York: Crown Publishers.

Carter, C. R., & Ellram, L. M. (1998). Reverse logistics: A review of the literature and framework for future investigation. *Journal of Business Logistics, 19*(1), 85–102.

Castillo, E. D., & Cochran, J. K. (1996). Optimal short horizon distribution operations in reusable container systems. *Journal of Operational Research Society, 47*, 48–60.

Chen, F. (1996). "The stationary beer game," Unpublished Working Paper, Columbia University, New York.

Chen, F., & Samroengraja, R. (2000). Information and incentives in supply chain management: The stationary beer games. *Production and Operations Management, 9*, 19–30, forthcoming.

Christensen, C. M. (1994). *The drivers of vertical disintegration*. Cambridge, MA: Harvard Business School.

Clark, A. J., & Scarf, H. (1960). Optimal policies for a multi-echelon inventory problem. *Management Science, 6*, 475–490.

Clendenin, J. A. (1997). Closing the supply chain loop: Reengineering the returns channel process. *Internationl Journal of Logistics Management, 8*(1), 75–85.

Cohen, M. A. & Agrawal, N. (1996). "An empirical investigation of supplier management practices," operations and information management department, Wharton School, University of Pennsylvania.

Cohen, M. A., & Huchzermeier, A. (1999). Global supply chain management: A survey of research and applications. In S. Tayur, M. Magazine, & R. Ganeshan (Eds.), *Quantitative models for supply chain management, 669–702*. Norwell, MA: Kluwer Academic Publishers.

Cohen, M. A., & Lee, H. L. (1990). Out of touch with customer needs? Spare parts and after sales service. *Sloan Management Review, 31*, 55–66.

Cohen, M., Kleindorfer, P., & Lee, H. (1986). Optimal stocking policies for Low usage items in multi-echelon inventory systems. *Naval Research Logistics, 33*, 17–38.

Cohen, M., Kamesam, P. V., Kleindorfer, P., Lee, H., & Tekerian, A. (1990). Optimizer: IBM's Multi-echelon inventory system for managing service logistics. *Interfaces, 20*(1), 65–82.

Cohen, M. A., Kleindorfer, P., & Lee, H. L. (1992). Multi-item service constrained (s, S) policies for spare parts logistics systems. *Naval Research Logistics, 39*, 561–578.

Cohen, M. A., Zheng, Y.-S., & Agrawal, V. (1997). Service parts logistics: A benchmark analysis. *IIE Transactions, 29*, 627–639.

Cohen, M. A., Zheng, Y.-S., & Wang, Y. (1999). Identifying opportunities for improving Teradyne's service-parts logistics system. *Interfaces, 29*, 1–18, forthcoming.

Cooper, M. C., Ellram, L. M., Gardner, J. T., & Hanks, A. M. (1997a). Meshing multiple alliances. *Journal of Business of Logistics, 18*(1), 67–89.

Cooper, M. C., Lambert, D. M., & Pagh, J. D. (1997b). Supply chain management: More than a new name for logistics. *Internationl Journal of Logistics Management, 8*(1), 1–14.

Copacino, W. C. (1997). *"Supply chain management: The basics and beyond*. Falls Church, VA: St Lucie Press.

Council of Logistics Management Consortium (1994). "Integrated-supply-chain performance measurement," October.

Cusumano, M. A., & Takeishi, A. (1991). Supplier relations and management: A survey of japanese, japanese-transplants, and U.S. Auto plants. *Strategic Management Journal, 12*, 563–588.

Davis, T. (1993). Effective supply chain management. *Sloan Management Review, 34*(4), 35–46.

Dornier, P., Ernst, R., Fender, M., & Kouvelis, P. (1998). *Global operations and logistics: Text and cases*. New York: John Wiley.

Drezner, Z. (1996). *Facility location: A survey of applications and methods*. New York: Springer Verlag.

Dyer, J. H. (1996). How chrysler created an american keiretsu. *Harvard Business Review, 74*, 42–56.

Feitzinger, E., & Lee, H. L. (1997). Mass customization at hewlett-packard: The power of postponement. *Havard Business Review, 75*(1), 116–121.

Ferrer, G. (1995). Parts recovery problem: The value of information in remanufacturing, INSEAD, Technology Management Area, Fontainebleau, France.

Fine, C. H. (1998). *Clock speed: Winning industry control in the age of temporary advantage*. Reading, MA: Perseus Books.

Fisher, M. L. (1997). What is the right supply chain for your product? *Harvard Business Review, 75*, 105–116.

Fisher, M. L., Hammond, J. H., Obermeyer, W. R., & Raman, A. (1994). Making supply meet demand in an uncertain world. *Harvard Business Review, 72*, 83–93.

Fisher, M., Ramdas, K., & Ulrich, K. (1999). Component sharing in the management of product variety: A study of automotive braking systems. *Management Science, 45*, 297–315.

Fites, D. V. (1996). Make your dealers your partners. *Harvard Business Review, 74*, 84–95.

Flaherty, M. T. (1996). *Global operations management*. New York: McGraw-Hill.

Fleischmann, M., Bloemhof-Ruwaard, J. M., Dekker, R., van der Laan, E., van Nunen, J. A. E. E., & Van Wassenhove, L. N. (1997). Quantitative models for reverse logistics: A review. *European Journal of Operational Research, 103*, 1–17.

Forrester, J. W. (1958). Industrial dynamics: A major breakthrough for decision makers. *Harvard Business Review, 36*, 37–66.

Forrester, J. W. (1961). *Industrial dynamics*. Cambridge, MA: Productivity Press.

Frankel, C. (1996). The environment. *IEEE Spectrum, 33*, 76–81.

Fuller, J. B., O'Conor, J., & Rawlinson, R. (1993). Tailored logistics: The next advantage. *Harvard Business Review, 71*, 87–93.

Ganeshan, R., Jack, E., Magazine, M., & Stephens, P. (1999). A taxonomic review of supply chain management research. In S. Tayur, M. Magazine, & R. Ganeshan (Eds.), *Quantitative models for supply chain management, 839–879*. Norwell, MA: Kluwer Academic Publishers.

Gendreau, M., Laport, G., & Seguin, R. (1996). Stochastic vehicle routing. *European Journal of Operational Research, 88*, 3–12.

Gilmore, J. H., & Pine, B. J. (1997). The four faces of mass customization. *Harvard Business Review, 75*, 91–101.

Goh, T. N., & Varaprasad, N. (1986). A statistical methodology for the analysis of the life–cycle of reusable containers. *IIE Transactions, 18*(1), 42–47.

Graves, S., Rinnooy Kan, A., & Zipkin, P. (Eds.). (1993). *Logistics of production and inventory* (Vol. 4). North-Holland/Amsterdam: Elsevier.

Greis, N. P., & Kasarda, J. D. (1997). Enterprise logistics in the information era. *California Management Review, 39*(3), 55–78.

Guide, V. D. R., & Spencer, M. S. (1997). Rough-cut capacity planning for remanufacturing firms. *Production Planning and Control, 8*, 237–244.

Hammond, J. H. (1994). "Barilla SpA (A)," Case Number 9-694-046, Harvard Business School, Cambridge, Massachusetts.

Hammond, J. H. & Kelly, M. (1990). Note on facility location, Unpublished Note, Harvard University, Cambridge, Massachusetts.

Handfield, R. B., & Nichols, E. Z. (1998). *Introduction to supply chain management*. Englewood Cliffs, NJ: Prentice-Hall.

Helper, S., & Sako, M. (1995). Supplier relations in japan and the united states: Are they converging? *Sloan Management Review, 36*, 77–84.

Herzlinger, R. (1994). The challenges of going green. *Harvard Business Review, 61*, 37–50.

Heyman, D. P. (1977). Optimal disposal policies for a single-item inventory system with returns. *Naval Research Logistics, 24*, 385–405.

Huchzermeier, A., & Cohen, M. A. (1996). Valuing operational flexibility under exchange rate risk. *Operations Research, 44*, 100–113.

Inderfurth, K. (1997). Simple optimal replenishment and disposal policies for a product recovery system with leadtimes. *OR-Spektrum, 19*, 111–122.

Jacobs, R. (2000). Playing the beer distribution game over the internet. *Production and Operations Management, 9*(1), 31–39, forthcoming.

Johnson, M. E. (1998a). Give them what they want. *Management Review, 3*, 62–67.

Johnson, M. E. (1998b). The impact of sorting strategies on automated sortation system performance. *IIE Transactions, 30*(1), 67–77.

Johnson, M. E. & Anderson, E. (1999). "The value of postponement in channel management," Unpublished Working Paper, Vanderbilt University, Nashville, Tennessee.

Johnson, M. E., & Brandeau, M. L. (1999). Design of an automated shop floor material handling system. *Operations Research, 47*, 65–80.

Johnson, M. E., & Davis, T. (1998). Improving supply chain performance using order fulfillment metrics. *National Productivity Review, 17*, 3–16.

Johnson, M. E., & Pyke, D. F. (2000a). A framework for teaching supply chain management. *Production and Operations Management, 9*, 2–18, forthcoming.

Johnson, M. E. & Pyke, D. F., (eds.), (2000b). Teaching supply chain management, production and operations management society.

Johnston, F. R., & Boylan, J. E. (1996). Forecasting for items with intermittent demand. *Journal of Operational Research Society, 47*, 113–121.

Kelle, P., & Silver, E. A. (1989). Purchasing policy of New containers considering the random returns of previously issued containers. *IIE Transactions, 21*, 349–354.

Kelley, B. (1995). Outsourcing marches on. *Journal of Business Strategy, 16*, 39–42.

Kopczak, L., Lee, H., & Whang, S. (1995). "Note on logistics," Unpublished Note, Stanford University, Stanford, California.

Kouvelis, P. (1999). Global sourcing strategies under exchange rate uncertainty. In S. Tayur, M. Magazine, & R. Ganeshan (Eds.), *Quantitative models for supply chain management, 625–668*. Norwell, MA: Kluwer Academic Publishers.

Lambert, D. M., Stock, J. R., Ellram, L. M., & Stockdale, J. (1997). *Fundamentals of logistics management*. New York: McGraw Hill.

Lee, H., & Billington, C. (1992). Managing supply chain inventories: Pitfalls and opportunities. *Sloan Management Review, 33*, 65–73.

Lee, H. & Kopczak, L. (1997). Responding to the Asia-Pacific challenge, *Supply Chain Management Review*, Spring, 8–9.

Lee, H. L., & Nahmias, S. (1993). Single-product, single-location models. In S. Graves, A. Rinnooy Kan, & P. Zipkin (Eds.), *Logistics of production and inventory* (Vol. 4). North-Holland/Amsterdam: Elsevier. Chapter 1.

Lee, H., & Whang, S. (1999a). Decentralized multiechelon supply chains: Incentives and information. *Management Science, 45*, 633–640.

Lee, H.L. & Whang, S. (1999b). Information sharing in a supply chain, *International Journal of Technology Management*, forthcoming.

Lee, H., Billington, C., & Carter, B. (1993). Hewlett-packard gains control of inventory and service through design for localization. *Interfaces, 23*(4), 1–11.

Lee, H., Padmanabhan, P., & Whang, S. (1997). The bullwhip effect in supply chains. *Sloan Management Review, 38*(3), 93–102.

Lee, H. L., So, K. C., & Tang, C. S. (1999). The value of information sharing in a two-level supply chain. *Management Science, 46*, 626–643.

Little, A. D. & Associates (1995). "An exchange of knowledge among leading practitioners in supply chain management," Unpublished Note, Boston, Massachusetts.

Magretta, J. (1998). The power of virtual integration: An interview with dell Computer's michael dell. *Havard Business Review, 76*(2), 72–84.

Marien, E. J. (1998). Reverse logistics as competitive strategy. *Supply Chain Management Review, 2*, 43–52.

McCutcheon, D. M., Raturi, A. S., & Meredith, J. R. (1994). The customization-responsiveness squeeze. *Sloan Management Review, 35*(2), 89–99.

McMillan, J. (1990). Managing suppliers: Incentive systems in japanese and U.S. Industry. *California Management Review, 32*(4), 38–55.

Meyer, M. (1997). "The performance imperative," Unpublished Working Paper, University of Pennsylvania, Philadelphia.

Milgrom, P., & Roberts, J. (1988). Communication and inventory as substitutes in organizing production. *Scandinavian Journal of Economics, 90*, 275–289.

Moinzadeh, K., & Aggarwal, P. K. (1997). An information based multi-echelon inventory system with emergency orders. *Operations Research, 45*, 694–701.

Muckstadt, J. A., & Isaac, M. H. (1981). An analysis of single item inventory systems with returns. *Naval Research Logistics, 28*, 237–254.

Narus, J. A., & Anderson, J. C. (1996). Rethinking distribution. *Harvard Business Review, 74*(4), 112–120.

O'Laughlin, K. A. (1997). Five steps to improved performance measurement, *Supply Chain Management Review, Fall*, 52–58.

Padmanabhan, V., & Png, I. P. L. (1995). Return policies: Make money by making good. *Sloan Management Review (Fall), 37*, 65–72.

Penev, K. D., & de Ron, A. J. (1996). Determination of a disassembly strategy. *International Journal of Production Research, 34*, 495–506.
Pittiglio Rabin Todd & McGrath (1997). "The keys to unlocking your supply chain's competitive advantage," Mountain View, California.
Pyke, D. F. (1994). Global sourcing at a second glance. *Global Competitor, 1*(3), 70–74.
Pyke, D. F. (1998). Strategies for global sourcing. *Financial Times, 20*, 2–4.
Quinn, J. B., & Hilmer, F. (1994). Strategic out-sourcing. *Sloan Management Review, 35*, 43–55.
Richter, K. (1996). The EOQ repair and waste disposal model with variable setup numbers. *European Journal of Operational Research, 95*, 313–324.
Robertson, P. L., & Langlois, R. N. (1995). Innovation, networks, and vertical integration. *Research Policy, 24*, 543–562.
Robertson, D., & Ulrich, K. (1998). Planning for product platforms. *Sloan Management Review, 39*, 19–31.
Rudi, N. & Pyke, D. F. (1999). "Product recovery at the Norwegian Health Insurance Administration," Interfaces, forthcoming.
Schonfeld, E. (1998). The customized, digitized, have-it-your-way economy. *Fortune, 138*, 115–124.
Senge, P. (1990). *The fifth discipline*. New York: Doubleday.
Sharman, G. J. (1997). Supply chain lessons from Europe. *Supply Chain Management Review Fall, 1*, 11–13.
Silver, E. A., Pyke, D. F., & Peterson, R. (1998). *Inventory management and production planning and scheduling* (3rd ed.). New York: John Wiley.
Simchi-Levi, D., Kaminsky, P., & Simchi-Levi, E. (1998). *Designing and managing the supply chain*. New York: Irwin/McGraw-Hill.
Srinivasan, K., Kekre, S., & Mukhopadhyay, T. (1994). Impact of electronic data interchange technology on JIT shipments. *Management Science, 40*, 1291–1304.
Sterman, J. D. (1989). Modeling managerial behavior: Misperceptions of feedback in a dynamic decision making experiment. *Management Science, 35*, 321–339.
Sterman, J. D. (1992). Teaching takes off: Flight simulators for management education. *OR/MS Today, 19*(5), 40–43.
Taleb, K. N., Gupta, S. M., & Brennan, L. (1997). Disassembly of complex product structures with parts and materials commonality. *Production Planning and Control, 8*, 255–269.
Taylor, D. (1997). *Global cases in logistics and supply chain management*. New York: International Thomson Business Press.
Tayur, S., Magazine, M., & Ganeshan, R. (Eds.). (1999). *Quantitative models for supply chain management*. Norwell, MA: Kluwer Academic Publishers.
Thierry, M., Salomon, M., Van Nunen, J., & Van Wassenhove, L. V. (1995). Strategic issues in product recovery management. *California Management Review, 37*(2), 114–135.
Towill, D., & Vecchio, D. (1994). The application of filter theory to the study of supply chain dynamics. *Production Planning and Control, 5*, 82–96.
Train, J. (1998). "Legal issues affecting distribution and supply," Unpublished Working Paper, Duke University, Durham, North Carolina.
Tsay, A. A., Hahmias, S., & Agrawal, N. (1999). Modeling supply chain contracts: A review. In S. Tayur, M. Magazine, & R. Ganeshan (Eds.), *Quantitative models for supply chain management, 299–336*. Norwell, MA: Kluwer Academic Publishers.
Ulrich, K., & Ellison, D. (1999). Holistic customer requirements and the design-select decision. *Management Science, 45*, 641–655.
van der Laan, E., Dekker, R., Salomon, M., & Ridder, A. (1996). An (s, Q) inventory model with remanufacturing and disposal. *International Journal of Production Economics, 46–47*, 339–350.
van der Laan, E., Salomon, M., & Dekker, R. (1997). Production planning and inventory control for remanufacturable durable products. *European Journal of Operational Research, 102*, 264–278.
van der Laan, E., Salomon, M., & Dekker, R. (1999). An investigation of lead-time effects in manufacturing/remanufacturing systems under simple PUSH and PULL control strategies. *European Journal of Operational Research, 115*, 195–214.
Venkatesan, R. (1992). Strategic sourcing: To make or not to make. *Harvard Business Review, 70*, 98–107.
Verity, J. W. (1996). Clearing the cobwebs from the stockroom. *Business Week, 21*, 140.
Waller, M., Johnson, M. E., & Davis, T. (1999). Vendor-managed inventory in the retail supply chain. *Journal of Business Logistics, 20*(1), 183–203.
Whang, S., Gilland, W., & Lee, H. (1995). "Information flows in manufacturing under SAP R/3," Unpublished Working Paper Stanford University, Stanford, California.
Williams, T. M. (1984). Stock control with sporadic and slow-moving demand. *Journal of Operational Research Society, 35*, 939–948.
Womack, J. P., Jones, D. T., & Roos, D. (1991). *The machine that changed the world: The story of lean production*. New York: Harper Perennial.
Woolley, S. (1997). Replacing inventory with information. *Forbes, 24*, 54–58.

Surplus Variable

A nonnegative variable that is added to a linear inequality of the form $\sum_j a_{ij} x_j \geq b_i$ to transform the inequality into an equation. The surplus variable measures the difference between the left-and right-hand-sides of the inequality.

See

- ▶ Logical Variables
- ▶ Slack Variable
- ▶ Surplus Vector

Surplus Vector

The column representation of a surplus variable in a linear-programming problem.

See

► Surplus Variable

Swarm Intelligence

Population-based metaheuristic search approaches that use groups of decentralized agents inspired by animal behavior from nature; examples include Ant Colony Optimization, Particle Swarm Optimization, and the Bees Algorithm.

See

► Ant Colony Optimization
► Metaheuristics
► Particle Swarm Optimization

References

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm intelligence: From natural to artificial systems*. New York: Oxford University Press.
Kennedy, J. F., Eberhart, R. C., & Shi, Y. (2001). *Swarm intelligence*. San Francisco: Academic Press/Morgan Kauffman Publishers.

Symmetric Matrix

A square matrix $\boldsymbol{A} = (a_{ij})$ is symmetric if $a_{ij} = a_{ji}$. Thus, $\boldsymbol{A} = \boldsymbol{A}^T$.

See

► Matrices and Matrix Algebra

Symmetric Primal-Dual Problems

The two linear-programming problems with the following form:

Primal

$$\begin{aligned} &\text{Minimize} && \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \\ &\text{subject to} && \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b} \\ & && \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

Dual

$$\begin{aligned} &\text{Maximize} && \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} \\ &\text{subject to} && \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} \leq \boldsymbol{c} \\ & && \boldsymbol{y} \geq \boldsymbol{0} \end{aligned}$$

See

► Strong Duality Theorem
► Unsymmetric Primal-Dual Problems

Symmetric Queueing Network

A queueing network of quasi-reversible nodes (stations) with additional properties that make its major performance measures (e.g., waiting times and queue lengths) insensitive to the service-time distributions, depending only on the mean service times.

See

► Insensitivity
► Networks of Queues
► Queueing Theory

Symmetric Zero-Sum Two-Person Game

A two-player game with a skew-symmetric payoff matrix. The amount lost by one player is the amount gained by the other player (zero-sum). Such a game has a value of zero and the optimal strategies of the two players are the same.

See

- ▶ Game Theory
- ▶ Skew-symmetric Matrix

System

A set of related elements organized to achieve a purpose.

See

- ▶ Systems Analysis

System Dynamics

George P. Richardson
University at Albany, State University of New York, Albany, NY, USA

Introduction

System dynamics is a computer-aided approach to policy analysis and design. It applies to dynamic problems arising in complex social, managerial, economic, or ecological systems – literally any dynamic systems characterized by interdependence, mutual interaction, information feedback, and circular causality.

The field developed initially from the work of Jay W. Forrester. His seminal book *Industrial Dynamics* (Forrester 1961) is still a significant statement of philosophy and methodology in the field. Within 10 years of its publication, the span of applications grew from corporate and industrial problems to include the management of research and development, urban stagnation and decay, commodity cycles, and the dynamics of growth in a finite world. It is now applied in economics, public policy, environmental studies, defense, theory-building in social science, and other areas, as well as its home field, management. The name industrial dynamics no longer does justice to the breadth of the field, so it has become generalized to system dynamics. The modern name suggests links to other systems methodologies, but the links are weak and misleading. System dynamics emerges out of servomechanisms engineering, not general systems theory or cybernetics (Richardson 1991).

The system dynamics approach involves:

- Defining problems dynamically, in terms of graphs over time.
- Striving for an endogenous, behavioral view of the significant dynamics of a system, a focus inward on the characteristics of a system that themselves generate or exacerbate the perceived problem.
- Thinking of all concepts in the real system as continuous quantities interconnected in loops of information feedback and circular causality.
- Identifying independent stocks or accumulations (levels) in the system and their inflows and outflows (rates).
- Formulating a behavioral model capable of reproducing, by itself, the dynamic problem of concern. The model is usually a computer simulation model expressed in nonlinear equations, but is occasionally left unquantified as a diagram capturing the stock-and-flow/causal feedback structure of the system.
- Deriving understandings and applicable policy insights from the resulting model.
- Implementing changes resulting from model-based understandings and insights.

Mathematically, the basic structure of a formal system dynamics computer simulation model is a system of coupled, nonlinear, first-order differential (or integral) equations,

$$\frac{d}{dt}x(t) = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{p}),$$

where x is a vector of levels (stocks or state variables), $\boldsymbol{p}$ is a set of parameters, and $\boldsymbol{f}$ is a nonlinear vector-valued function.

Simulation of such systems is easily accomplished by partitioning simulated time into discrete intervals of length dt and stepping the system through time one dt at a time. Each state variable is computed from its previous value and its net rate of change $x'(t) : x(t) = x(t - dt) + dt * x'(t - dt)$. In the earliest

simulation language in the field (DYNAMO) this equation was written with time scripts K (the current moment), J (the previous moment), and JK (the interval between time J and K): X.K = X.J + DT*XRATE.JK (see, e.g., Richardson and Pugh 1981). The computation interval *dt* is selected small enough to have no discernible effect on the patterns of dynamic behavior exhibited by the model. In more recent simulation environments, more sophisticated integration schemes are available (although the equation written by the user may look like this simple Euler integration scheme), and time scripts may not be in evidence. Important current simulation environments include Vensim (Ventana Systems, www.vensim.com/) STELLA and iThink (iSee Systems, www.iseesystems.com/), PowerSim (www.powersim.com/), and AnyLogic (xj technologies, www.xjtek.com/).

Forrester's original work stressed a continuous approach, but increasingly modern applications of system dynamics contain a mix of discrete difference equations and continuous differential or integral equations. Some practitioners associated with the field of system dynamics work on the mathematics of such structures, including the theory and mechanics of computer simulation, analysis and simplification of dynamic systems, policy optimization, dynamical systems theory, and complex nonlinear dynamics and deterministic chaos.

The main applied work in the field, however, focuses on understanding the dynamics of complex systems for the purpose of policy analysis and design. The conceptual tools and concepts of the field – including feedback thinking, stocks and flows, the concept of feedback loop dominance, and an endogenous point of view – are as important to the field as its simulation methods. The material in the next three sections is abstracted from Richardson (1991a, b).

Feedback Thinking

Conceptually, the feedback concept is at the heart of the system dynamics approach. Diagrams of loops of information feedback and circular causality are tools for conceptualizing the structure of a complex system and for communicating model-based insights. Intuitively, a feedback loop exists when information resulting from some action travels through a system and eventually returns in some form to its point of origin, potentially influencing future action. If the tendency in the loop is to reinforce the initial action, the loop is called a positive or reinforcing feedback loop; if the tendency is to oppose the initial action, the loop is called a negative or balancing feedback loop. The sign of the loop is called its polarity. Balancing loops can be variously characterized as goal-seeking, equilibrating, or stabilizing processes. They can sometimes generate oscillations, as when a pendulum seeking its equilibrium goal gathers momentum and overshoots it. Reinforcing loops are sources of growth or accelerating collapse; they are disequilibrating and destabilizing. Combined, reinforcing and balancing circular causal feedback processes can generate all manner of dynamic patterns.

Loop Dominance and Nonlinearity

The loop concept underlying feedback and circular causality by itself is not enough, however. The explanatory power and insightfulness of feedback understandings also rest on the notions of active structure and loop dominance. Complex systems change over time. A crucial requirement for a powerful view of a dynamic system is the ability of a mental or formal model to change the strengths of influences as conditions change, that is to say, the ability to shift active or dominant structure.

In a system of equations, this ability to shift loop dominance comes about endogenously from nonlinearities in the system. For example, the S-shaped dynamic behavior of the classic logistic growth model $\left(dP/dt = aP - bP^2\right)$ can be seen as the consequence of a shift in loop dominance from a positive, self-reinforcing feedback loop (aP) producing exponential-like growth to a negative balancing feedback loop ($-bP^2$) that brings the system to its eventual goal. Only nonlinear models can endogenously alter their active or dominant structure and shift loop dominance. From a feedback perspective, the ability of nonlinearities to generate shifts in loop dominance and capture the shifting nature of reality is the fundamental reason for advocating nonlinear models of social system behavior.

The Endogenous Point of View

The concept of endogenous change is fundamental to the system dynamics approach. It dictates aspects of model formulation: exogenous disturbances are seen at most as triggers of system behavior (like displacing a pendulum); the causes are contained within the structure of the system itself (like the interaction of a pendulum's position and momentum that produces oscillations). Corrective responses are also not modeled as functions of time, but are dependent on conditions within the system. Time by itself is not seen as a cause.

But more importantly, theory building and policy analysis are significantly affected by this endogenous perspective. Taking an endogenous view exposes the natural compensating tendencies in social systems that conspire to defeat many policy initiatives. Feedback and circular causality are delayed, devious, and deceptive. For understanding, system dynamics practitioners strive for an endogenous point of view. The effort is to uncover the sources of system behavior that exist within the structure of the system itself.

System Structure

These ideas are captured in Forrester's (1969) organizing framework for system structure:

Closed boundary

- Feedback loops
  - Levels
  - Rates
    - Goal
    - Observed condition
    - Discrepancy
    - Desired action

The closed boundary signals the endogenous point of view. The word closed here does not refer to open and closed systems in the general system sense, but rather refers to the effort to view a system as causally closed. The modeler's goal is to assemble a formal structure that can, by itself, without exogenous explanations, reproduce the essential characteristics of a dynamic problem.

The causally closed system boundary at the head of this organizing framework identifies the endogenous point of view as the feedback view pressed to an extreme. Feedback thinking can be seen as a consequence of the effort to capture dynamics within a closed causal boundary. Without causal loops, all variables must trace the sources of their variation ultimately outside a system. Assuming instead that the causes of all significant behavior in the system are contained within some closed causal boundary forces causal influences to feed back upon themselves, forming causal loops. Feedback loops enable the endogenous point of view and give it structure.

Levels and Rates

Stocks (levels) and the flows (rates) that affect them are essential components of system structure. A map of causal influences and feedback loops is not enough to determine the dynamic behavior of a system. A constant inflow yields a linearly rising stock; a linearly rising inflow yields a stock rising along a parabolic path, and so on. Stocks (accumulations, state variables) are the memory of a dynamic system and are the sources of its disequilibrium and dynamic behavior.

Forrester (1961) placed the operating policies of a system among its rates (flows), many of which assume the classic structure of a balancing feedback loop striving to take action to reduce the discrepancy between the observed condition of the system and a goal. The simplest such rate structure results in an equation of the form NETFLOW = (GOAL - STOCK)/(ADJTIM), where ADJTIM is the time over which the level adjusts to reach the goal.

Behavior Is a Consequence of System Structure

The importance of levels and rates appears most clearly when one takes a continuous view of structure and dynamics. Although a discrete view, focusing on separate events and decisions, is entirely compatible with an endogenous feedback perspective, the system dynamics approach emphasizes a continuous view. The continuous view strives to look beyond events to see the dynamic patterns underlying them. Moreover,

the continuous view focuses not on discrete decisions but on the policy structure underlying decisions. Events and decisions are seen as surface phenomena that ride on an underlying tide of system structure and behavior. It is that underlying tide of policy structure and continuous behavior that is the system dynamicist's focus.

Therefore, there is a distancing inherent in the system dynamics approach – not so close as to be confused by discrete decisions and myriad operational details, but not so far away as to miss the critical elements of policy structure and behavior. Events are deliberately blurred into dynamic behavior. Decisions are deliberately blurred into perceived policy structures. Insights into the connections between system structure and dynamic behavior, which are the goal of the system dynamics approach, come from this particular distance of perspective.

Concluding Remarks

System Dynamics Review, the journal of the System Dynamics Society, is the best source of current activity in the field, including methodological advances and applications. Other important journal sources include *Management Science*, the *European Journal of Operational Research* (EJOR), the *Journal of the Operational Research Society* (JORS), and *Systems Research and Behavioral Science*. For texts on the modeling process in system dynamics, see Sterman (2000), Maani and Cavana (2007), Ford (2009), Morecroft (2007), Wolstenholme (1990), and Richardson and Pugh (1981).

An early interesting collection of applications is Roberts (1978); Richardson (1996) is a more recent two-volume edited collection in the same spirit, containing prize-winning work in philosophical background, dynamic decision making, applications in the private and public sectors, and techniques for modeling with management.

One direction within the field is the use of model-based insights for organizational learning, represented most forcefully in Senge (1990) and Morecroft and Sterman (1994). The important effort to build models with relatively large groups of experts and stakeholders, known as group model building, is described in Vennix (1996) and Richardson and Anderxsen (2010).

Richardson (1991/1999) puts the endogenous feedback perspective of the system dynamics approach in its historical context and includes an extensive bibliography.

References

Ford, A. (2009). *Modeling the environment*. Washington, DC: Island Press.

Forrester, J. W. (1961). *Industrial dynamics*. Cambridge, MA: The MIT Press. (Reprinted by Pegasus Communications, Waltham, MA)

Forrester, J. W. (1969). *Urban dynamics*. Cambridge, MA: The MIT Press. (Reprinted by Pegasus Communications, Waltham, MA)

Maani, K. E., & Cavana, R. Y. (2007). *Systems thinking, system dynamics: Understanding change and complexity*. Auckland: Prentice Hall.

Morecroft, J. D. W. (2007). *Strategic modeling and business dynamics: A feedback systems approach*. Chichester: Wiley.

Morecroft, J. D. W., & Sterman, J. D. (Eds.). (1994). *Modeling for learning organizations. System dynamics series*. Cambridge, MA: Pegasus Communications.

Richardson, G. P. (1991/1999). *Feedback thought in social science and systems theory*. Philadelphia: University of Pennsylvania Press. (Reprinted by Pegasus Communications, Waltham, MA)

Richardson, G. P. (Ed.). (1996). *Modelling for management: Simulation in support of systems thinking* (International library of management). Aldershot: Dartmouth Publishing Company.

Richardson, G. P., & Andersen, D. F. (2010). Systems thinking, mapping, and modeling for group decision and negotiation. In C. Eden & D. N. Kilgour (Eds.), *Handbook for group decision and negotiation* (pp. 313–324). Dordrecht: Springer.

Richardson, G. P., & Pugh III, A. L. (1981). *Introduction to system dynamics modeling with DYNAMO*. Cambridge, MA: The MIT Press. (Reprinted by Pegasus Communications, Waltham, MA)

Roberts, E. B. (Ed.). (1978). *Managerial applications of system dynamics*. Cambridge, MA: The MIT Press. (Reprinted by Pegasus Communications, Waltham, MA)

Senge, P. M. (1990). *The fifth discipline: The art and practice of the learning organization*. New York: Doubleday/Currency.

Sterman, J. D. (2000). *Business dynamics: Systems thinking and modeling for a complex world*. Boston: Irwin McGraw-Hill.

System dynamics review. (1985–present). Chichester: Wiley-Blackwell.

Vennix, J. A. M. (1996). *Group model building: Facilitating team learning using system dynamics*. Chichester: Wiley.

Wolstenholme, E. F. (1990). *System enquiry: A system dynamics approach*. Chichester: Wiley.

System Reliability

► Reliability of Stochastic Systems

Systems Analysis

Sue A. Conger[1] and Richard O. Mason[2]
[1]University of Dallas, Irving, TX, USA
[2]Southern Methodist University, Dallas, TX, USA

Introduction

Systems analysis is a broad term applied to the study of real-world processes. It involves the careful examination of systems – entities, organisms, organizations, beings and things. Systems analysis breaks problems associated with entities down into their component parts and relationships in order to formulate a conceptual definition of the situation. The purpose is to develop "an overall understanding of optimal solutions to executive type problems" (Churchman et al. 1957, p. 7). The resulting conceptual definition is then often translated into a Web site, a process-support system, or a mathematical model. Systems analysis has been applied to complex, dynamic systems — both physical and social — such as businesses, governments, and computer software, as well as to economic, weapons, mechanical, and manufacturing systems. While it is ultimately a subjective form of innovation, systems analysis is based on a growing set of key theories: systems, cybernetics, mathematical modeling, graphical design, data management for knowledge management, and computational linguistics.

Systems theory describes how related elements can be organized to achieve a purpose. Elements form "an interconnected complex of functionally related components" (Churchman et al. 1957, p. 7), each having inputs, processes, and outputs. At the most detailed and fundamental level of analysis, elements are generally treated as 'black boxes.' At a high level of abstraction, what goes into and out of each black box is described, but the activities within the box are not described. Each black box is analyzed in turn to define the transformation process through which its inputs generate its outputs. The concepts of flow, relationship, message, initiator, terminator, and connection are used to portray the structure of the system being analyzed. These terms describe the interrelationships of its elements. The transformation processes are described in terms including transaction, process, and problem.

Cybernetic theory integrates feedback in systems. Feedback provides communication about the system's outputs, which in turn causes the system to adjust either the inputs or the process, as necessary, to achieve the system's purpose. This is called control. Mathematical system theories define a "collection of mathematical relationships which characterize the feasible programs" for improving a system (Dantzig 1963). Building a mathematical model provides insight into a system and its properties, and the model elements can be manipulated to derive conclusions about the system.

Mathematical models and other operations research/management science (OR/MS) techniques may be applied to the conceptual definition of a system and used to determine the best possible solution — the optimum decision, policy, or design — for the problem the system represents.

Graphs and graph theory are the basis for systems analysis relating to Web site and page design for target business processes. A graph is a collection of points and lines connecting these points used to represent relations between sets of objects. Graph theory is used to study some of the many possible properties of the identified objects (Berge 1962). Graphical theory is applied to business process information to develop abstract presentation forms using everyday metaphors and otherwise meaningful renditions. Then, the abstractions are translated into individual Web components that are compiled into Web pages, which together comprise a Web site.

Theories on data management and knowledge management provide a basis for data warehousing and retrieval systems (Alavi and Leidner 2001; Jarke and Vassiliou 1997). Like graphical theory, the emphasis of these theories is on business practice and use of system artifacts in addition to efficient uses of technology resources and application functionality. As a result, the expanded skills needed to analyze data technology and its use add to the techniques for systems analysis.

Churchman (1968) posed five necessary conditions for completing any systems analysis:

1. The total system objectives or, more specifically, the performance measures for the whole system;
2. The system's environment of fixed constraints, which are outside the system;

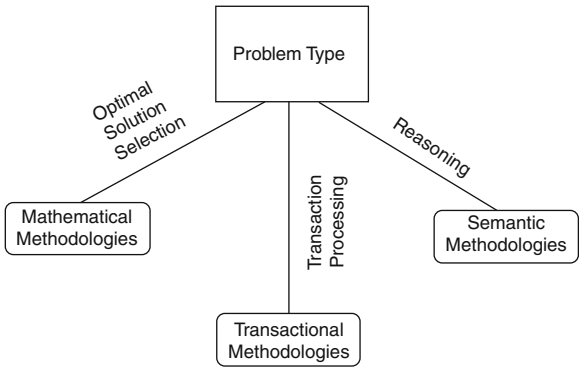
3. System resources that are capabilities found "inside" the system and that, therefore, can be analyzed and possibly designed;
4. System elements, including their activities, functions, goals, and measures of performance;
5. System management processes for allocating resources to system elements.

Most systems contain recognizable sub-systems, sub-sub-systems, and so on, organized in a hierarchy. Their arrangement often involves a "Chinese box" form of nesting that permits them to be defined by recursion. A solution that is best for the system as a whole is called an optimum, whereas a solution that is best relative to the functioning of one or more elements is called a sub-optimum or local optimum. One of the challenges of systems analysis is to improve the performance of a sub-system in terms of its own goals and purposes — sub-optimization — without either harming the total system or, worse, defeating the system's overall purpose.

Systems Analysis and Computer Systems

Systems analysis is the first stage in presenting any large task to a computer (the other principle stages being design and implementation). It is performed by a systems analyst and consists of analyzing the whole task in its setting and deciding how best to arrange it for processing by a computer. It includes estimation of how much work is involved, how powerful a computer is required, and the quality of the operating environment for security, recoverability, and reliable availability of computing resources. A problem is divided into a number of relatively independent parts that are specified, together with their interconnections, in sufficient detail for a programmer to take over. Options for arrangement of problem components can be hardware, firmware, or software; in hierarchies, sequences, or networks; for local, remote, or virtual computing.

Computer applications are developed through a series of translations. The first translation, as noted above, is from a real-world situation to a conceptual definition of the situation. This conceptual model is thentranslated via a design activity to an implementation model that can still be read by human beings and that describes the conceptual model in a language related to the target computer environment.



Systems Analysis, Fig. 1 Methodology classes

The implementation model is then translated into the specific coded language(s) of the target (hardware, software, firmware, and data) environment. These three translations define phases of activity that constitute an application's development life cycle. The translations relate to the thinking processes involved and are called analysis, design, and implementation, respectively. Implementation can be divided into sub-phases for programming, testing, and production.

Software development methodologies are used to guide the development processes through the life cycle. (Technically, methodology is the study of tools, techniques, and guidelines for choosing among them; methods are specific tools and techniques to be chosen and applied to a given situation. The common term for system development methods used as a package of tools and techniques is methodologies, and it is used here.) The different approaches currently used are: mathematical, process, data, object, information, and artificial intelligence. The techniques typically use top-down strategies for problem solving and progressively decompose a target task area into smaller, solvable tasks for independent solution (Laszlo 1972); however, bottom-up and middle-out strategies can also be applied to aspects of problems. The approaches can be further divided into classes: mathematical, transactional, semantic, and informational — depending on the type of problem being solved (Fig. 1). Mathematical methodologies solve selection and alternative analysis problems. Process, data, and object methodologies solve transactional processing problems. Information methodologies solve data storage, retrieval, and presentation problems. Semantic methodologies, in general, deal with understanding complex information

and artificial intelligence problems. Applications often encompass several of these problem types, requiring hybrid approaches to their solution.

Mathematical Methodologies

Mathematical methodologies employ mathematical models of a system and focus on the logical relationships within the system. They are often formulated by interdisciplinary teams who adapt scientific theories and methods to solve practical problems (Ackoff and Rivett 1963). OR/MS and cybernetic methods are applied. Classes of problems to which mathematical methods apply include inventory, allocation, sequencing, queueing, routing, replacement, competition, and search. The problems solved by OR techniques all deal with selection from many alternatives and sensitivity analysis to develop alternative, robust, or contingent courses of action.

Mathematical cybernetic systems seek optimal solutions based on unambiguous but possibly incomplete information. The inputs to mathematical applications define the alternatives and resources available from which an optimal selection must be made. The tools and techniques used to develop mathematical models include linear, network, dynamic, and stochastic programming methods. The results of these applications are usually presented in the form of suggested machine schedules, resource allocations, and so on. These problems, while the original focus of computing in the 1950s, have mostly been reduced to software packages and are rarely developed as custom software. As a result, application of mathematical methodologies has become a scarce skill.

Transactional Methodologies

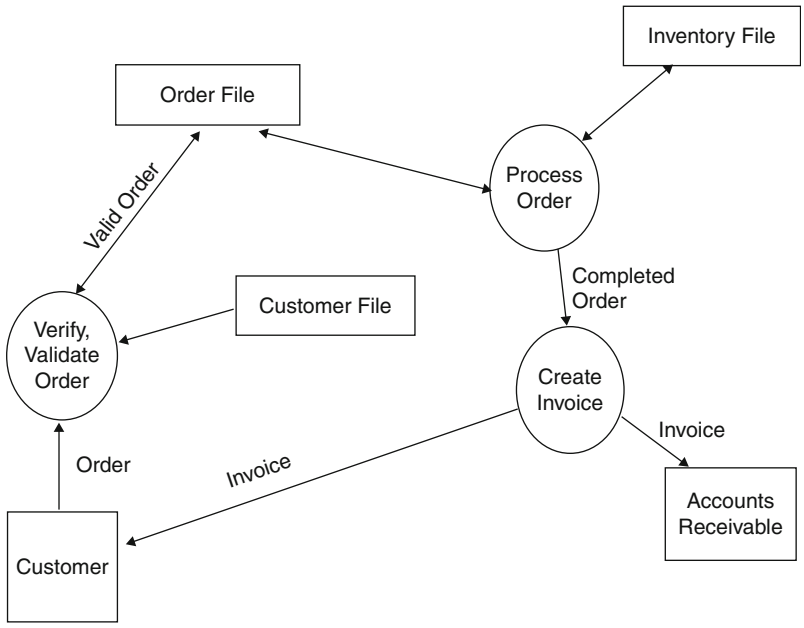
Transactional methods focus on the flows of information between the elements of a system. Three different methodology classes have evolved to develop transactional, information retrieval, and data analysis applications: process, data, and object. No single methodology currently supports all three application types well. Further, as the demand for client/server systems and distributed systems evolves, improved methodologies have evolved to support their development.

Process Methodologies — Process methods were developed during the 1950s and 1960s to mirror von Neumann computer architecture, which separates inputs and outputs from processes. Since computing was the difficult issue at the time, processing was the initial focus of process methods. The types of problems automated included accounting procedures, order entry, inventory, and other back-office applications. These applications all deal with transactions that support the basic white-collar operations of an organization.

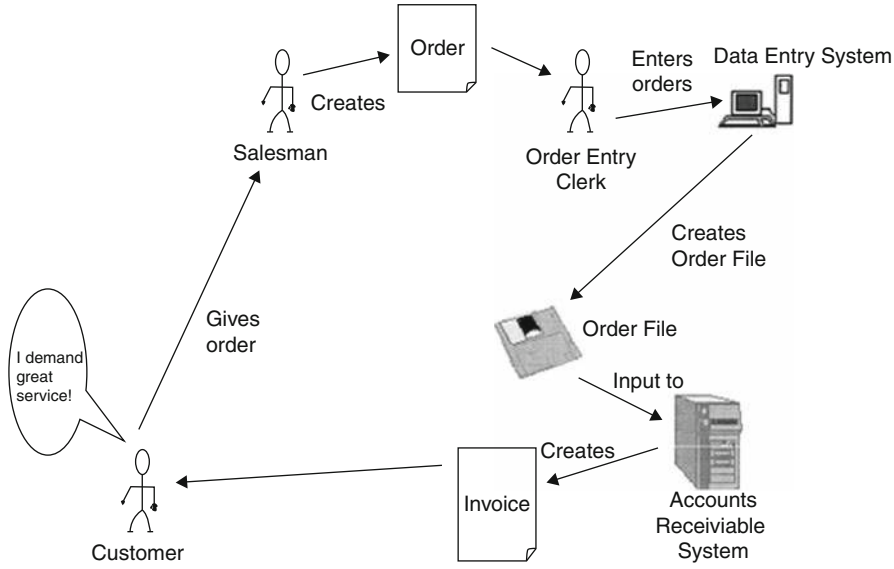
The development techniques focus on data flowing between processes, which transforms the data in some way (DeMarco 1979; Jackson 1983; Yourdon and Constantine 1979), or on data flows between people, each performing different processes (Checkland 1981; Checkland and Holwell 1998). The sample process data flow diagram in Fig. 2 shows the processes as circles connected via directed lines (i.e., data flows) to external entities and data stores. External entities are depicted on the diagram as squares and represent people, organizations, or other computer systems from which and to which information flows. Data stores, depicted in the diagram as open-ended rectangles, indicate files of information that persist over time. The lines connecting the other icons indicate temporary data flowing through the system, hence the term data flow diagram.

Process methodologies and methods have undergone several iterations of refinement to support real-time systems development and increased evaluation of the ethical and human aspects of systems (Ward and Mellor 1985, 1986; Checkland and Holwell 1998; Avison and Wood-Harper 1990). The lack of integration of data throughout analysis and design has led to an abandonment of process methods per se in favor of techniques that provide such integration.

Figure 3 shows the same order problem as that in Fig. 2, but in the more detailed European-school view, which shows people who act as agents involved in the work process. This type of diagram and the related methodologies are explicitly less mechanistic and more humanistic than their American data-flow diagram (DFD) counterpart. As a result, Soft Systems Methods explicitly deal with the nature, type, and impact of human-computer interactions more than other methods of systems development (Checkland and Scholes 1990).



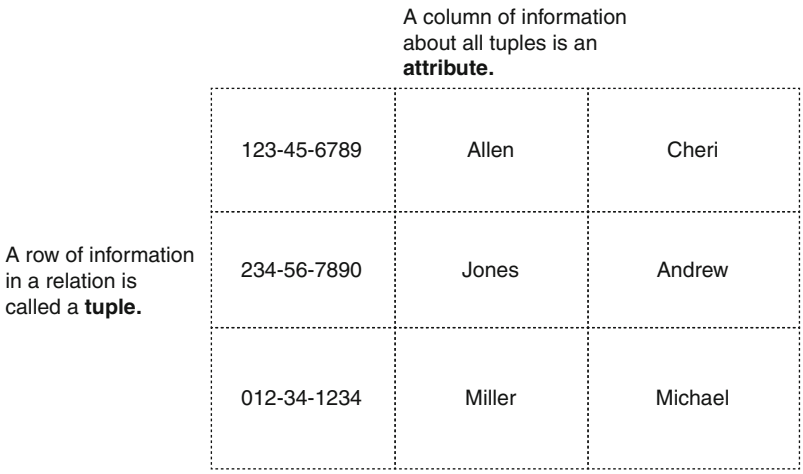
Systems Analysis, Fig. 2 Sample data flow diagram



Systems Analysis, Fig. 3 Human-centric work flow diagram

Data Methodologies — Data methodologies developed as the database technologies that matured in the 1960s and 1970s were found to require specific attention to data design. Data methods are based on theories of semantic modeling (Chen 1981), relational database design (Codd 1972), and data normalization (Kent 1983). These theories are significant in business because they result in mathematically, provably correct processing of data, a key in mission-critical applications. They are also significant because they encouraged the application of mathematical foundations to transaction processing, which had previously relied primarily on analyst and programmer ingenuity and accuracy.

The essence of relational data design is that information should look to the user as if it were composed of rows and columns, similar to a spreadsheet (Fig. 4). The physical implementation



Systems Analysis, Fig. 4 Relational database view

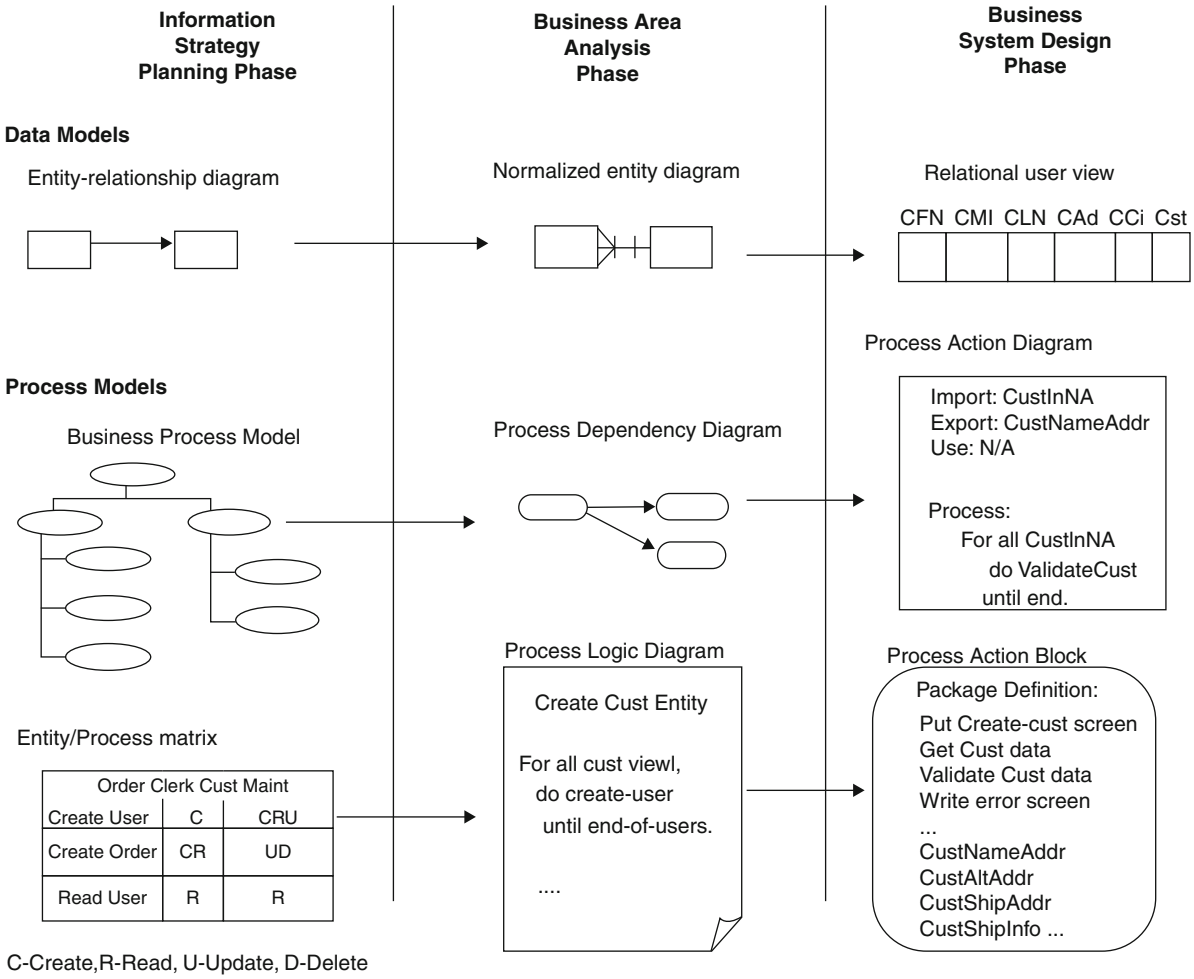
should be transparent, and entity and referential integrity must be maintained. Entity integrity refers to primary keys as unique identifiers of relations and states that no component of a key may accept null values (Date 1990). Referential integrity guarantees that no relation contains unmatched foreign key values. A foreign key is a primary key in one relation that appears as an attribute in another relation (Date 1990).

Early data methods focused only on data, with the assumption that all primitive processing — create, retrieve, update, and delete — followed logically from the correct definition of data (Warnier 1981). Demands to capture the complexities of the world led to significant extensions of process data flow analysis and the development of data modeling with integration of data and process throughout the information engineering (IE) methodology (Martin and Finkelstein 1981). Data models in the form of entity-relationship diagrams, process models similar in form to data flow diagrams, and integration models that link data and process are all found at each stage of information engineering (Fig. 5).

Information-engineered applications are assumed to integrate traditional process-oriented languages with database technologies. Computer-aided software engineering (CASE) tools that support the development of IE applications also generate process program code with imbedded relational database code (e.g., COBOL with embedded SQL). Data methodologies assume on-line applications but can be used for batch processing as well. They are less adapted to real-time applications. Data methodologies are widely used in large, U.S. Fortune 500 organizations that rely on databases containing millions of tuples (e.g., data records that consist of ordered lists of elements).

Object Methodologies — Object techniques were formalized for commercial computing at Xerox PARC in the 1970s with the development of *Smalltalk* and eventual commercialization of the Apple Lisa. As online and real-time technologies migrated from the aerospace and defense industries to commercial development of client–server applications, improved methods were needed to explain the interactions of system elements. Object-oriented analysis (OOA) was the proposed solution. It involves development of three models: (1) an information model describes elements in terms of objects and attributes, (2) a state model describes object behaviors and relationships over time, and (3) a process model specifies object actions in terms of elementary and reusable processes (Schlaer and Mellor 1992).

The goal of object methods is complete integration of data and processes in encapsulated objects (Fig. 6). Objects may be members of classes and exhibit inheritance, a property such that the properties, data, and processes of related objects may be reused without redefinition — that is, inherited. Objects may have multiple inheritances from competing objects throughout a hierarchy (Fig. 7). Objects may also exhibit polymorphism, i.e., the ability to have the same process, using one public name, take different forms when associated with different objects (Booch 1987, 1991). Client/server technology embodies the

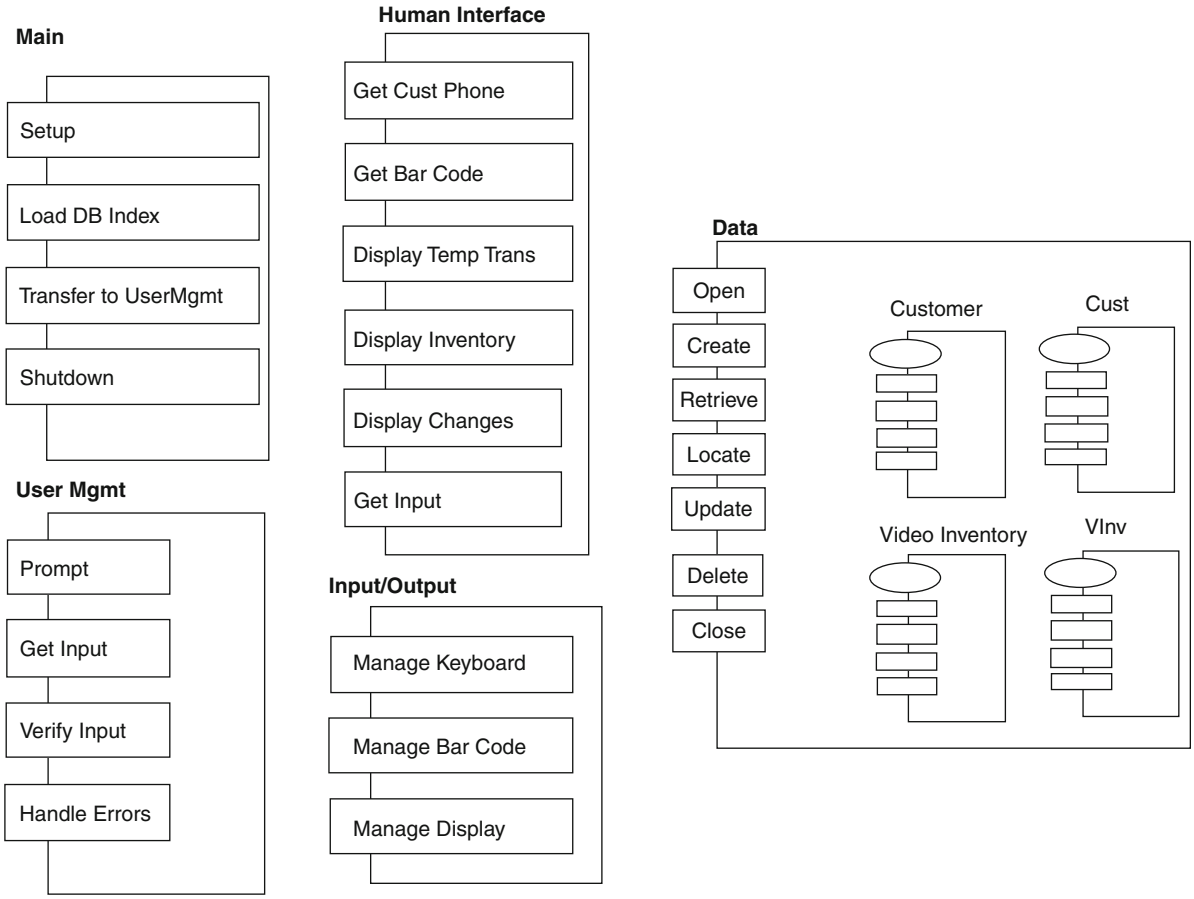


Systems Analysis, Fig. 5 Information engineering data, process, and integration models

concepts of object orientation. Client objects request a process from a supplier or server; server objects perform the requested process.

Object orientation is based on the same theories that data methodologies are based on, carrying normalization to the encapsulated (data + process) object units (Kent 1983). The most visible example of object applications is MS Windows, which uses windows, icons, menus, and pointers in an object-oriented human interface to personal computers. Object-oriented methodologies are currently adopted widely in the embedded systems and software markets (e.g., graphical user interfaces, or GUIs, such as MS Windows).

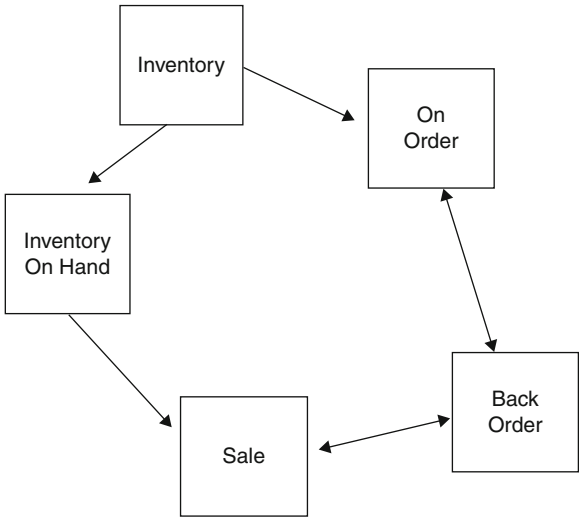
Object methods, which were experimental in the late 1980s, matured considerably during the 1990s, and are applied with great success to problems such as client-server applications. One issue with object methods was the need for an object broker software that interpreted requests, forwarded requests for processing, and coordinated the return of the response. In early object-oriented applications, each application built its own broker. In the late 1990s, broker software was commercialized and popularized through services oriented architectures (SOA). SOA applications are suites of self-contained, loosely coupled preprogrammed services, capable of communicating with others. The services are compiled and made available for applications with reusability a key characteristic. SOA defines how two or more computing entities such as programs are allowed to interact so as to enable each entity to perform a unit of work that can be passed on to another entity.



Systems Analysis, Fig. 6 Encapsulated objects

A mashup (a term from the field of music) is a variation on SOA that combines preprogrammed functionality derived from any number of sources to create a new application or service. A successful mashup removes the burdens of detailed analysis and coding and allows the designer to simply choose and connect to desired functions, thus creating a new application.

The success of object methods notwithstanding, obstacles that appeared intractable in the short run still remain. The two problems that were the biggest obstacles originally still remain the biggest problems — data persistence and standardization of object analysis and design methods. The issue of persistence means that few object-oriented systems are purely object oriented. Rather, they have program code objects that revert



Systems Analysis, Fig. 7 Inheritance lattice of objects

to data methods for database interactions. The issue with standards diminished with the integration of the key methodologies of Booch et al. (1999), coalescing the knowledge base.

Semantic Methodologies

Semantic refers to meaning. Semantic methods focus on the role of knowledge and meaning in a system. They imbed meaning in data and in the reasoning rules used to process it by drawing on theories of cognitive development as it applies to computer reasoning and learning. Two types of semantic methodology have evolved, one dealing with artificial intelligence and the other with presentation of World Wide Web-like business applications.

Artificial Intelligence methodologies — Artificial intelligence (AI) methodologies are used to design computer systems that are able to understand languages, learn, reason, solve problems, and exhibit other characteristics associated with intelligence in human beings. AI methodologies differ from mathematical and transactional methodologies in several ways. AI methods produce a decision, a course of action, or an answer to a question based on the application of qualitative knowledge and information. The more qualitative the systems situation being examined, therefore, the more advanced the techniques required. For instance, whether or not you have toast for breakfast in the morning might depend on numerous factors such as how well you slept the night before, what you had for dinner, and so on. This simple example illustrates two of the many non-trivial problems AI applications must solve: making their reasoning sufficiently generic and identifying all of the relevant quantitative and qualitative relationships in the system.

AI problems deal with incomplete information, probabilistic outcomes, and ambiguities in the reasoning and data to be used in developing a solution. In contrast, traditional methods assume complete information and single outcomes with few or no ambiguities. Similarly, AI methods differ from OR methods. In AI problems, complexes of potentially conflicting rules are expertly reasoned through to a logical conclusion. OR methodologies describe probable situational relationships to develop an optimal solution from an unlimited number of possible outcomes. The difference here is that AI problems have an unknown number of relevant inputs whereas OR problems have an unknown number of outcomes.

A program called DENDRAL was an early result of AI research. DENDRAL is a chemist's assistant that interprets data from a mass spectrograph and infers the chemical structure of an unknown organic compound. The program is based on an algorithm, developed by J. Lederberg in 1964, that generates all possible acyclic graphs given the number of systems elements (the compound's chemical composition) and the number of links (relationships) pertaining to each element (the technical valences). The number of possibilities generated for any given compound is enormous. To avoid an exponential search, DENDRAL automates rules to apply heuristics and knowledge gained from practicing chemists to delimit radically the number of alternatives that must be evaluated to determine the compound's molecular structure. DENDRAL introduced the idea of using *rules* to represent expert knowledge, a concept that has prevailed in AI work since (Feigenbaum et al. 1971). DENDRAL outperforms expert chemists on this task (Buchanan and Feigenbaum 1981; Churchman 1971; Feigenbaum et al. 1971; Smith et al. 1973).

New in the 1990s, neural networks model human intellectual activity on a broad scale by mirroring human brain functioning. A neuron is the smallest possible processing element and is related to other neurons via synapses. Objects called dendrites are message transmitters that flow between neurons over synaptic connections. Single neurons can have thousands of synapses. Inputs via dendrites can either excite (i.e., initiate) or inhibit action of a neuron. The number and frequency of messages sent to a neuron create an activity level that can be triggered when some predefined threshold is reached. Each neuron has axons through which output signals are transmitted to the dendrite network. These terms have parallels in the other AI methodologies but work slightly differently in neural nets. Neural network problems, however, are different in kind from those solved by the other AI methodologies with applications to machine learning, generalization based on past reasoning success, and partial

matching. Familiar applications include pattern recognition of written characters, voices, or faces. Neural net analysis is applied to problems that have subjective outcomes, ambiguous reasoning and components, are vague in definition, or have contradictory results.

The techniques that constitute semantic methodologies are not mature enough to strictly qualify as methodologies. Rather they are individually applied, taught in a master-apprentice relationship, and based on practical experience with a given set of problems.

At this time, different types of reasoning problems require different types of methods and approaches to automating intelligence (Winograd and Flores 1986). The problem types that are addressed by semantic methods include language understanding and translation, sensory understanding (i.e., sight, touch, etc.), memory recall and forgetting, and coordination and control of movement. The most common of these are expert systems, which exhibit intelligence in selecting an action by reasoning through numerous, sometimes contradictory, rules. Expert systems have found acceptance in industry and government for applications such as surveillance of nuclear plant operations, selection of geological drilling sites, and diagnosis of medical problems (Kaufmann and McCorduck 1979). Expert reasoning systems, such as DENDRAL, are ubiquitous and are used in most areas of human activity from basic product design, to agriculture, appliances, medicine, and finance.

AI techniques and methodologies are in an emergent stage, experiencing continuous refinement and evolution. Like object-oriented methods, AI methods are also closely coupled to the target implementation language. For instance, some AI languages require data integration with reasoning rules while others require separation of data from reasoning rules. Most languages offer one reasoning approach that determines the nature of the reasoning process as forward, backward, depth-first, breadth-first, custom-defined, or other. The major commercial promise of AI is to augment existing applications by including reasoning about the processes and data they maintain. Neural nets are promising as generic reasoning systems that may coalesce these diverse methods and techniques some time in the future.

Information Methodologies

Data refers to elementary facts and figures that can be used as a basis for reasoning, calculation or discussion. Information is data that has meaning. High quality information typically is accurate, complete, consistent, unique, and timely. Data, numbers and letters, becomes information (what) which leads to knowledge (how) which, in turn, leads to wisdom (why) (Buckland 1991; Langefors 1966). Wisdom relating to information systems is the desired outcome of information methodologies and technologies. As globalization increases, multi-location and multi-national organizations require historical data storage in a manner that

- Supports large-scale data integration from any number of sources
- Priortizes data source for 'official' data when multiple copies occur
- Stores data for simplicity and speed of retrieval
- Allows multiple views of the data
- Provides easily-used data query and retrieval capabilities.

The first four goals are supported through information methodologies and technologies for information storage and retrieval. The last two goals are supported through methods and technologies for information presentation.

Information Storage and Retrieval – A data warehouse is a collection of data from any number of sources that supports analysis and decision making. Three main areas of concern are imbedded in information methodologies that are de facto conventions rather than formal standards. The term 'data warehouse' first appeared in Inmon (1993) as a tool supporting organizational decision-making. A data warehouse provides for both storage and retrieval of its contents.

The first task of data warehouse analysis is the extract- transform-load (ETL) process through which data is taken from multiple sources, integrated, cleaned, and loaded into a consolidated database. If multiple views of data are provided, this outcome is also known as a 'data mart.' Data warehouses are often referred to as a 'cube,' consisting of data over time and over organizations (hence three dimensions). Design complexity derives from multiple issues relating to logical and physical data design and to tradeoffs between speed, simplicity, and completeness.

Logical design refers to design of the business data and its characteristics, including data type, length, name, source, and so on. Logical design decisions relate to data relationships as hierarchic or networked, how to prioritize data from multiple sources, and data cleansing. Physical design refers to design of the data warehouse within its intended hardware infrastructure. Some key physical design decisions include items such as whether or not to provide in-memory indices, size of data blocks, number and type of storage devices, managing and synchronizing data requests, and number of data request queues.

Data retrieval relates to development of reports through a data warehouse. Three options for reporting include pre-defined queries, structured language queries, and ad hoc queries. As hoc queries often use a capability called 'natural language processing' (NLP). Though data retrieval is typically provided through data warehouse software, the three types of queries are not necessarily guaranteed. These in turn have their own issues with which analysts contend. Pre-defined queries can only be defined for about 80% of processing that is recurring and standardized. An example of this is a monthly report. Structured language and ad hoc queries are useful for the remaining 20% of queries. Both of these raise issues relating to ambiguity, such as 'all sales in New York.' The reference 'New York' could be the city or the state. Thus, the software needs a means to recognize and resolve ambiguities in real-time. Other design issues include how to deal with inaccurate or missing data – reporting it or not, using it in computations or not, and so on. NLP has given rise to one of the emerging problem areas discussed below.

Information Presentation – With Internet commercialization in 1994 and the advent of sophisticated, user-friendly graphical user interface browser software in 1993, businesses flocked to the World Wide Web (Web). By 1996, companies saw the potential to eliminate internal communication bottlenecks via intranets, which also use browser interfaces. By 1997, business-to-business relationships were supported by private extranets and in 1998 by virtual private networks (VPNs) on the Internet. Web innovation was followed shortly by unprecedented acceleration and growth in technologies supported by Web browsers, from text-only in 1994 to full multimedia, audio, video, and telephony support in 1998. The increasing population of Web users has been equally astonishing: from two million (mostly academic users) in the early 1990s to 100 million in 2000 to over 4.5 billion in 2010.

The popularity of the Web as a vehicle for electronic commerce (E-commerce) necessitated development of advanced developer skill sets to support the new technologies of integration and graphical composition. The new analytical skills focus on understanding and presenting business information in ways that add value to users (to keep them coming back to the Web site), and, therefore, increase the profits of the presenting organizations.

Information analysis methodologies, based on systems theory evolved to provide techniques and methods for Web presentation development (Conger and Mason 1998). The first part of information analysis incorporates analysis of user groups' needs and wants and the information the business developing the presentation wants to provide. Then, information analysis organizes data objects into relational structures reflecting how the business and the user groups view information. Information objects are then expressed in unambiguous, concise language for Web presentation. Once information objects are defined, graphical design techniques and methods are applied to identify individual Web pages and to guide the layout of information on each page. Hyperlink analysis is a form of systems analysis used to determine the set of threads between pages that will best accommodate the view of the information objects and their interrelationships that the business wants to present and the user want to see. (Nelson 1981) Finally, multimedia are used to enhance the value of information to users either by augmenting the current presentation or by replacing words with some other, more easily digested representation of the information.

Concluding Remarks

The problems companies seek to solve through automation have shifted to dealing with legacy applications, loosely defined as any application that has been operational in an organization for more than

seven years, to the development of data warehouses to support extended trend analysis. One element that is common to analytical problems is that their solutions will depend on the integration of different theories and methodologies. This is a major shift in emphasis from prior to 2000, during which problem types were addressed by means of separate and distinct problem-solving activities. Three problems persist in systems analysis at present – legacy applications integration, application quality, and universal translation. The first two are persistent problems that have been intractable but appear to have solutions on the horizon. The last is a more recent issue that exemplifies the desire to automate beyond current technical and analytical capabilities.

Legacy Applications — Legacy applications may have been operational anywhere from 7 to 40 years. These applications manage much valuable corporate information and are the cash cows of the information systems world. Legacy applications still contain more than one trillion lines of COBOL code, down by 50% in 10 years. COBOL was developed in 1959 by Grace Hopper well before most current methodologies were adopted by large corporations. Replacement of these applications represents a prohibitively large investment that most companies are reluctant to make unless forced to do so. Legacy applications are often poorly designed, poorly coded, and undocumented. Maintaining this inherently inefficient and bug-ridden code requires increasingly expensive maintenance support staff whose programming skills are essentially obsolete. The challenge to corporations is to replace or up-grade non-strategic legacy applications with superior packaged software or custom-built applications or to outsource operations without making a huge financial reinvestment.

Application Quality — As sophisticated as systems analysis has become, it still often leads to failed applications. Application success and use is best summarized by the DeLone and McLean (1992, 2003) and Petter et al. (2008), who found the following constructs of importance in Table 1.

DeLone and colleagues built on hundreds of other research projects to develop both a parsimonious list of critical factors that generally fits all applications. The details of each characteristic is beyond the scope of this paper, but the key drivers are of interest because they span all applications types with many sub-factors seeming to be universal. Three types of quality are expected of successful applications: System, information, and service (Conger 2011). Systems quality refers to the application in its operational environment and the extent to which it performs at the time needed and in the manner expected. System quality is important because inattention to system quality early in the development cycle can easily result in poor quality upon implementation.

Systems Analysis, Table 1 Key drivers of successful information systems (DeLone and McLean 2003, p. 26)

Key driver	Sub-Characteristics
Systems quality	Adaptability
	Availability
	Reliability
	Response time
	Usability
Information quality	Completeness
	Ease of understanding
	Personalization
	Relevance
	Security
Service quality	Assurance
	Empathy
	Responsiveness
Use	Nature of use
	Navigation patterns
	Number of site visits
	Number of transactions executed
User satisfaction	Repeat purchases
	Repeat visits
	User surveys
Net benefits	Cost savings
	Expanded markets
	Incremental additional sales
	Reduced search costs
	Time savings

Information quality refers to the suitability and usefulness of the data provided to the user. Information quality in any transactional system needs to be complete and accurate. Similarly, relevant, secure data seem to be universal in their appropriateness across application types.

Service quality also may be appropriate for all applications but in a different sense than expressed by the sub-factors provided here. The sub-factors in the De Lone and McLean list are from SERVQUAL,

a well-researched model of service quality in an online environment (Parasuraman et al. 1988). SERQUAL needs additional research to determine characteristics that fit other arenas of IT support. Gap analysis to evaluate expectations versus attributes of objective product, specific characteristics of service quality and how they are developed within IT analysis methodologies. Further, contextualizing service concepts may lead to more accurate service design. For instance, in e-commerce, service and system quality are used interchangeably and no known research has teased out the nuances of their differences.

Thus, completing an application with technical quality is insufficient to develop a contribution to its using organization. Rather, the application in use must comply with all of its needs. Yet, application developers persist in thinking of 'needs' as confined to functional requirements. Rather, functional and non-functional requirements are necessary, as are requirements for more ephemeral aspects of contribution such as simplicity, learnability, usability, and so on (Nielsen 2000). To determine value added to an organization, application development must attend to an application's use in context, particularly as it relates to the using organization's success. Current thinking on these operational activities is that taking a services orientation that mirrors the services orientation the organization seeks to perfect, will lead to value-adding outcomes for IT. IT service management is in its infancy as it relates to systems analysis but promises to become an increasingly important aspect of analysis activities (Conger 2011).

Universal Translation — The ultimate goal in machine translation (also called computational linguistics) is to support machine-only, seamless translation from one language to any other without special equipment or human intervention. Work on universal translation dates to the 1950s and was then conducted using semantic methods. However, as understanding of the uses for this technology became more global, the work focused less on software and more on data. The idea was for language translation technology to be imbedded in any type of application and applied to data into and out of the application while being stored in the original language. With this technology, a purchase via the Web, conducted through, e.g., a smart phone, would identify global goods and prices all displayed in the user's preferred language.

For systems analysis, hybrid techniques, including aspects of mathematical, AI, information, and object methodologies are all required to result in embeddable translation capabilities. In addition, these computer-oriented methodologies must be successfully coupled with linguistics methodologies for morphology, syntax, semantics, and inference understanding and automation. This is a huge task that has had great success in limited domains, such as automated voice response systems and online web sites such as babelfish.com.

For the first 30 years, machine translation technology focused on a process that decoded source language to an intermediate, neutral language then encoded the neutral language to target language. This approach has been marginally successful. More recently, research that removes the intermediate language has shown promise but runs into language accuracy issues. Multiple 'grammars' have been developed to identify intra-lingual relationships including, generative, dependency, unification, and case. Then, hybrid methods arose that combine elements of intermediary language with some direct translation. Probably the most successful translations are performed within a known domain, such as going to a restaurant, however, no complete, unaided translation systems have been successful to date. The issues relating to this capability lie not only in technology but also in linguistics knowledge and capabilities to describe languages.

Technologies for machine translation capabilities exist in definitional tools such as extensible markup (XML), OWL (web ontology language), and other World Wide Web Consortium (W3C) developments for web services. One problem is that vendor support for these languages imbed custom tags and, therefore, require adherence to some vendor's products. In addition, attempts at reaching agreement on a global ontology to categorize domain information, such as health care, have not been successful. As with all systems analyses, global projects take time and at some point participants become impatient, leave, and develop their own ontology, often thinking that it will become a de facto standard. In developing something custom, these ontologies automatically become biased and/or incomplete. In addition, these technologies and lingual grammars are complex. Automating them is even more complex, thus rendering them beyond the reach of the average application analyst.

The state of linguistics knowledge also delays development of universal translation. Much is understood about languages and language structures in terms of their basic syntax (e.g., noun – verb-object structures versus other orders or alternatives). However, issues such as ambiguity, ('the man hit the woman with the baby', 'whatever'), alternative meanings and uses of words (e.g., "hit" as subject, verb, or object), and idioms ('he went to town on that steak') all cause havoc with translators and these are just English examples. Further, until projects such as propbank, verbnet, and semlink are completed in many languages, unaided, accurate machine translation will remain elusive.

See

- ▶ Artificial Intelligence
- ▶ Control Theory
- ▶ Cybernetics and Complex Adaptive Systems
- ▶ Data Warehousing
- ▶ Expert Systems
- ▶ Mathematical Model
- ▶ Neural Networks
- ▶ Practice of Operations Research and Management Science
- ▶ System Dynamics

References

Ackoff, R. L., & Rivett, P. (1963). *A manager's guide to operations research*. New York: Wiley.

Alavi, M., & Leidner, D. (2001). Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues. *MIS Quarterly, 25*(1), 107–136.

Applebaum, R., Cline, M., & Girou, M. (1998). *CORBA FAQs*. Downloaded September 10, 1998, from http://octavia.anu.edu.au/markus/corbafaq

Avison, D., & Wood-Harper, T. (1990). *Multiview: An exploration in information systems development*. Oxford, UK: Blackford Press.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., et al. (2001). *The agile manifesto*. http://agilemanifesto.org/

Berge, C. (1962). *The theory of graphs and its applications*. New York: Wiley.

Booch, G. (1987). *Software engineering with Ada* (2nd ed.). Menlo Park: Benjamin/Cummings Publishing.

Booch, G. (1991). *Object oriented design with applications*. Redwood City: Benjamin/Cummings Publishing.

Booch, G., Rumbaugh, J., & Jacobson, I. (1999). *The unified language user guide*. Reading: Addison-Wesley.

Buchanan, B., & Feigenbaum, E. (1981). DENDRAL and meta DENDRAL: Their application dimension. *Artificial Intelligence, 11*(1), 5–24.

Buckland, M. (1991). *Information and information systems*. New York: Praeger.

Checkland, P. (1981). *Systems thinking, systems practice*. Chichester: Wiley.

Checkland, P., & Holwell, S. (1998). *Information, systems and information systems. Making sense of the field*. Chichester: Wiley.

Checkland, P., & Scholes, J. (1990). *Soft systems methodology in action*. Chichester: Wiley.

Chen, P. P.-S. (1981). A preliminary framework for entity-relationship models. In P. P.-S. Chen (Ed.), *Entity-relationship approach to information modeling and analysis*. Saugus: ER Institute.

Churchman, C. W. (1968). *The systems approach*. New York: Delta (Dell) Publishing.

Churchman, C. W. (1971). *The design of inquiring systems*. New York: Basic Books.

Churchman, C. W., Ackoff, R. L., & Arnoff, E. L. (1957). *Introduction to operations research*. New York: Wiley.

Codd, E. F. (1972). A relational model of data for large shared data banks. *Communications of the ACM, 13*(6), 377–387.

Conger, S. (1994). *The new software engineering*. Belmont: Wadsworth.

Conger, S. (2011, forthcoming). Software development life cycles and methodologies: Fixing the old and adopting the new. *International Journal of Information Technologies and the Systems Approach* (*IJITSA*), *4*(1).

Conger, S., & Mason, R. O. (1998). *Planning and designing effective web sites*. Boston: Course Technology.

Dantzig, G. (1963). *Linear programming and extensions*. New Jersey: Princeton University.

Date, C. J. (1990). *An introduction to database systems* (5th ed.). Reading: Addison-Wesley.

DeLone, W. H., & McLean, E. R. (1992). Information systems success: The quest for the dependent variable. *Information Systems Research, 3*(1), 60–95.

DeLone, W. H., & McLean, E. R. (2003). The DeLone and McLean model of information systems success: A ten-year update. *Journal of Management Information Systems, 19*(4), 9–30.

DeMarco, T. (1979). *Structured analysis*. New York: Yourdon Press.

Feigenbaum, E., Buchanan, B., & Lederberg, J. (1971). On generality and problem solving. In B. Beltzer & D. Michie (Eds.), *Machine intelligence* (pp. 165–190). New York: Elsevier.

Hackman, J. R., & Oldham, G. R. (1980). *Work redesign*. Reading: Addison-Wesley.

Inmon, W. H. (1993). *Building the data warehouse* (1st ed.). New York: Wiley.

Jackson, M. (1983). *Systems development*. London: Prentice-Hall.

Jarke, M., & Vassiliou, Y. (1997). Data warehouse quality: A review of the DWQ project. In *Proceedings of the 2nd conference on information quality*. Cambridge: Massachusetts Institute of Technology.

Kasabov, N. K. (1996). *Foundations of neural networks, fuzzy systems, and knowledge engineering (computational intelligence)*. Boston: The MIT Press.

Kaufmann, K., & McCorduck, P. (1979). *Machines who think*. New York: W. H. Freeman.

Kent, W. (1983). A simple guide to five normal forms in relational database theory. *Communications of the ACM, 26*(2), 120–125.

Langefors, B. (1966). *Theoretical analysis of information systems*. Lund: Studentlitteratur.

Laszlo, E. (1972). *The systems view of the world*. New York: Wiley.

Martin, J., & Finkelstein, C. (1981). *Information engineering*. Englewood Cliffs: Prentice-Hall.

Minsky, M. (Ed.). (1968). *Semantic information processing*. Cambridge, MA: MIT Press.

Nelson, T. (1981). *Literary machines*. Sausalito: Mindful Press.

Nielsen, J. (2000). *Usability engineering*. San Diego: Kaufmann.

Object Management Group. (1998). As downloaded September 10, 1998, from http://www.omg.org

Papert, S. (1980). *Mind-storms: Children, computers, and powerful ideas*. New York: Basic Books.

Parasuraman, A., Zeithaml, V., & Berry, L. L. (1988). SERVQUAL: A multiple-item scale for measuring consumer perceptions of service quality. *Journal of Retailing, 64*(1), 12–37.

Petter, S., Delone, W., & Mclean, E. (2008). Measuring information systems success: Models, dimensions, measures, and interrelationships. *European Journal of Information Systems. Basingstoke, 17*(3), 236–263.

Schlaer, S., & Mellor, S. J. (1992). *Object lifecycles: Modeling the world in states*. Englewood Cliffs: Yourdon Press.

Smith, D., Buchanan, B., Engelmore, R., Adlercreutz, J., & Djerassi, C. (1973). Application of artificial intelligence for chemical inference IX. *Journal of the American Chemical Society, 95*, 6078.

Ward, P. T., & Mellor, S. J. (1985, 1986). *Structured development for real-time systems*. New York: Yourdon Press.

Warnier, J. D. (1981). *The logical construction of systems*. New York: Van Nostrand Reinhold.

Winograd, T., & Flores, F. (1986). *Understanding computers and cognition*. Norwood: Alex.

Yourdon, E., & Constantine, L. L. (1979). *Structured design: Fundamentals of a discipline of computer program and systems design*. Englewood Cliffs: Prentice-Hall.

T

Tableau

► Simplex Tableau

Tabu Search

Fred W. Glover
OptTek Systems, Inc., Boulder, CO, USA

Introduction

Tabu Search (TS) is a metaheuristic that guides a local heuristic search procedure to explore the solution space beyond local optimality. Widespread successes in practical applications of optimization include finding better solutions to problems in scheduling, sequencing, resource allocation, investment planning, telecommunications and many other areas. Some of the diversity of tabu search applications is shown in Table 1. (For a more comprehensive list of applications, see the book by Glover and Laguna 1997.)

Tabu search is based on the premise that methods for complex optimization problems, particularly those arising in real world applications, can function more effectively if they incorporate flexible and responsive memory. Accompanying this premise is the corollary that such memory is employed together with strategies expressly designed for exploiting it. More broadly, tabu search embodies the following principle: If a problem has exploitable features, but contains a structure sufficiently complex to prevent these features from being known in advance, then a method can derive advantages by monitoring its behavior in relation to the space in which it operates. The purpose of the monitoring is effectively to generate a map of the regions the method has visited as a foundation for modifying its behavior, where this map can take multiple forms that ultimately become expressed in the decision rules employed to negotiate the solution space. The hallmark of a TS method is therefore a capacity to guide its progress by reference to its own unfolding history. Such a method evidently is implicitly or explicitly structured to employ learning. Based on this perspective, methods that incorporate a significant portion of the tabu search framework are sometimes called Adaptive Memory Programming (AMP) methods.

The emphasis on responsive exploration (and hence purpose) in tabu search, whether in a deterministic or probabilistic implementation, derives from the supposition that a bad strategic choice can yield more information than a good random choice. In a system that uses memory, a bad choice based on strategy can provide useful clues about how the strategy may profitably be changed. Even in a space with significant randomness – which fortunately is not pervasive enough to extinguish all remnants of order in most real-world problems – a purposeful design can be more adept at uncovering the imprint of structure, and thereby at affording a chance to exploit the conditions where randomness is not all-encompassing.

These basic elements of tabu search have several important features that are summarized in Table 2.

Tabu search is concerned with finding new and more effective ways of taking advantage of the concepts embodied in Table 2, and with identifying associated principles that can expand the foundations

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Tabu Search, Table 1 Illustrative tabu search applications

Scheduling	**Telecommunications**
Flow-Time Cell Manufacturing	Call Routing
Heterogeneous Processor Scheduling	Bandwidth Packing
Hub Facility Location	
Workforce Planning	Path Assignment
Classroom Scheduling	Network Design for Services
Machine Scheduling	Customer Discount Planning
Flow Shop Scheduling	Failure Immune Architecture
Job Shop Scheduling	Synchronous Optical Networks
Sequencing and Batching	
Design	**Production, Inventory and Investment**
Computer-Aided Design	Flexible Manufacturing
Fault Tolerant Networks	Just-in-Time Production
Transport Network Design	Capacitated MRP
Architectural Space Planning	Part Selection
Diagram Coherency	Multi-item Inventory Planning
Fixed Charge Network Design	Volume Discount Acquisition
Irregular Cutting Problems	Fixed Mix Investment
Lay-Out Planning	
Location and Allocation	**Routing**
Multicommodity Location/ Allocation	Vehicle Routing
Quadratic Assignment	Capacitated Routing
Quadratic Semi-Assignment	Time Window Routing
Multilevel Generalized Assignment	Multi-Mode Routing
	Mixed Fleet Routing
	Traveling Salesman
	Traveling Purchaser
	Convoy Scheduling
Logic and Artificial Intelligence	**Graph Optimization**
Maximum Satisfiability	Graph Partitioning
Probabilistic Logic	Graph Coloring
Clustering	Clique Partitioning
Pattern Recognition/ Classification	Maximum Clique Problems
Data Integrity	Maximum Planner Graphs
Neural Network Trainings	P-Median Problems
Neural Network Design	
Technology	**General Combinational Optimization**
Seismic Inversion	Zero–one Programming
Electrical Power Distribution	Fixed Charge Optimization
Engineering Structural Design	Nonconvex Nonlinear Programming
Minimum Volume Ellipsoids	All-or-None Networks
Space Station Construction	Bilevel Programming
Circuit Cell Placement	General Mixed Integer Optimization
Off-Shore Oil Exploration	

Tabu Search, Table 2 Principal tabu search features

Adaptive Memory
Selectivity (including strategic forgetting)
Abstraction and decomposition (through explicit and attributive memory)
Timing:
recency of events
frequency of events
differentiation between short term and long term
Quality and impact:
relative attractiveness of alternative choices
magnitude of changes in structure or constraining relationships
Context:
regional interdependence
structural interdependence
sequential interdependence
Responsive Exploration
Strategically imposed restraints and inducements (tabu conditions and aspiration levels)
Concentrated focus on good regions and good solution features (intensification processes)
Characterizing and exploring promising new regions (diversification processes)
Non-montonic search patterns (strategic oscillation)
Integrating and extending solutions (path relinking)

of intelligent search. As this occurs, new strategic mixes of the basic ideas emerge, leading to improved solutions and better practical implementations.

Tabu Search Foundations

The basis for tabu search may be described as follows. Given a function $f(x)$ to be optimized over a set X, TS begins in the same way as ordinary local search, proceeding iteratively from one point (solution) to another until a chosen termination criterion is satisfied. Each $x \in X$ has an associated neighborhood $N(x) \subset X$, and each solution $x' \in N(x)$ is reached from x by an operation called a move.

TS goes beyond local search by employing a strategy of modifying $N(x)$ as the search progresses, effectively replacing it by another neighborhood $N^*(x)$. As the previous discussion intimates, a key aspect of tabu search is the use of special memory structures which serve to determine $N^*(x)$, and hence to organize the way in which the space is explored.

The solutions admitted to $N^*(x)$ by these memory structures are determined in several ways. One of these, which gives tabu search its name, identifies solutions encountered over a specified horizon (and implicitly, additional related solutions), and forbids them to belong to $N^*(x)$ by classifying them tabu (The tabu terminology is intended to convey a type of restraint that embodies a cultural connotation – i.e., one that is subject to the influence of history and context, and capable of being surmounted under appropriate conditions).

The process by which solutions acquire a tabu status has several facets, designed to promote a judiciously aggressive examination of new points. A useful way of viewing and implementing this process is to conceive of replacing original evaluations of solutions by tabu evaluations, which introduce penalties to significantly discourage the choice of tabu solutions (i.e., those preferably to be excluded from $N^*(x)$, according to their dependence on the elements that compose tabu status). In addition, tabu evaluations also periodically include inducements to encourage the choice of other types of solutions, as a result of aspiration levels and longer term influences. The following subsections describe how tabu search takes advantage of memory (and hence learning processes) to carry out these functions.

Explicit and *Attributive Memory* – The memory used in TS is both explicit and attributive. Explicit memory records complete solutions, typically consisting of elite solutions visited during the search (or highly attractive but unexplored neighbors of such solutions). These special solutions are introduced at strategic intervals to enlarge $N^*(x)$, and thereby provide useful options not in $N(x)$.

TS memory is also designed to exert a more subtle effect on the search through the use of attributive memory, which records information about solution attributes that change in moving from one solution to another. For example, in a graph or network setting, attributes can consist of nodes or arcs that are added, dropped or repositioned by the moves executed. In more abstract problem formulations, attributes may correspond to values of variables or functions. Sometimes attributes are also strategically combined to create other attributes by using vocabulary building methods (Glover and Laguna 1993; Glover 1999; Glover et al. 2000).

Short-Term Memory and its Accompaniments – An important distinction in TS arises by differentiating between short-term memory and longer-term memory. Each type of memory is accompanied by its own special strategies. The most commonly used short-term memory keeps track of solution attributes that have changed during the recent past, and is called recency-based memory. To exploit this memory, selected attributes that occur in solutions recently visited are designated tabu-active, and solutions that contain tabu-active elements, or particular combinations of these attributes, are those that become tabu. This prevents certain solutions from the recent past from belonging to $N^*(x)$ and hence from being revisited. Other solutions that share such tabu-active attributes are also similarly prevented from being revisited. The use of tabu evaluations, with large penalties assigned to appropriate sets of tabu-active attributes, can allow tabu status to vary by degrees.

Managing Recency-Based Memory – The process is managed by creating one or several tabu lists, which record the tabu-active attributes and implicitly or explicitly identify their current status. The duration that an attribute remains tabu-active (measured in numbers of iterations) is called its tabu tenure. Tabu tenure can vary for different types or combinations of attributes, and can also vary over different intervals of time or stages of search. This varying tenure makes it possible to create different kinds of tradeoffs between short-term and longer-term strategies. It also provides a dynamic and robust form of search. (See, e.g., Glover 1990; Taillard 1991, Glover and Laguna 1993, 1997.)

Aspiration Levels – An important element of flexibility in tabu search is introduced by means of aspiration criteria. The tabu status of a solution (or a move) can be overruled if certain conditions are met, expressed in the form of aspiration levels. In effect, these aspiration levels provide thresholds of attractiveness that govern whether the solutions may be considered admissible in spite of being classified tabu. Clearly a solution better than any previously seen deserves to be considered admissible. Similar criteria of solution quality provide aspiration criteria over subsets of solutions that belong to common regions or that share specified features (such as a particular functional value or level of infeasibility). Additional examples of aspiration criteria are provided later.

Candidate List Strategies – The aggressive aspect of TS is reinforced by seeking the best available move

that can be determined with an appropriate amount of effort. It should be kept in mind that the meaning of best is not limited to the objective function evaluation. (As already noted, tabu evaluations are affected by penalties and inducements determined by the search history.) For situations where $N^*(x)$ is large or its elements are expensive to evaluate, candidate list strategies are used to restrict the number of solutions examined on a given iteration.

Because of the importance TS attaches to selecting elements judiciously, efficient rules for generating and evaluating good candidates are critical to the search process. Even where candidate list strategies are not used explicitly, memory structures to give efficient updates of move evaluations from one iteration to another, and to reduce the effort of finding best or near best moves, are often integral to TS implementations. Intelligent updating can appreciably reduce solution times, and the inclusion of explicit candidate list strategies, for problems that are large, can significantly magnify the resulting benefits.

The operation of these short-term elements is illustrated in Fig. 1. The representation of penalties in Fig. 1 either as large or very small expresses a thresholding effect: either the tabu status yields a greatly deteriorated evaluation or else it chiefly serves to break ties among solutions with highest evaluations. Such an effect of course can be modulated to shift evaluations across levels other than these extremes. If all moves currently available lead to solutions that are tabu (with evaluations that normally would exclude them from being selected), the penalties result in choosing a "least tabu" solution.

The TS variant called probabilistic tabu search follows a corresponding design, with a short-term component that can be represented by the same diagram. The approach additionally keeps track of tabu evaluations generated during the process that results in selecting a move. Based on this record, the move is chosen probabilistically from the pool of those evaluated (or from a subset of the best members of this pool), weighting the moves so that those with higher evaluations are especially favored. Fuller discussions of probabilistic tabu search are found in Glover (1989), Glover and Laguna (1997), Soriano and Gendreau (1993) and Crainic et al. (1993).

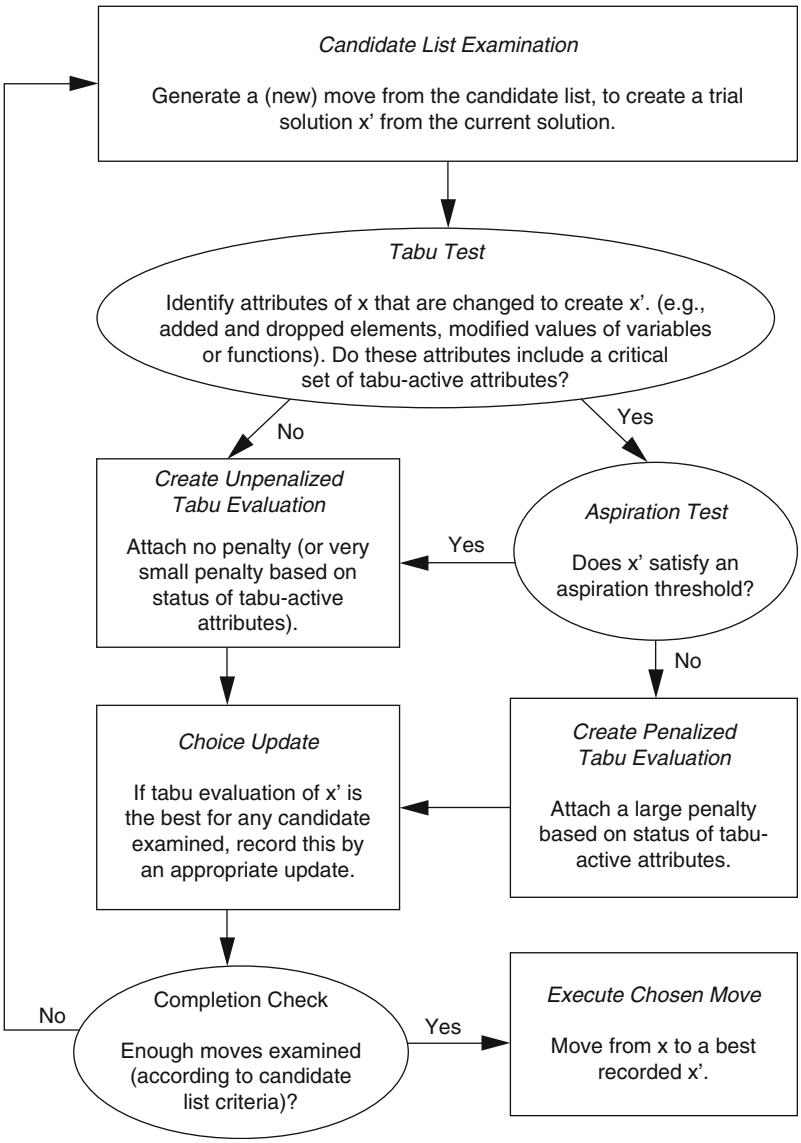
Longer-Term Memory – In some applications, the short-term TS memory components are sufficient to produce very high quality solutions. However, in general, TS becomes significantly stronger by including longer-term memory and its associated strategies.

Special types of frequency-based memory are fundamental to longer-term considerations. These operate by introducing penalties and inducements determined by the relative span of time that attributes have belonged to solutions visited by the search, allowing for regional differentiation.

Perhaps surprisingly, the use of longer-term memory does not require long solution runs before its benefits become visible. Often its improvements begin to be manifest in a relatively modest length of time, and can allow solution efforts to be terminated somewhat earlier than otherwise possible, due to finding very high quality solutions within an economical time span. The fastest methods for job shop and flow shop scheduling problems, for example, are based on including longer-term TS memory. On the other hand, it is also true that the chance of finding still better solutions as time grows – in the case where an optimal solution is not already found – is enhanced by using longer-term TS memory in addition to short-term memory.

Intensification and Diversification – Two highly important longer-term components of tabu search are intensification strategies and diversification strategies. Intensification strategies are based on modifying choice rules to encourage move combinations and solution features historically found good. They may also initiate a return to attractive regions to search them more thoroughly. A simple instance of this second type of intensification strategy is shown in Fig. 2.

The strategy for selecting elite solutions is italicized in Fig. 2 due to its importance. Two variants have proved quite successful. One, due to, introduces a diversification measure to assure the solutions recorded differ from each other by a desired degree, and then erases all short-term memory before resuming from the best of the recorded solutions. The other variant, due to Nowicki and Smutnicki (1993), keeps a bounded length sequential list that adds a new solution at the end only if it is better than any previously seen. The current last member of the list is always the one chosen (and removed) as a basis for resuming search. However, TS short-term memory that accompanied this solution also is saved, and the first move also forbids the move previously taken from this solution, so that a new solution path will be launched.

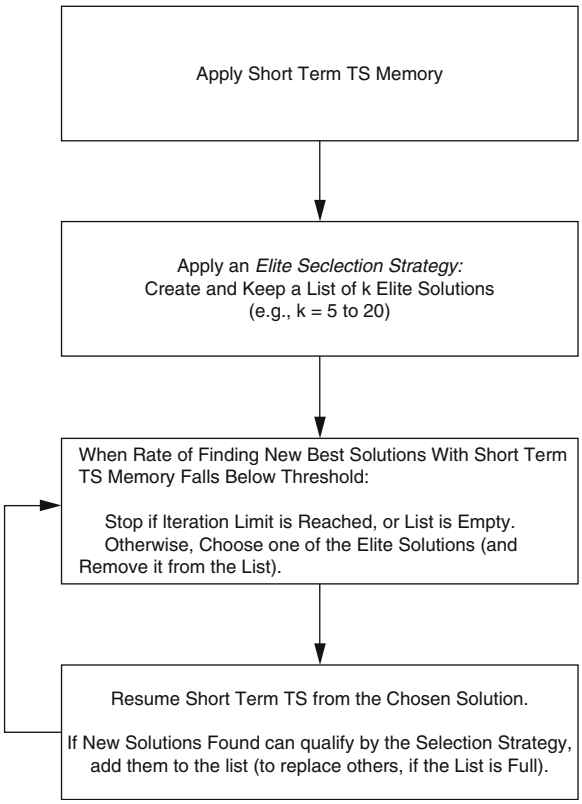


Tabu Search, Fig. 1 Tabu evaluation (short term memory)

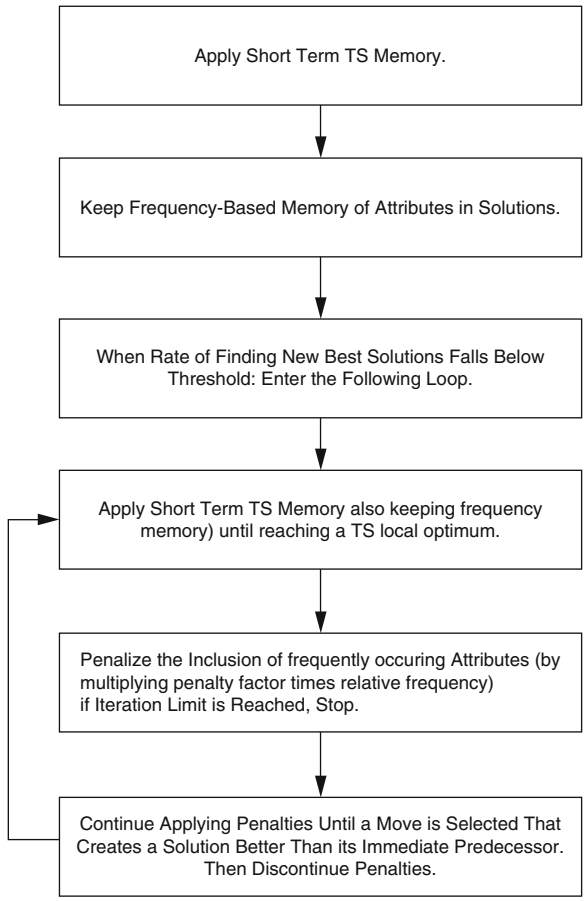
This second variant is related to a strategy that resumes the search from unvisited neighbors of solutions previously generated (Glover 1990). Such a strategy keeps track of the quality of these neighbors to select an elite set, and restricts attention to specific types of solutions, such as neighbors of local optima or neighbors of solutions visited on steps immediately before reaching such local optima. This type of unvisited neighbor strategy has been little examined. It is noteworthy, however, that the two variants previously indicated have provided solutions of remarkably high quality.

Diversification Strategies – TS diversification strategies, as their name suggests, are designed to drive the search into new regions. Often they are based on modifying choice rules to bring attributes into the solution that are infrequently used. Alternatively, they may introduce such attributes by partially or fully re-starting the solution process.

The same types of memories previously described are useful as a foundation for such procedures, although these memories are maintained over different (generally larger) subsets of solutions than those maintained by intensification strategies. A simple diversification



Tabu Search, Fig. 2 Simple TS intensification approach



Tabu Search, Fig. 3 Simple TS diversification approach

approach that keeps a frequency-based memory over all solutions previously generated, and that has proved very successful for machine scheduling problems, is shown in Fig. 3. Significant improvements over the application of short term TS memory have been achieved by this procedure.

Diversification strategies that create partial or full restarts are important for problems and neighborhood structures where a solution trajectory can become isolated from worthwhile new alternatives unless a radical change is introduced. Diversification strategies can also utilize a long-term form of recency-based memory, which results by increasing the tabu tenure of solution attributes.

The two special TS strategies called path relinking and strategic oscillation embody aspects of both intensification and diversification and have proved highly effective in a variety of contexts (Glover and Laguna 1993; Yagiura et al. 2006). The determination of effective ways to balance the concerns of intensification and diversification represents a promising research area. These concerns also lie at the heart of effective parallel processing implementations. The goal from the TS perspective is to design patterns of communication and information sharing across subsets of processors in order to achieve the best tradeoffs between intensification and diversification functions. General analyses and studies of parallel processing with tabu search are given in Taillard (1991, 1993), Battiti and Tecchiolli (1992), Chakrapani and Skorin-Kapov (1993), and Crainic et al. (1993a, 1993b).

Concluding Remarks

Complementarities among the perspectives of tabu search and those favored by the artificial intelligence and neural network communities raise the possibility of creating systems that integrate their fundamental

concerns. Examples are provided by the creation of tabu training and learning models (de Werra and Hertz 1989; Beyer and Ogier 1991; Battiti and Tecchioli 1993; Gee and Prager 1994) and tabu machines (Chakrapani and Skorin-Kapov 1993). The outcomes from this work have shown promising consequences for supplementing customary connectionist models – as by yielding levels of performance notably superior to that of models based on Boltzmann machines, and by yielding processes for modifying network linkages that give more reliable mappings of inputs to outputs.

The practical successes of tabu search have promoted useful research into ways to exploit its underlying ideas more fully. At the same time, many facets of these ideas remain to be explored. The issues of identifying best combinations of short- and long-term memory and best balances of intensification and diversification strategies still contain many unexamined corners (Glover 2007), and some of them undoubtedly harbor important discoveries for developing more powerful solution methods in the future.

Fundamental advances in applications of tabu search have been assembled in a collection of "Tabu Search Vignettes" accessible via the Internet at the author's Web site. These include summaries of key developments in a variety of areas, including:

Constraint Solving and Its Applications (Resource Assignment, Planning and Timetabling, Integer Programming Feasibility, Satisfiability, Mobile Network Frequency Assignment)
Chemical Industry Applications (Computer Aided Molecular Design (CAMD), Heat Exchanger Network (HEN) Synthesis, Phase Equilibrium Calculations, Gibbs Free Energy Minimization, Optimal Component Lumping Problems)
Classification
Feature Selection
Satellite Range Scheduling
Maritime Transportation for International Trade
Conservation Area Network Design
High Level Synthesis
Graph Coloring
Delivery
Routing with Loading and Inventory Constraints
Heterogeneous Routing and Scheduling
Capacitated Facility Location
Multi-period Forest Harvesting
Manpower Scheduling
DNA Sequencing
Airline Disruption Management
Internet Traffic Engineering
Matrix Bandwidth Minimization
Generalized Assignment
Constraint Satisfaction (Work Shift Scheduling, Set-Covering and Nurse Scheduling)
Resource-Constrained Project Scheduling
Dynamic Optimization (Trade Market Prediction, Meteorological Forecast, Robotics Motion Control)

See

- ▶ Artificial Intelligence
- ▶ Heuristics
- ▶ Metaheuristics
- ▶ Neural Networks

References

Battiti, R., & Tecchioli, G. (1993). *Training neural nets with the reactive tabu search*. Technical Report UTM 421, University of Trento, Italy, November.

Battiti, R., & Tecchiolli, G. (1992). Parallel biased search for combinatorial optimization: Genetic algorithms and TABU. *Microprocessors and Micro-Systems, 16*, 351–367.

Battiti, R., & Tecchiolli, G. (1994). The reactive tabu search. *ORSA Journal on Computing, 6*, 126–140.

Beyer, D., & Ogier, R. (1991). Tabu learning: A neural network search method for solving nonconvex optimization problems. *Proceedings of the International Joint Conference on Neural Networks*, IEEE and INNS, Singapore.

Chakrapani, J., & Skorin-Kapov, J. (1993). Connection machine implementation of a tabu search algorithm for the traveling salesman problem. *Journal of Computing and Information Technology (CIT), 1*(1), 29–36.

Crainic, T. G., Gendreau, M., Soriano, P., & Toulouse, M. (1993). A tabu search procedure for multi-commodity location/allocation with balancing requirements. *Annals of Operations Research, 41*(1–4), 359–383.

Crainic, T. G., Toulouse, M., & Gendreau, M. (1993a). *A study of synchronous parallelization strategies for tabu search*. Publication 934, Centre de recherche sur les transports, Université de Montréal.

Crainic, T. G., Toulouse, M., & Gendreau, M. (1993b). *Appraisal of asynchronous parallelization approaches for tabu search algorithms*. Publication 935, Centre de recherche sur les transports, Université de Montréal.

de Werra, D., & Hertz, A. (1989). Tabu search techniques: A tutorial and an applications to neural networks. *OR Spectrum, 11*, 131–141.

Gee, A. H., & Prager, R. W. (1994). Polyhedral combinatorics and neural networks. *Neural Computation, 6*, 161–180.

Gendreau, M., Soriano, P., & Salvail, L. (1993). Solving the maximum clique problem using a tabu search approach. *Annals of Operations Research, 41*, 385–404.
Glover, F. (1989). Tabu search-part I. *ORSA Journal on Computing, 1*, 190–206.
Glover, F. (1990). Tabu search-part II. *ORSA Journal on Computing, 2*, 4–32.
Glover, F. (1995). Tabu thresholding: Improved search by nonmonotonic. *ORSA Journal on Computing, 7*, 426–442.
Glover, F. (1999). Scatter search and path relinking. In D. Corne, M. Dorigo, & F. Glover (Eds.), *New ideas in optimization* (pp. 297–316). UK: McGraw Hill.
Glover, F. (2007). Tabu search – uncharted domains. *Annals of Operations Research, 149*(1), 89–98.
Glover, F., & Laguna, M. (1993). Tabu search. In C. Reeves (Ed.), *Modern heuristic techniques for combinatorial problems* (pp. 70–141). Oxford: Blackwell.
Glover, F., & Laguna, M. (1997). *Tabu search*. Norwell, MA: Kluwer.
Glover, F., Laguna, M., & Marti, R. (2000). Fundamentals of scatter search and path relinking. *Control and Cybernetics, 29*(3), 653–684.
Glover, F., Laguna, M., Taillard, E., & de Werra, D. (Eds.) (1993). *Tabu search*. Special issue of the Annals of Operations Research (Vol. 41). J.C. Baltzer.
Hansen, P., & Jaumard, B. (1990). Algorithms for the maximum satisfiability problem. *Computing, 44*, 279–303.
Hertz, A., & de Werra, D. (1991). The tabu search metaheuristic: How we used it. *Annals of Mathematics and Artificial Intelligence, 1*, 111–121.
Nowicki, E., & Smutnicki, C. (1993). *A fast taboo search algorithm for the job shop problem*. Report 8/93, Institute of Engineering Cybernetics, Technical University of Wroclaw.
Soriano, P., & Gendreau, M. (1993). *Diversification strategies in tabu search algorithms for the maximum clique problem*. Publication #940, Centre de Recherche sur les Transports, Université de Montréal.
Taillard, E. (1991). *Parallel tabu search technique for the job shop scheduling problem*. Research Report ORWP 91/10, Departement de Mathematiques, Ecole Polytechnique Federale de Lausanne.
Taillard, E. (1993). Parallel iterative search methods for vehicle routing problems. *Networks, 23*, 661–673.
Yagiura, M., Ibaraki, T., & Glover, F. (2006). A path relinking approach with ejection chains for the generalized assignment problem. *European Journal of Operational Research, 169*, 548–569.

Taguchi Loss Function

► Total Quality Management

Tail Distribution Function

For a random variable X, $\Pr\{X>x\}$. For a c.d.f. F, $F^c = 1 - F$, also known as the complementary CDF.

Tandem Queues

Queues in series.

See

► Networks of Queues

Technological Coefficients

The generic name given to the a_{ij} coefficients of the constraint set of a linear-programming problem.

Telecommunication Networks

► Communications Networks
► Queueing Theory

Terminal

A location used by a carrier for freight consolidation, break-bulk, interchange, and shipment and vehicle service.

See

► Logistics and Supply Chain Management

The Institute of Management Sciences (TIMS)

Founded in 1953, The Institute of Management Sciences (TIMS) was an international organization for management science professionals and academics. It was merged with the Operations Research Society of America (ORSA) into the Institute for Operations Research and the Management Sciences (INFORMS) effective January 1, 1995. The objectives of TIMS were

(1) to identify, extend and unify scientific knowledge contributing to the understanding and practice of management, (2) to promote the development of the management sciences and the free interchange of information about the practice of management among managers, scientists, scholars, students, and practitioners of the management sciences within private and public institutions, (3) to promote the dissemination of information on such topics to the general public, and (4) to encourage and develop educational programs in the management sciences. TIMS published the journal *Management Sciences* (in 40 volumes) and other publications (some jointly with ORSA). It held national meetings (jointly with ORSA), sponsored meetings by its technical colleges and geographic sections, and held international meetings in various countries.

See

- ▶ Institute for Operations Research and the Management Sciences (INFORMS)
- ▶ Operations Research Society of America (ORSA)

Theorem of Alternatives

Many such theorems exist, with a typical one being: either $\boldsymbol{Ax} = \boldsymbol{b}$ has a solution or $\boldsymbol{yA} = \boldsymbol{0}$, $\boldsymbol{yb} \neq \boldsymbol{0}$ has a solution. They can be shown to be equivalent to the strong duality theorem of linear programming.

See

- ▶ Farkas' Lemma
- ▶ Gordan's Theorem
- ▶ Strong Duality Theorem
- ▶ Transposition Theorems

Theory of Constraints

Graham K. Rand
Lancaster University, Lancaster, UK

In the early 1980s, a novel was published which has subsequently been read all over the world by many executives, production planners and shop floor workers. *The Goal* sets out Eli Goldratt's ideas on how production should be planned (Goldratt and Cox 2004). The ideas were developed in the production planning system OPT (Optimized Production Technology) which was marketed by Creative Technology, Inc. (Rand 1990). These ideas were later broadened to encompass other areas such as marketing, distribution and project management in two further novels, *It's Not Luck* (Goldratt 1994) and *Critical Chain* (Goldratt 1997), and the theory widened to become the Theory of Constraints. In the novel, *Necessary but Not Sufficient* (Goldratt et al. 2000), set in the computer software industry, it is argued that although new technology may be necessary for major improvements, it is not sufficient. The theory has been applied to retailing through two further books, first by means of a conversation between Goldratt and his daughter, *The Choice* (Goldratt 2008), and in the novel, *Isn't it Obvious?* (Goldratt et al. 2009). Among the methods in his approach, Evaporating Clouds and Current Reality Tree have become widely used. Technical details are found in Goldratt (1990a, b).

See

- ▶ Production Management

References

Goldratt, E. M. (1990a). *What is this thing called the Theory of Constraints?* Great Barrington, MA: North River Press.
Goldratt, E. M. (1990b). *The haystack syndrome*. Great Barrington, MA: North River Press.
Goldratt, E. M. (1994). *It's not luck*. Great Barrington, MA: North River Press.
Goldratt, E. M. (1997). *Critical chain*. Great Barrington, MA: North River Press.
Goldratt, E. M. (2008). *The choice*. Great Barrington, MA: North River Press.
Goldratt, E.M., & Cox, J. (2004). *The goal* (3rd Rev. Ed.). Great Barrington, MA: North River Press.
Goldratt, E. M., Eshkoli, I., & Brownleer, J. (2009). *Isn't it obvious?* Great Barrington, MA: North River Press.
Goldratt, E. M., Schragenheim, E., & Ptak, C. A. (2000). *Necessary but not sufficient*. Great Barrington, MA: North River Press.
Rand, G. K. (1990). RP, JIT and OPT. In L. C. Hendry & R. W. Eglese (Eds.), *Operational research tutorial papers, 1990*. Birmingham: Operational Research Society.

Thickness

The minimum number of edge-disjoint planar subgraphs into which a graph can be decomposed.

See

▶ Graph Theory

Time Series Analysis

Christina M. Mastrangelo[1], James R. Simpson[2] and Douglas C. Montgomery[3]
[1]University of Virginia, Charlottesville, VA, USA
[2]Florida State University, Tallahassee, FL, USA
[3]Arizona State University, Tempe, AZ, USA

Introduction

A time series is an ordered sequence of observations. This ordering is usually through time, although other dimensions, such as spatial ordering, are sometimes encountered. A time series can be continuous, as when an electrical signal such as voltage is recorded. Typically, however, most industrial time series are observed and recorded at specific time intervals and are said to be discrete time series. If only one variable is observed, the time series is said to be univariate. However, some time series involve simultaneous observations on several variables. These are called multivariate time series.

There are three general objectives for studying time series: 1) understanding and modeling of the underlying mechanism that generates the time series, 2) prediction of future values, and 3) control of some system for which the time series is a performance measure. Examples of the third application occur frequently in industry. Almost all time series exhibit some structural dependency. That is, the successive observations are correlated over time, or autocorrelated. Special classes of statistical methods that take this autocorrelative structure into account are required.

Figure 1 shows examples of time series with distinctly different features. In Fig. 1a, the time series x_t appears to vary around a constant level. Such a time series is said to be stationary in the mean. In Fig. 1b, non-stationary behavior can be observed, i.e., the time series x_t drifts with no obvious fixed level. Some nonstationary time series may exhibit trends, or the variance of the series may increase as the level of the time series increases. Seasonal variation is illustrated in Fig. 1c.

The autocorrelation function is a very useful tool in characterizing time series behavior. The autocorrelation between x_t and x_{t+k} is defined as

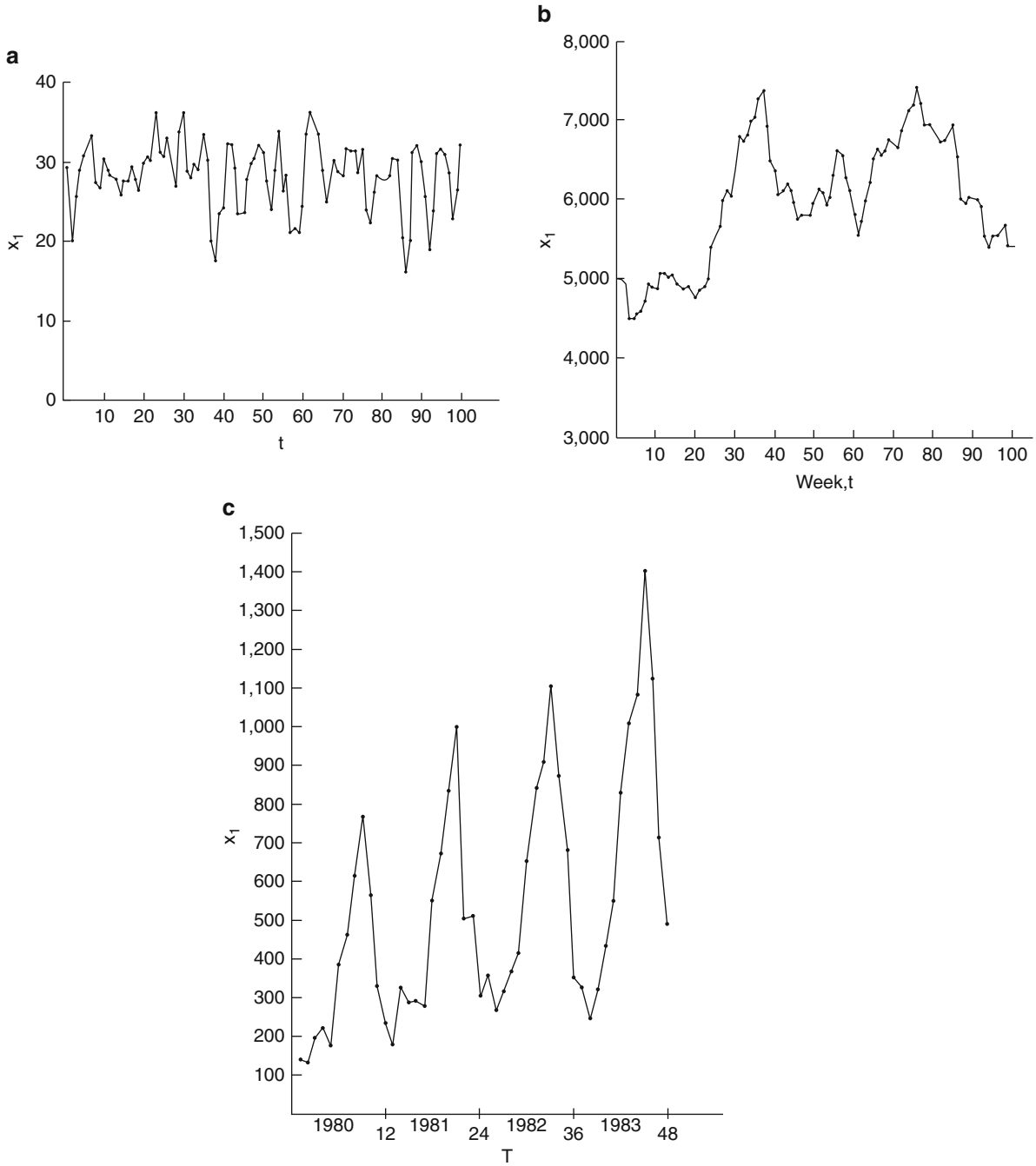
$$\rho_k = \frac{\mathrm{cov}(x_t, x_{t+k})}{\sqrt{V(x_t)V(x_{t+k})}} = \frac{\gamma_k}{\gamma_0}$$

where $\mathrm{cov}(x_t, x_{t+k}) = E[(x_t - m)(x_{t+k} - m)]$. This is called the autocorrelation at lag k. The usual estimate of ρ_k, $k = 1, 2, \ldots, K$, is the sample autocorrelation function

$$r_k = \frac{\hat{\gamma}_k}{\hat{\gamma}_0} = \frac{\sum_{t=1}^{n-k}(x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^{n}(x_t - \bar{x})^2}$$

Figure 2 shows the sample autocorrelation function for the time series in Fig. 1a. The dotted lines are two standard error limits. Notice that there is a large positive value or spike at lag 1 and the sample autocorrelation function decays as a damped sine wave from lag 1. The sample autocorrelation function is very useful in the identification of an appropriate time series model.

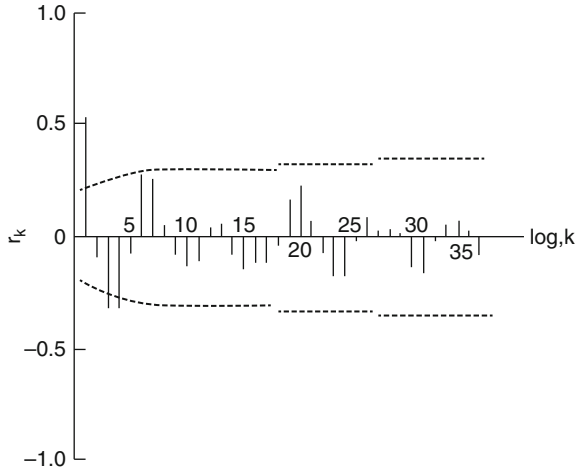
The partial autocorrelation function, denoted by ϕ_{kk}, is also useful in the identification process. It can be interpreted as the simple correlation between two random variables x_t and x_{t-k} after adjusting for the intermediate variables x_{t-1}, $x_{t-2}, \ldots, x_{t-k+1}$. Once the sample autocorrelation and partial autocorrelation functions are estimated, they may be plotted. A tentative model is then identified by comparing the observed patterns with the theoretical function patterns. For an autoregressive process of order p, ϕ is nonzero when k is less than or equal to p and greater than zero for k greater than p. In other words, while



Time Series Analysis, Fig. 1 (**a**) Viscosity of a chemical product. (**b**) Demand for a plastic container. (**c**) Monthly demand for a 48-oz soft drink in hundreds of cases

the autocorrelation function of an autoregressive process decays in an exponential fashion, the partial autocorrelation function cuts off after lag p.

In addition, the inverse autocorrelation function and the extended sample autocorrelation function are useful in time series model identification. See Fuller (1996), Montgomery, Johnson, and Gardiner (1990), Cleveland (1972), and Abraham and Ledolter (1983) for definitions of these functions and more details.



Time Series Analysis, Fig. 2 Sample autocorrelation function

Time Series Modeling Methods

There are several widely-used approaches for modeling and analysis of time series data. Regression methods play a fundamental role. If y_t represents the time series of interest and $x_{jt}, j = 1, 2, \ldots, k$ are a collection of other time series thought to be related to y_t, then it is possible to fit a regression model of the form

$$y_t = \beta_0 + \sum_{t=1}^{n} \beta_j x_{jt} + \varepsilon_t, \qquad t = 1, \ldots, n$$

using least squares or some suitable variation. Usually, however, the errors e_t are autocorrelated and more complex estimation schemes are needed. Several estimation methods are available which result in estimates similar to least squares estimates, but the standard errors may be very different. Yule-Walker estimation uses the Yule-Walker equations to estimate the autoregressive parameters of the errors and generalized least squares to estimate $\boldsymbol{\beta}$. Harvey (1990) gives a full description of this and other methods.

Smoothing methods are frequently used in time series analysis. In particular, exponential smoothing is widely used for producing short-term forecasts of many types of industrial time series. Much of the original work in this area is by Brown (1962), Holt (1957), and Winters (1960). Exponential smoothing is often developed heuristically starting with a simple model such as $x_t = b + e_t$, where e_t are independent random variables and b is an unknown constant. Simple or first-order exponential smoothing is defined as

$$S_t = \alpha x_t + (1 - \alpha) S_{t-1}$$

where $0 \leq \alpha \leq 1$. The smoothed statistic S_t estimates the constant b, so the forecast for any future observation $X_{t+\tau}$ made at the end of period t is

$$\hat{x}_{t+\tau}(t) = S_t$$

Extensions of this methodology to forecasting linear and quadratic trend and incorporating seasonal behavior are described in Montgomery, Johnson and Gardiner (1990). Goodman (1974) and Cogger (1974) showed that exponential smoothing for a kth order polynomial results in forecasts that are optimal in a mean square error sense for certain classes of non-stationary time series. McKenzie (1978) extended these results to models that may include transcendental terms.

The class of autoregressive integrated moving averages (ARIMA) models proposed by Box, Jenkins and Reinsel (2008) and Jenkins (1979) have been very successful for time series modeling and forecasting. The general form for this family of models is

$$(1 - \phi_1 \beta - \phi_2 B^2 - \cdots - \phi_p B^p)(1 - B)^d x_t = \theta_0 + (1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q)\varepsilon_t$$

where ϕ_i are the autoregressive parameters, θ_j are the moving average parameters, B is a backshift operator defined such that $B^r x_t = x_{t-r}$,$(1 - B)^d = \Delta^d$ is the backward difference operator, and $\in_t$ is an uncorrelated sequence of random disturbances with mean zero and variance σ^2. This model can also be extended to incorporate seasonal behavior (see Box et al. 2008; Montgomery et al. 1990). One chooses a model by specifying the integers p, d, and q, resulting in an ARIMA(p, d, q) model. This is usually done by examining the sample autocorrelation and partial autocorrelation function. For example, if the sample autocorrelation function decays as a damped sine wave and the partial autocorrelation function has large spikes only at lags 1 and 2, a tentative ARIMA model estimation with $p = 2$ and $q = 0$ might be considered.

Nonlinear regression methods are used to estimate the parameters ϕ_i and θ_j. The approach requires initial point estimates of the parameters and then uses an iterative search technique to minimize the residual sums of squares. Most computer packages implement a modification of the Gauss-Newton method suggested by Marquardt (1963). The Gauss-Newton method first linearizes the nonlinear function with a Taylor series expansion and then iterates to find improved parameter estimates. Unfortunately, the original Gauss-Newton approach will not always converge. So Marquardt proposed a modified search procedure that adds a small bias to the parameter estimates to ensure convergence to the minimum residual sums of squares. Computer packages provide reasonable initial point estimates making the estimation routine transparent to the user.

Finally, the residuals from the fitted model are studied to test model adequacy. Generally, one should examine the autocorrelation function of the residuals, for if the model is adequate, the residuals should be approximately uncorrelated. The tests on residual autocorrelations suggested by Box and Pierce (1970) and Ljung and Box (1978) are useful in this regard. Residual plots, such as a plot of residuals verses the fitted x_t, and a normal probability plot of the residuals, are useful in detecting model inadequacy. Thus model estimation is typically iterative involving cycles of tentative model identification, estimation, and residual analysis.

To illustrate, consider the container demand data from Fig. 1b. It can be shown that an appropriate choice of p, d, and q is $p = 0$, $d = 1$, and $q = 1$, resulting in the ARIMA(0,1,1) = IMA(1,1) model

$$(1 - B)x_t = (1 - \theta B)\varepsilon_t.$$

The least squares estimate of the parameter θ in this model is $\bar{\theta} = -0.70$. Therefore, the final model is

$$x_t = x_{t-1} + \varepsilon_t + 0.7\varepsilon_{t-1}.$$

This model is satisfactory with respect to the adequacy criteria cited above.

Forecasting

An important objective of any time series model is forecasting future values. The term forecasting is used in the time series analysis literature although most results are based on the general theory of linear prediction developed by Kalman (1960), Whittle (1963), Box, Jenkins and Reinsel (2008), and many others. The objective is to produce minimum mean square error forecasts.

Minimum mean square error forecasts for ARIMA models are obtained by taking the conditional expectation $E(X_{t+\tau}|X_t, X_{t-1}, \ldots)$. For example, the minimum mean square error forecast for the ARIMA (0,1,1) = IMA(1,1) model shown earlier for the container data is

$$E(x_{t+\tau}|x_t, x_{t-1}, \ldots) \equiv \hat{x}_{t+\tau}(t) = x_t + 0.7\varepsilon_t \quad (1)$$

where $e_t(1) = x_t - \hat{x}_t(t-1)$ is the one-step ahead forecast error. Figure 3 shows the forecasts obtained from this model. It is usually necessary to provide prediction intervals for forecasts as well as point estimates. Figure 3 shows the 50% and 95% prediction limits for the forecast of future container demand. For details of the construction of these limits, see Box, Jenkins and Reinsel (2008) and Montgomery, Johnson and Gardiner (1990).

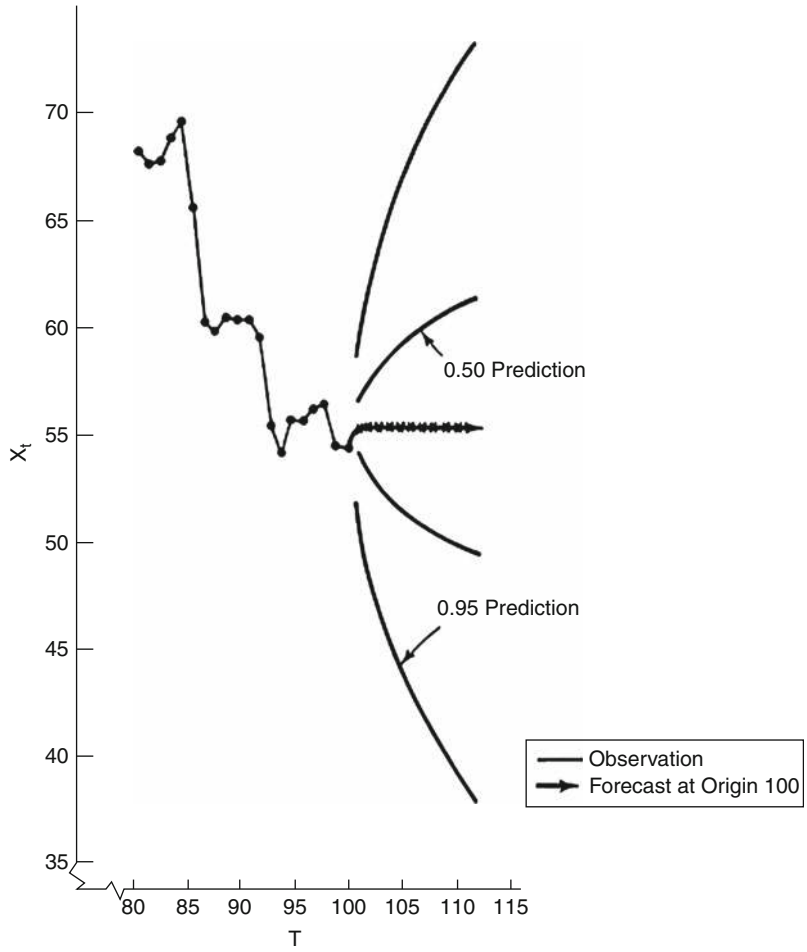
Forecasts from ARIMA models are equivalent to forecasts produced by other methods in certain cases. For example, the forecasts from an IMA(1,1) model, such as that given above for the container demand data, are identical to those produced by simple first-order exponential smoothing. Other relationships between exponential smoothing and ARIMA models are given by Box, Jenkins and Reinsel (2008) and Pandit and Wu (1974).

The form of the eventual forecast function for ARIMA models is also of interest, because it leads in some cases to efficient methods for forecast generation and updating. The form of the forecast function or several common ARIMA models is given in Box, Jenkins and Reinsel (2008).

Transfer Functions and Related Topics

If y_t and x_t are two stationary time series related through the mean filter

$$y_t = V(B)x_t + \varepsilon_t$$



Time Series Analysis, Fig. 3 Forecast of plastic container demand at origin 100, with 0.50 and 0.95 percent prediction limits

then $V(B) = \sum_{j=-\infty}^{\infty} v_j B^j$ is called the transfer function of the filter and e_t is called the noise series of the system. Typically, x_t and e_t are assumed to follow ARMA = ARIMA(p,0,q) models. It is customary to write

$$V(B) = \frac{\omega_S(B)B^b}{\delta_r(B)}$$

where $\omega_S(B) - \omega_0 - \omega_1 B - \omega_2 B^2 - \ldots - \omega_S B^S$, $\delta_r(B) - \delta_0 - \delta_1 B - \delta_2 B^2 - \ldots - \delta_r B^r$ and b is a delay representing the time before the input at time t produced an effect on the output. A transfer function model is identified by choosing appropriate values of s, r, b, and a model for the noise e_t. Usually s, r, and b will be no larger than 2. The cross-correlation function is useful in model identification.

Once a suitable transfer function model is identified, the parameters are estimated by nonlinear regression methods, and diagnostics checks are applied, much like in classical univariate ARIMA modeling. Minimum mean square error forecasts are generated using a similar approach, based on conditional expectation at time t of $y_{t\tau}$. For detailed examples of identification, estimation, diagnostic checking, and forecasting with transfer functions, see Box, Jenkins and Reinsel (2008) and Montgomery, Johnson and Gardiner (1990). The latter authors presented an example showing that for relatively short forecast lead times, the forecasts from a transfer function model will usually be superior to those produced by a univariate ARIMA model.

An important special case of the transfer function occurs when the input series x_t is a sequence of indicator variables that represent the occurrence

of identifiable, unique events that are thought to influence the output y_t. These events are called interventions and the resulting models are called intervention models. An intervention model is often used to provide a statistical basis for concluding that the identifiable event has resulted in a change in the time series.

Box and Tiao (1975) developed the basic intervention analysis methodology and applied it to photo chemical pollution data from the Los Angeles basin. They showed that the opening of the Golden State Freeway and the adoption of a new law, that reduced the proportion of reactive hydrocarbons in local gasoline, reduced ozone levels, and that required changes in automobile engines reduced ozone levels only in warm weather months. Other intervention studies were reported by Montgomery and Weatherby (1980) and Wichern and Jones (1977).

Intervention models are also useful in the study of time series outliers. Fox (1972) proposed two types of outliers, additive and innovational. Other useful references on this topic are Tsay (1986) and Chang, Tiao and Chen (1988).

In some time series problems, one observes m different variables x_{1t}, x_{2t},..., x_{mt} in a multivariate framework. One way to model this structure is with a multivariate ARIMA model of the form

$$\Phi_p(B)X_t = \Theta_q(B)\varepsilon_t$$

where $x_r' = [x_{1t}, x_{2t}, \ldots, x_{mt}]$, Φ_p (B) and Θ_q (B) are matrix polynomials of autoregressive and moving average parameters, respectively, and ε_t is a sequence of independent multivariate random vectors each with mean zero and covariance matrix $\sum$. These are sometimes called vector time series models. Basic references for these models include Jenkins (1979), Granger and Newbold (1977), and Hannan (1970). The state space modeling approach is also useful for representing multiple series. See Hannan (1970) and Akaike (1976) for a complete description of state space modeling.

Computing

A number of software packages perform the time series modeling and forecasting functions previously described, including some spreadsheet statistical analysis add-ins. The two high-end software support tools commonly used by researchers and practitioners are SAS and S-Plus. Both programs provide a wide range of modeling options including various smoothing alternatives and extensive ARIMA modeling features. SAS is also capable of developing transfer function and intervention models. S-Plus provides the capability to model time series in the presence of outliers. More advanced procedures are also available from SAS and S-Plus. Several other PC-based software programs, including MINITAB, STATGRAPHICS, R, JMP, Autobox, and EViews, provide high-quality time series modeling and forecasting support. For ARIMA modeling, the software programs provide the plots, nonlinear estimation, and forecasting tools necessary to develop successful models.

See

- ▶ Exponential Smoothing
- ▶ Forecasting
- ▶ Quality Control
- ▶ Regression Analysis

References

Abraham, B., & Ledolter, J. (1983). *Statistical methods for forecasting*. New York: John Wiley.

Akaike, H. (1976). Canonical correlations analysis of time series and the use of an information criterion. In R. Mehra & D. G. Lainiotis (Eds.), *Advances and case studies in system identification*. New York: Academic Press.

Box, G. E. P., & Pierce, D. A. (1970). Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. *Journal of American Statistical Association, 64*.

Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (2008). *Time series analysis, forecasting and control* (4th ed.). New York: Wiley.

Box, G. E. P., & Tiao, G. C. (1975). Intervention analysis with applications to economic and environmental problems. *Journal of the American Statistical Association, 70*, 70–79.

Brown, R. G. (1962). *Smoothing, forecasting and prediction of discrete time series*. Englewood Cliffs, NJ: Prentice-Hall.

Chang, I., Tiao, G. C., & Chen, C. (1988). Estimations of time series parameters in the presence of outliers. *Technometrics, 30*, 193–204.

Cleveland, W. S. (1972). The inverse autocorrelations of a time series and their applications. *Technometrics, 14*, 277–293.

Cogger, K. O. (1974). The optimality of general-order exponential smoothing. *Operations Research, 22*, 858–867.

Fox, A. J. (1972). Outliers in time series. *Journal of the Royal Statistical Society, Series B, 43*, 350–363.
Fuller, W. A. (1996). *Introduction to statistical time series*. New York: John Wiley.
Goodman, J. L. (1974). A new look at higher-order exponential smoothing for forecasting. *Operations Research, 22*, 880–888.
Granger, G. W. C., & Newbold, P. (1977). *Forecasting economic time series*. New York: Academic Press.
Hanan, E. J. (1970). *Multiple time series*. New York: John Wiley.
Harvey, A. C. (1990). *The econometric analysis of time series* (2nd ed.). Cambridge, MA: MIT Press.
Holt, C. C. (1957). *Forecasting trends and seasonal by exponentially weighted moving averages*. ONR Memorandum No. 52, Carnegie Institute of Technology.
Jenkins, G. M. (1979). *Practical experiences with modeling and forecasting time series*. Lancaster, England: GJM Publications.
Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering for Industry, Series D, 82*, 35–45.
Ljung, G. M., & Box, G. E. P. (1978). On a measure of lack of fit in time series models. *Biometrika, 65*, 297–303.
Marquardt, D. W. (1963). An algorithm for least squares estimation of nonlinear parameters. *Journal of the Society of Industrial and Applied Mathematics, 2*, 431–441.
McKenzie, E. (1978). The monitoring of exponentially weighted forecasts. *Journalof the Operational Research Society, 29*.
Montgomery, D. C., Johnson, L. A., & Gardiner, J. S. (1990). *Forecasting and time series analysis* (2nd ed.). New York: McGraw-Hill.
Montgomery, D. C., & Weatherby, G. (1980). Modeling and forecasting time series using transfer function and intervention methods. *AIIE Transactions, 12*, 289–307.
Pandit, S. M., & Wu, S. M. (1974). Exponential smoothing as a special case of a linear stochastic system. *Operations Research, 22*, 868–879.
Tsay, R. S. (1986). Nonlinearity tests for time series. *Biometrika, 73*, 461–466.
Whittle, P. (1963). *Prediction and regulation by linear least-square methods*. Princeton, NJ: Van Nostrand.
Wichern, D. W., & Jones, R. H. (1977). Assessing the input of market disturbances using intervention analysis. *Management Science, 21*, 329–337.
Winters, P. R. (1960). Forecasting sales by exponentially weighted moving averages. *Operations Research, 22*, 858–867.

Time/Cost Trade-offs

An approach to scheduling where the project duration is shortened with a minimum of added costs.

See

► Network Planning

Time-stepped Simulation

A computer model in which time is incremented by a simulated clock. Each appropriate function is recomputed after the clock is incremented in a cyclic manner. A model may be linearly coded and entirely time-stepped or an event-driven simulation may use time-stepping for some critical function with a cycle of sub-functions.

See

► Event-driven Simulation
► Simulation of Stochastic Discrete-Event Systems

Timetabling

Michael W. Carter
University of Toronto, Toronto, Ontario, Canada

Introduction

Most dictionaries do not include the word timetabling as a single word. It is often listed as either two words (time table) or hyphenated (as time-table). The Oxford English Dictionary defines a timetable as:

> A tabular list or schedule of times at which successive things are to be done or happen, or of the times occupied in the parts of some process. spec. **a.** A printed table or book of tables showing the times of arrival and departure of railway trains at and from the stations; also a similar table of times of arrival and departure of passenger boats or other public conveyances. **b.** A chart used in railway traffic offices, showing by means of cross lines, in one direction representing hours and minutes and in the other miles, the position of the various trains at any given moment. **c.** A time-sheet on which a record is kept of the time worked by each employee. **d.** A table showing how the schedule of a school or other educational institution, for any day, or for a week, is allotted to the various classes and subjects. **e.** Mus. A table of notes showing their relative time value.

The Oxford dictionary also defines the verb time-table as "To schedule, to plan or arrange according to a timetable, to include in a timetable. Hence time-tabled and timetabling."

Professor Anthony Wren, at the first Practice and Theory of Automated Timetabling (PATAT) conference in Edinburgh, 1995, defined timetabling as "the allocation, subject to constraints, of resources to objects being placed in space-time, in such a way as to satisfy as nearly as possible a set of desirable objectives. Examples are class and examination time-tabling and some forms of personnel allocation, for example manning of toll booths subject to a given number of personnel." In the latter case, the process is defined in terms of developing timetables for each individual employee.

In other words: timetabling involves deciding when events/activities will take place in time; but it does not involve assigning resources to those activities. For example, a bus timetable for a particular metropolitan bus route may require "one bus to leave the main terminal every 30 minutes between 6:00 am and 11:00 p.m.; and every 10 minutes during rush hours 7:00 am to 9:00 am and 4:00 p.m. to 6:00 p.m.". The time table does not specify which buses or drivers should be allocated to each trip. In course timetabling, the objective is to decide what day and time each section of each course should be held. It does not specify which students will be assigned to each section.

Normally, when one sees the word timetabling in an operations research context, people are referring to problems relating to timetabling of courses or examinations in a school. Furthermore, it refers to the concept of developing algorithms, usually computer programs, for the automatic construction of time-tables. There are a number of other related problems in timetabling which will be described; but they are often referred to under different titles. As described by McCollum and Burke (2010) in the Preface to the Proceedings for PATAT 2010, "computer-aided timetable generation ... includes personnel rostering, school timetabling, sports scheduling, transportation timetabling and university timetabling."

Timetabling can also be described as a subset of the larger discipline called scheduling. One can define scheduling as the more general problem of determining the times for activities and assigning the necessary resources. In some cases, for example in Sports Time tabling, once it is decided when a match will occur between a pair of teams, (and who the home team is), all major resources have already implicitly been assigned (the two teams and the stadium). Hence Sports Time tabling is commonly referred to as Sports Scheduling. In this case, the terms are justifiably interchangeable.

It will be frequently distinguished between feasibility and optimality. A feasible solution is any solution that satisfies all of the constraints. An optimal solution is the (possibly unique) solution among all feasible answers which maximizes (minimizes) some objective function. In some timetabling problems, it is sufficient to find a feasible solution.

Examination Timetabling

Examination Timetabling is the simplest timetabling problem to describe, although it is not always easy to solve. The basic problem is to assign examinations to a limited number of available periods in such a way that there are no conflicts or clashes. That is, no student is required to write two examinations at the same time. The problem is closely related to the graph coloring problem. Each examination is represented by a node. Two nodes are connected by an edge if there is at least one student who is required to write the two corresponding exams. The graph coloring problem asks the question: Can the nodes of this graph be colored using p colors such that no two nodes with the same color are connected by an edge? If each color represents an examination period, and if p is the number of periods available, then coloring the graph is equivalent to finding a conflict free assignment of exams to the available periods.

In practice, the basic feasibility issue may be the critical problem. In particular, for any given problem instance, there is a minimum number of periods required to allow a feasible solution. In graph theory terminology, this is called the chromatic number of a graph. If the number of periods provided is close to the theoretical minimum, then you need an algorithm that concentrates on finding a feasible solution. There has been considerable research on good coloring algorithms. Given plenty of periods, it is easy to find a conflict free timetable. The coloring problem is trivial, and efforts can be focused on searching for a good answer using some secondary objectives. Without enough periods, it is not possible to find a feasible solution, and the objective must be changed to something like minimize the number of student conflicts.

The most common secondary objective is to try to spread each student's exams as evenly as possible. Each institution will impose a variety of additional constraints on the basic model such as:

- Some exams may have precedence constraints (e.g., "exam A must precede exam B");
- Some exams must be consecutive (e.g., "exam C must immediately precede exam D");
- Some exams are excluded from certain periods;
- Limited available rooms and/or seats; and
- There may be special resource requirements.

For a more comprehensive description of the exam timetabling problem and a survey of practical approaches, refer to Carter (1986), and Qu et al. (2009).

School Timetabling

Class-Teacher timetabling is normally associated with high schools or elementary level schools where the students are grouped into a set of classes and each class has a set of courses that it must take. Professor Dominique de Werra (1985) defines the basic class-teacher model in the following terms. Let $C = \{c_1, c_2, \ldots, c_m\}$ be a set of classes and $T = \{t_1, t_2, \ldots, t_n\}$ be a set of teachers. An $n \times m$ requirements matrix, $R = \{r_{ij}\}$ is given where r_{ij} is the required number of times class c_i must meet with teacher t_j. In the basic model, it is assumed that all lectures are the same length (say one period). Given a set of p periods, the problem is to assign each meeting to some period such that no teacher (and no class) is involved in more than one meeting at a time. The basic problem has no objective function, so the issue is simply to find a feasible solution.

It can be shown that this problem is easy to solve (in the computational complexity sense) in that there exists a polynomial algorithm to find a solution (using a matching algorithm) under the simple and obvious conditions that no teacher (or class) is required to attend more than p periods. The problem remains easy if the basic model is extended to include assigning meetings over a week, where limits are imposed on the number of times each class-teacher pair can meet on any one day.

Unfortunately, most practical problems will have a few extra conditions, and the problem quickly becomes computationally intractable (NP-Complete). For example, if it is assumed that some of the teachers (and/or classes) are not available in every period, then the problem is no longer easy. This is also true if the teachers and classes are available every period, but some of the meetings have been preassigned to specific periods. Another common complication is that some meetings are for more than one period. For example, some meetings may require two or three consecutive periods.

The problem is also often complicated by adding room availability constraints. For example, there may be certain meetings (science, physical education, music, etc.) which require specific rooms. This problem can be expressed using a three dimensional requirements matrix that specifies the number of meetings between class i and teacher j in a room of type k, where there are a limited number of each type of room. This problem is also NP-Complete. Refer to Kingston (2008) for more details.

Course Timetabling

Course timetabling is normally associated with universities, and involves the assignment of sections of courses (lectures, laboratories, tutorials, seminars, etc.) to specific days of the week and times of day. In the course-timetabling problem, unlike the class concept, each student selects a set of courses personally tailored to their own needs. (In practice, many students will have very similar selection patterns.) The primary objective is often to find a timetable that minimizes the expected number of student conflicts.

Strictly speaking, based on the definition given here, course timetabling does not include the assignment of resources (teachers, rooms, special equipment, or even students). In many practical instances, most teachers will be assigned to teach specific course sections before timetabling, while rooms, special equipment and students are assigned after time-tabling. In large schools, many of the courses will be offered in more than one section. Students must be divided up into (roughly equal) groups and assigned to separate sections. This problem is referred to as sectioning or student scheduling. Some packages have been designed to attack all of these problems simultaneously. However, due to the large number of variables involved, most practical methods approach the

problems sequentially. The basic course-timetabling problem will be described here. The interested reader can refer to Lewis (2008) for a more detailed discussion of each of the subproblems, and references to practical applications.

The basic course-timetabling problem usually includes a number of side constraints. Courses and course sections should be spread in a particular way throughout the week. For example, an institution may require that all sections of the same course be timetabled at the same times. A course may be divided into multiple meetings (two or three times per week), and there may be restrictions on the meeting patterns that can be used (e.g., Mon., Wed., Fri. at 9:00 a.m.). Some schedules include an allowance for lunch periods, travel time between classes, and the number of hours per day for students and teachers.

In practice, there are two main variations of the course-timetabling problem: the master timetable approach, and the demand driven system. Practitioners typically feel very strongly about their preference for one or the other. Under a master-time-tabling system, the institution will first create a course timetable, and then students register for courses (after consulting a list that describes when each class is offered). The term master timetable refers to the common practice of starting this year's timetable based on the previous year, and making any required changes based on revisions to course offerings. With a demand driven timetable, the institution posts a list of (proposed) course offerings without any times, and students pre-register for courses before timetabling is performed.

The main advantage of a demand driven system is that the timetable can be constructed using actual student course requests. With a master-timetable system, the timetable must be developed without knowing what the students really want or need. Individual department timetable representatives try to build a timetable that will work for students in their own program in each year. This is very difficult unless the programs are highly structured. In more flexible environments, students often have difficulty selecting the credits that they need without conflicts. A major problem in the U.S. today is that students in many institutions find it impossible to complete their program in the nominal program length due to timetable issues.

There are several disadvantages of a demand-driven system. It requires additional data collection effort, since students must pre-register for courses (typically 4–5 months before term starts) and then, when they get the results of their requests, they start making changes in a second round. In a master-timetabling system, students should be able to construct a conflict free timetable on the first attempt. A demand-driven system also puts fairly tight time constraints on the timetabling process. In a master timetable system, the institution can construct the timetable a year in advance, and some schools publish the times in the course calendar. In a demand-driven system, the students submit course requests a few months before the term starts, and all of the timetabling activity is compressed.

One of the curious issues in the timetabling problem creates a bit of a paradox in the demand-driven system when courses are taught in multiple sections. You cannot assign students to sections (conflict-free) until you have timetabled the sections; but, you cannot timetable the sections until you know which students are in each section. One solution is to assign students to a specific section in advance of timetabling, for the purpose of finding good times. These assignments can be re-evaluated in the student scheduling phase at the end.

Anyone interested in timetabling should refer to the Web site maintained by the University of Nottingham, on automated scheduling, optimisation and planning.

There are a number of other (less common) problems that share the basic timetabling structure. Sports timetabling is the problem of trying to find a rotation for a set of teams such that each team can play every other team twice (once at home and once away). If there are no side constraints, there are some elegant solutions related to tournaments, including a mathematical construction based on permutations (see survey by Kendall et al. 2010). There has also been some research on Employee Timetabling/Rostering, where you want to determine shift work patterns for employees in order to meet a given demand pattern. A particular well-studied variation on this problem is the nurse-rostering problem (see review by Burke et al. 2004).

See

► Computational Complexity
► Graph Theory

- ▶ Higher Education
- ▶ Scheduling and Sequencing
- ▶ Sports

References

Burke, E. K., De Causmaecker, P., Vanden Berghe, G., & Van Landeghem, H. (2004). The state of the art of nurse rostering. *Journal of Scheduling, 7*, 441–499.

Carter, M. W. (1986). A survey of practical applications of examination timetabling algorithms. *Operations Research, 34*, 193–202.

de Werra, D. (1985). An introduction to timetabling. European *Journal of Operational Research, 19*, 151–162.

Kendall, G., Knust, S., Ribeiro, C. C., & Urrutia, S. (2010). Scheduling in sports: An annotated bibliography. *Computers and Operations Research, 37*, 1–19.

Kingston, J. H. (2008). Resource assignment in high school timetabling. *Proceedings of the Seventh International Conference on the Practice and Theory of Automated Timetabling*, August 2008.

Lewis, R. (2008). A survey of metaheuristic-based techniques for university timetabling problems. *OR Spectrum, 30*, 167–190.

McCollum, B., & Burke, E. (2010). Preface. *Proceedings of the 8th International Conference on the Practice and Theory of Automated Timetabling*, 10–13 August 2010 (Queen's University of Belfast).

Oxford English Dictionary, On-line edition as of March 2011.

Qu, R., Burke, E. K., McCollum, B., Merlot, L. T. G., & Lee, S. Y. (2009). A survey of search methodologies and automated system development for examination timetabling. *Journal of Scheduling, 12*, 55–89.

Wren, A. (1996). Scheduling, timetabling and rostering – A special relationship? In Burke & Ross (Eds.), *Practice and theory of automated timetabling: Vol. 1153. Lecture notes in computer science*. Springer.

TIMS

- ▶ The Institute of Management Sciences (TIMS)

Tolerance Analysis

A sensitivity analysis procedure applied to a linear-programming problem that allows for simultaneous changes of the objective function cost coefficients and/or right-hand-sides of the constraints.

See

- ▶ Hundred Percent Rule
- ▶ Sensitivity Analysis

Total Float

The amount of time a project work time can be delayed without affecting the duration of the project. Total float can be used in only one activity in a path. If no schedule times are specified for starting and finishing the various activities, then the float is calculated as the difference between the latest start time and the earliest start time, or the difference between the latest finish time and the earliest finish time. Float can be positive, negative or zero.

See

- ▶ Network Planning

Total Quality Management

John S. Ramberg
Pagosa Springs, CO, USA

Introduction

During the decade of the 1980s, U.S. corporations recognized the quality achievements of their Japanese counterparts and began to understand the messages being delivered by Deming, Juran and others on the importance of quality (Deming 2000; Defeo and Juran 2010). They devised methods for obtaining, understanding and communicating customer needs and requirements within their organizations, developed strategies for improving their engineering design, development, manufacturing and delivery processes, and created new corporate cultures that included the formation of self-directed working groups and encouragement of employee participation. Through this focus on quality and the development and adaptation of techniques for achieving customer

satisfaction, some of these corporations have demonstrated improvement in achieving high quality, timely deliveries at low costs and ultimately improved their business performance. Many of these firms called this new management and operations philosophy Total Quality Management or TQM.

At the outset, many TQM programs were simply copies of Japanese efforts. As cultural differences between the Japanese and western world were better understood, and as other quality contributions were recognized, many U.S. firms developed their own unique quality programs. See Prybutok and Zhang (2010) and Vol. 4 of *Quality Management Journal* for health care agency examples.

Other firms, frustrated by false starts and questionable implementations, began to question the value of total quality management, and some have given up, regarding it as just another fad (Senge 1993). In many of these latter situations, quality efforts have been misdirected or unfocused. In some cases, quality improvement activities were simply knee-jerk reactions to the customers who complained most vehemently to the highest level of the organization. Ramberg (1994) described some of the scurrilous characters who proclaim TQM, while delivering just another program; he raised the question, "TQM: Thought Revolution or Trojan Horse." Three decades later, many organizations, especially nonprofits and governmental, are still not aware of total quality management.

While TQM connotes much more than simply the three words total, quality and management, nevertheless, definitions of each of the three words seem an appropriate place to begin. A typical dictionary definition of total is: all or whole, that is constituting the whole; complete. The definition of quality is a bit more difficult to comprehend as U.S. firms have come to understand. A formal definition, as given by the American Society for Quality (ASQ). "The totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs." Finally, management is the act, be it a science, an art or manner, of planning, directing, organizing and controlling a firm's decisions and actions. As an aside, it is interesting to note that the phrase "to manage" originated as "to train (a horse) in his paces, or to cause to do the exercises of the manage (Merriam-Webster, 2004)!"

A Profound Understanding of Quality

Quality is the pivotal word in TQM. A fundamental reason for the U.S. losing world leadership in manufacturing during the 1960s and 1970s was its lack of a profound understanding of the Q word. The gurus of quality, in the interest of developing a better understanding of the importance of quality, created shorter, more explicit, operationally oriented definitions such as "fitness for use" (Juran, 1988), "conformance to specifications" (Crosby 1989), "long term loss to society" (Taguchi 1986), and "a predictable degree of uniformity and dependability, at a low cost and suited to the market" (Deming, as paraphrased by Gitlow, Oppenheim, and Oppenheim, 1995).

Some have made light of the differences in these operational definitions of quality. A few have concluded that even the quality gurus cannot agree on the definition of quality. They should be viewed as being complementary, each definition emphasizing its definer's experience base in relation to the customer in question. "Fitness for use" is an appropriate operational definition of quality in the creation and marketing of a product or service on the production floor, where an employee may be far removed from the customer, the translation of quality performance measures into specific dimensions having specified targets and specification limits seems a necessity. Finally, if "loss to society" is thought of as "long-term business loss," then its relation to the other two operational definitions becomes clearer. Deming's definition exhibits his emphasis on variability and its reduction as a fundamental step in improving quality.

A first step in attaining a profound understanding of quality is the realization that it is customer-driven. It not only begins with the customer, in the end, it is judged by the customer. While the "voice of the customer" is imperative, a customer may not be able to fully articulate his needs and desires. Even the most sophisticated customers are not likely to be able to envision all of the characteristics of a product that will satisfy and "delight" them. Expert panels can serve an important role, but they too have their limitations. Obtaining this information is a complex task. Based on this input, product creators, developers and deliverers must envision these dimensions of quality that will satisfy and delight their customer.

Furthermore, they must maintain a dialog with the customer so that they will continue to understand and respond to this dynamic "voice of the customer."

Traditionally managers have viewed quality and cost as a zero-sum game. That is, any improvement in quality will occur only at a substantial additional cost. The following quote from Vaughn (1967) illustrates this "conventional wisdom:" "The trade off is between the effects of less emphasis on quality and the cost of more of it–." The following is a counter example. A citizens' group discovered that their water company was losing 40% of its treated water, prior to delivery. This meant that they were treating 167% more water than demanded, and hence 167% additional treatment facilities were required. Rather than making any attempt to reduce losses, the leadership had committed its users to millions of dollars of debt for a land purchase to build an un-needed reservoir.

Juran (1988) categorized quality associated costs (and estimates of the associated percentage for one industry) into four broad groups, those due to Internal Failures (30%), External Failures (40%), Appraisal (25%), and Detection and Prevention (5%). He also discussed how these percentages are dependent upon the maturity of the product line and effort expended on quality improvement. Juran's classic model for optimum quality levels also emphasized that there is a tradeoff between quality and cost. He stated that "failure costs decline until they are over-taken by the increasing costs associated with appraisal and prevention. At this point total costs increase." Juran also made clear the cost of quality through the phrase, hidden factory, where he exhibited the additional resources necessary to deliver products and services.

Cole (1992) made an excellent case for a fundamental paradigm shift regarding quality and costs and timeliness, based on the achievements of the Japanese. His conclusions are given in Table 1. Compare Cole's views with the old quality paradigm, "you get what you pay for." The truth is that high cost, alone, is not a guarantee that a product will be of high quality. Indeed, some times the contrary is true, resulting in what *Consumer Reports* refers to as "best buys." The six achievements of the Japanese, cited by Cole, have an important impact on conclusions drawn from quality cost models. Specifically, they indicate that the point at which it is no longer cost effective to improve quality is at a much lower defective rate than previously thought.

Total Quality Management, Table 1 Cole's underlying reasons for Japanese achievements in quality

"-realized that the costs of poor quality were far larger than had been recognized."
"-recognized that focusing on quality improvement as a firm-wide effort improved a wide range of performance measures."
"-established a system that moved toward quality improvement and toward low-cost solutions simultaneously."
"-focused on preventing error at the source, thereby dramatically reducing appraisal costs."
"-shifted the focus of quality improvement from product attributes to operational procedures."
"-evolved a dynamic model in which customer demands for quality rise (along with their willingness to pay for these improvements)."

Juran (1988) noted that many disagreements about achieving quality result from the fact that there are two fundamentally different quality issues, one income oriented and the other cost oriented. Features that produce customer satisfaction are income oriented. They are the key to attracting new customers and through satisfaction of retaining them. Cost oriented quality issues are the defects and failures that incur. They cause dissatisfaction and the loss of customers. As customers become aware of a product and indeed a producers track record through publications such as *Consumer Reports*, they also impact the ability of attracting and retaining customers. Furthermore, they impact the profitability of the firm through the dollars lost internally in defectives and rework and externally through warranty costs and other required services.

Establishing, appraising or judging the quality of a product are far more difficult than simply defining it. In his highly acclaimed book *Management of Quality*, Garvin (1988) elaborated eight dimensions of quality, including performance, features, reliability, conformance, durability, serviceability, aesthetics, and perceived quality. Through his study on air conditioners, he illustrated the differences in the perception of quality of various constituencies, noting that customers, companies (as represented by first line supervisors), service personnel and *Consumer Reports* view quality quite differently and elaborated on the reasons for these different perceptions.

While top-level management communicates in dollars, operations level personnel must be bilingual,

communicating in both dollars and things, that is, in product units and performance measures. Taguchi popularized the use of loss functions to provide a link between these two languages. They provide a means for expressing the deviation of product characteristics from their targeted values in dollars. These loss functions can be determined through internal costs of a product at each stage of design, development, manufacture and delivery.

Quality is achieved by elaborating the important product characteristics, their targets and specifications. The ability of a product to meet these specifications depends upon its design, development and the processes employed in its manufacture. Product and process information is often gathered through capability studies, where measurements are obtained on important product characteristics, and control charts are employed to address the stability of the processes and the predictability of future performance. These product characteristics are frequently summarized by a statistical distribution, or even more succinctly, by the process capability, six sigma.

Process capability indices are typically employed to combine this voice of the customer with the voice of the product/process into a dimensionless measure. Pignatiello and Ramberg (1996) reviewed this approach, stressing the importance of an appropriate data collection scheme and the statistical analysis and summarization of results. These indices, which are dimensionless quantities, are then employed in quality improvement project selection.

Total Quality

The term quality has traditionally been associated with manufacturing, and more explicitly, with the products, processes, functions, and facilities associated with manufacturing. The modern total quality viewpoint extends this factory oriented view of quality to encompass all products, goods and services whether they are for sale or not.

Total quality proponents embrace training and education as universal, in direct contrast to the Taylor system, a system to which U.S. leadership in productivity has been attributed. Taylor made a strategic decision to separate planning and execution. This decision was based on his assessment that the then immigrant work force was uneducated and that it was not economically feasible to educate them in a timely manner. A more highly educated work force represents an untapped resource for improving quality and productivity. Total quality proponents recognized this improvement in the educational level, and the responsibility for not only utilizing this resource, but improving it through the continuing education of the work force. Furthermore, they recognized these workers as stakeholders, and that by empowering these stakeholders, productivity and quality can be further enhanced.

Total Quality Management

While neither embraced the term Total Quality Management, its origins can be traced to the work of W. Edwards Deming and Joseph J. Juran, and through the implementation of their quality philosophy, concepts and methods in Japanese industry. Kolesar (1994, 2008) discusses the contributions of Deming and Juran to the Japanese quality revolution following WWII. The importance of TQM became fully recognized in the U.S. only after its successful Japanese implementation. The domination of their products, as a direct result of their outstanding quality, especially in the auto industry, could not go unrecognized. With this recognition, Deming and Juran gained the attention of enlightened U.S. corporate and government leaders.

Deming is perhaps best known for the Shewhart/Deming PDCA cycle, and his 14 point manifesto, which is fundamental to TQM philosophy. The PDCA cycle, now called the PDSA cycle, meaning Plan, Do, Study, and Act, provides a fundamental structure for achieving quality. Gitlow et al. (1995) give an excellent discussion of Deming's 14 points and employed the PDSA approach for achieving quality improvement. Scherkenbach (1986, 1991) provides a balanced view of the key characteristics of the philosophy of Deming given in Table 2. For example, one of Deming 14 points is "reduce waste," which Scherkenbach has balanced with "add value."

Kolesar (2008) states, "Juran's 1954 lectures have been credited with being seminal contributions to the Japanese quality control movement." Juran (1988) recognized the importance of including quality in the management game plan, as well as the

Total Quality Management, Table 2 Key characteristics of the Deming philosophy, from W.W. Scherkenbach (1991)

Reduce waste	Add value
Constancy of purpose	Continual improvement
Improvement	Innovation
Team	Individual
Long-term	Short-term
Inputs	Outputs
Synthesis	Analysis
Knowledge	Action

Total Quality Management, Table 3 Quality tools — the magnificent seven plus one

Control Charts
Check Sheets
Histograms
Pareto Diagrams
Ishikawa Fishbone Diagrams
Scatter Plots
Flow Charts or Process Diagrams
Multi-Variate Charts

need for developing managerial processes in managing quality. He noted financial management included three processes: financial planning (producing the budget), financial control (assuring that the budget will be met), and financial improvement (ways of increasing income and decreasing costs). Translated to quality, these are known as the Juran Trilogy: Quality Planning, Quality Control and Quality Improvement. A major advantage facilitating the implementation of these ideas is that senior management already understands them in the financial arena. Juran also stated "universal sequences for accomplishing these processes, the quality planning road map, quality control and the quality improvement processes." Fundamental to his methodology is the recognition of the presence of chronic quality wastes resulting from disconnected alarm systems.

Senge (1993) presents a TQM paradigm that is based on the three cornerstones: Guiding Ideas, Infrastructure, and Theory, Tools and Methods. He noted that guiding ideas are based on a vision. Without this vision, everything is mechanical and pedestrian. Leaders expressing this vision and these guiding ideas must practice them. When they make a decision differently, their colleagues and subordinates will know! However, these ideas and the behavior and actions of the leaders is not enough. An infrastructure is necessary for diffusing these ideas. Conflicts in goals must be resolved and this implies the importance of accountability and an appropriate reward structure. Finally, there is the theory, tools and methods cornerstone. Again, a necessary and important part of the structure, but certainly not sufficient on its own. OR/MS tends to be tool oriented.

Tables 3 and 4 list these essential tools, which seem so simple that they are frequently neglected in college courses. These tools of TQM are communications enhancers that assist one in listening and talking to processes, products, systems and people. Smith (1998) described these tools and more advanced problem solving methods within the context of diagnostic disciplines.

Total Quality Management, Table 4 Quality management — the seven tools

Affinity Diagram
Interrelationship Digraph
Tree Diagram
Prioritization Matrices
Matrix Diagram
Process Decision Program Chart
Activity Network Diagram

Transformation to Quality Organizations

Implementation of total quality management in a firm requires a transformation of the organization, and any transformation of an organization is doomed to failure if it does not recognize the importance of the human aspect. Scherkenbach (1991) elaborated a theory of transformation that emphasizes this human aspect of quality. Scherkenbach notes how differently people view the world and why they are motivated by different means. Some, such as management scientists and operations researchers, live in the logical world. They tend to proceed on the basis of logical actions. Others, including many top-level managers and workers alike, live in a physical world. This is the world of policies, procedures, standards, rewards, and punishments. They do it by the book. Still others, such as sales personnel, marketing specialists and artists live in the emotional world, typified by the statement, "The force is with you."

Scherekenbach's point is not to create stereotypes, but to enable a better understanding of why arguments made in one of these domains often do not have a substantive impact on people living in another domain, i.e., when dealing with others, it is imperative to recognize that they may not be motivated by different forces. To make progress in relationships with others, one needs to be cognizant of their view and address them in an appropriate manner. As a point of exclamation to those of us who live in the logical world, Scherkenbach quotes Schopenhauer: "No one ever convinced anybody by logic; and even logicians use logic only as a source of income."

He goes on to describe transformation through three process relationships: one for each world view, and all given in terms of different mind states or attitudes dependent, independent and interdependent. Many people function solely in either the dependent or independent mode. An important aspect of the quality transformation is to facilitate the move to the interdependent mode.

TQM and Principle-Based Management

Each of us holds an important key to any quality transformation process in which we are involved. Covey (2004) suggested that we begin the quality transformation by taking action on ourselves first; then proceed through the four steps of his inside-out principle based management. He described these four steps as self, interpersonal, managerial and organizational. At the self level, he stresses the need to carefully develop our vision, decide what our life is about and develop those principles that will serve as our guidelines in making all of our decisions in life. Next is the need to act on this vision in a consistent manner that builds an internal source of security. Immediate or complete success should not be expected since this is a learning process. Incorporating and practicing the Shewhart/Deming PDSA cycle in our own work is an important method for improving the quality or our own work.

As we achieve some comfort with ourselves, and create a more positive opinion of ourselves, we will be able to move on to the interpersonal level. Covey stated that quality at the interpersonal level means that we live by the correct principles in our relationship with other people. Here Covey used the analogy of a bank account, that is, we make deposits to and withdrawals from an emotional bank account. He stated three important ground rules for achieving quality in interpersonal relationships. First, when we have a problem with a person, we should go directly to them and explain it. The second relates to the conduct of meetings. His ground rule is that no one is allowed to make a point in a meeting until they restate the point of their predecessor, and state it in a manner that is satisfactory to that person. He notes that this eliminates the majority of disagreements, since most of them are simply misunderstandings. Through this mechanism potential misunderstandings can be quickly clarified, avoiding arguments, further miscommunications and withdrawals from the emotional bank account. Furthermore, having greatly reduced the number of misunderstandings, there is a better chance to disagree agreeably when new disagreements take place. An important question is do we have the courage to practice this ground rule and continue to practice it even if the rest of group does not.

Finally, when we do make mistakes, we need to have the courage to say that we were wrong. No excuses. We must apologize to the person; we must also apologize to the other people involved. At the managerial level, quality means that we attempt to empower people. In this way they become increasingly independent of us. They supervise themselves, and we become a source of help, rather than a micromanager. Empowerment begins with self-control and self-inspection and extends to self-directing work teams. These teams plan processes, establish schedules, assign personnel and maintain discipline through peer pressure. They accomplish the work that was once limited to managers and specialists. Juran (1988) suggests that this system could be the successor to the Taylor system. It offers the opportunity to step off of the productivity and quality plateaus, which have been directly traced to the lack of involvement of the total work force, a result of not questioning the assumptions underlying Taylor's original separation of planning and execution. A craftsperson created a product from start to finish, and thus recognized the impact of each step on the following one. The production worker, as the execution of production was broken into individual components, had a smaller and decreasing opportunity to comprehend his role in achieving quality. As a result, inspection departments and later quality

departments emerged, acting as policing units in the goal to achieve quality.

At the organizational level, the key is in the structures and the leadership styles. Are the leaders in harmony with the mission statement? Was everyone involved in the development of the mission statement?

TQM and the Malcolm Baldrige Award

The Malcolm Baldrige Award framework provides an excellent road map for implementing TQM, as well a method for evaluating a firm's progress (NIST 1999). The framework emphasizes dynamic relationships between eleven categories of core values and concepts. These underlying core values and concepts are: customer-driven quality, leadership, continuous improvement and learning, employee participation and development, fast response, design quality and prevention, long-range view of the future, management by fact, partnership development, corporate responsibility and citizenship and results orientation.

The stated goals are customer satisfaction, customer satisfaction relative to competitors, customer retention and market share gain as measured by product and service quality, productivity improvement, waste reduction/elimination, supplier performance and financial results. Leadership is viewed as the "driver" category of core values and concepts, driving the two categories: business results and customer focus and satisfaction through a system of processes. The system of processes consists of four "well-defined and well-designed processes" for achieving the firm's performance requirements and the firm's customer requirements. These four system categories are information and analysis, strategic planning, human resource development and management, and process management. The criteria, which are updated annually, are disseminated by the American Society for Quality Control and the National Institute of Standards and Technology.

TQM and Six Sigma

Six Sigma is a relatively new program for accomplishing institutionalizing quality. The fundamental concept was created by a Motorola reliability engineer. Lean six sigma, a more recent development, incorporates fundamental industrial engineering and business "lean practices," with six sigma quality principles. Ramberg (2000) describes six sigma programs, and details its history in "Six Sigma: Fad or Fundamental."

The Status of TQM

One of the first evaluations of TQM was conducted by Senge. In his 1993 ASQ Annual Conference keynote address, titled "The Health and Well Being of the TQM Movement," he posed the following questions: "Are fundamental breakthroughs being made? Are they being made in your organization?" Following this opening, he summarized surveys by Arthur D. Little and McKinsey, and made the following conclusions. Out of 500 firms surveyed, less than a third were accomplishing anything! Two thirds of the TQM programs had ground to a halt! He went on to diagnose TQM failures and successes. Based on his case studies, he concluded that there were only a few major reasons for failure. The three major ones were: conflict between time and effort; wavering goals, and employee perception that their job was at risk.

Even where TQM has "succeeded," there are questions about the measures used to judge that success. That is, in many cases, even where the TQM indicators improved, the health of the company (e.g., as judged by its price) did not get any better, even over a reasonably long term. That is, TQM did not improve the health of the organization as judged by its stockholders. Reporting on the root cause of these problems, Senge concluded that a major reason was that most organizations viewed TQM as programmatic. Presented or implemented in this manner, TQM is certain to be DOA.

Comparative studies measuring the impact of TQM on a firm's business performance also began to appear. Jarrell and Easton (1994) reported some evidence that long-term performance of firms adopting TQM is improved. This result is consistent across the accounting and stock price performance measures examined. Similar, but overall stronger results, were found when the analysis was limited to a subsample of pilot firms identified as having more mature and well-integrated TQM systems. Hendricks and Singhal (1999) concluded that effective implementation of TQM "pays off in a big way." They made this

conclusion by comparing the business performances of firms judged to have successfully implemented TQM with a control group of firms.

van der Wiele et al. (2000) examined TQM through a "fad, fashion, and fit" analysis. Utilizing a range of research studies, which began in the late 1980s, they identified three stages in the evolution by which a fad can achiever a fit with previous management practice. In stage 1, the fad must be clearly defined and measurable. For TQM this clarification was ISO 9000 and the Baldridge Award. Stage 2 is the move to a fashion, which happens when major pressures toward widespread adoption of the fad are present. Again, ISO 9000 serves as an example, because suppliers experienced a pressure from major customers to achieve certification. van der Wiele et al. (2000) state, "As a consequence, the ISO 9000 series became a fast-spreading fashion." They elaborate that "Stage 3 is the move either from fad to fit or from fashion to fit. Fit into normal management practice means that the original fad will have effected the normal way of working within whole organizations and not just a small part such as would be the case in the adoption of a mere fashion." Their fieldwork shows that such a change will only occur when there is strong internal motivation and emotional involvement to implement TQM. They also point out that, "Should such a move take place from fad or fashion to fit, then the chances are that organizational performance will also be perceived to have been effected in a positive way."

Prajogo and Brown (2004) examining the relationship between TQM practices and quality performance in Australian organizations. They compared organizations that adopted formal TQM programs with those without a formal program. They concluded that the lack of a formal program did not necessarily mean TQM principles were not being practiced. Their findings also showed that the firms adopting formal TQM programs implemented several TQM practices at a higher level than those that did not have TQM programs. However, they did not observe a significant difference between organizations implementing formal TQM programs, and those organizations simply adopting TQM practices, suggesting that it is the adoption of quality practices that matters rather than formal programs per se.

While some researchers have given a rather pessimistic view on the future of the quality management movement, Kujala and Lillrank (2004) note that quality management has survived the failure of some of its success stories, such as those of Motorola and Xerox. They affirm that TQM remains to be properly defined, and that its scientific foundations are still not transparent.

Cheng (2007) explored a model for integrating TQM and Six Sigma with business strategy. He concluded that, "Implementing Six Sigma has become a common theme in organizations of all sizes, within a TQM infrastructure."

To summarize, it seems that TQM and its derivatives are fitting into management infrastructure. However, it is important for quality proponents that TQM is not the only thing. TQM will continue to require definition and structural development based on scientific foundations. Most recent of these has come from the Six Sigma movement, and more recently from Lean Six Sigma. Transformational leadership remains a requirement for continued success.

See

- ▶ Quality Control
- ▶ Reliability of Stochastic Systems

References

Cheng, J. L. (2007). Six sigma and TQM in Taiwan: An empirical study. *Quality Management Journal, 14*(2), 7–18.

Cole, R. E. (1992). The quality revolution. *Production and Operations Management, 1*, 118–120.

Covey, S. (2004). *The 7 habits of highly effective people*. New York: Fireside Books.

Crosby, P. B. (1989). What are requirements? *Quality Progress*, 47. ASQC.

Defeo, J. A., & Juran, J. M. (2010). *Juran's Quality handbook* (6th ed.). Southbury, CT: Juran Institute.

Deming, W. E. (2000). *Out of the crisis*. Cambridge, MA: MIT Press.

Garvin, D. A. (1988). *Managing quality*. New York: Free Press.

Gitlow, H., Oppenheim, A., & Oppenheim, R. (1995). *Quality management: Tools and methods for improvement*. Homewood, IL: Irwin.

Hendricks, K. B., & Singhal, V. R. (1999). Don't count TQM out. *Quality Progress*, 35–42.

Jarrell, S. L., & Easton, G. S. (1994). An exploratory empirical investigation of the effects of total quality management on corporate performance. In P. Lederer (Ed.), *The practice of quality management*. Cambridge, MA: Harvard Business School Press.
Juran, J. M. (1988). *Juran's quality control handbook* (4th ed.). New York: McGraw-Hill.
Kolesar, P. J. (1994). What Deming told the Japanese in 1950. *Quality Management Journal, 2*(1), 9–24.
Kolesar, P. J. (2008). Juran's lectures to Japanese executives in 1954: A perspective and some contemporary issues. *Quality Management Journal, 15*(3), 7–16.
Kujala, J., & Lillrank, P. (2004). Total quality management as a cultural phenomenon. *Quality Management Journal, 11*(4), 43–55.
Merriam Webster Staff. (2004). *The Merriam-Webster dictionary*. Springfield, MA: Merriam-Webster.
NIST (National Institute of Standards and Technology). (1999). *Ten years of business excellence for America*. Gaithersburg, MD: National government publication.
Pignatiello, J. J., Jr., & Ramberg, J. S. (1996). Process capability: Engineering and statistical issues. In J. B. Keats & D. C. Montgomery (Eds.), *Statistical applications in process control*. New York: Marcel Dekker.
Prajogo, D., & Brown, A. (2004). The relationship between TQM practices and quality performance and the role of formal TQM programs: An Australian empirical study. *Quality Management Journal, 15*(3), 32–42.
Prybutok, V. R., & Zhang, X. (2010). Introduction to the special issue on quality management in healthcare. *Quality Management Journal, 17*(4), 7.
Ramberg, J. S. (1994). TQM: Thought revolution or Trojan horse? *OR/MS Today, 21*(4), 18–24.
Ramberg, J. S. (2000). Six sigma: Fad or fundamental. *Quality Digest*, May 2000.
Scherkenbach, W. W. (1986). *The Deming route to quality and productivity: Road maps and road-blocks*. Washington, DC: ASQC Press/Washington CEE Press.
Scherkenbach, W. W. (1991). *Deming's road to continual improvement*. Knoxville, TN: SPC Press.
Scholtes, P. R., & Hacquebord, H. (1988). Six strategies for beginning the quality transformation (Part III). *Quality Progress*, 28–33.
Senge, P. (1990). *The fifth discipline: The Art and practice of the learning organization*. New York: Doubleday.
Senge, P. (1993). Quality management: Current state of the practice. *Keynote speech at the American Quality Congress*.
Smith, G. F. (1998). Determining the cause of quality problems: Lessons from diagnostic disciplines. *Quality Management Journal, 5*(2), 24–41.
Taguchi, G. (1986). *Introduction to quality engineering*. Tokyo: Asian Productivity Organization.
van der Wiele, A., Williams, A. R. T., & Dale, B. G. (2000). Total quality management: Is it a fad, fashion, or fit? *Quality Management Journal, 65*(2), 65–79.
Vaughn, R. C. (1967). *Introduction to industrial engineering*. Ames, IA: Iowa State University Press.
Wooden, J., & Yaeger, D. (2009). *A game plan for life: The power of mentoring*. New York: Bloomsbury Press.

TQC

Total quality control.

See

- ▶ Quality Control
- ▶ Total Quality Management

TQM

Total quality management.

See

- ▶ Quality Control
- ▶ Total Quality Management

Traffic Analysis

Denos C. Gazis
PASHA Industries, Katonah, NY, USA

Introduction

Traffic analysis has flourished since the 1950s, stimulated from the need to address the ever-growing traffic problems of cities around the world. In true scientific tradition, it has yielded an understanding of the fundamental characteristics of automobile traffic, which in turn spawned significant contributions in the management and optimization of traffic facilities. This article outlines some of the most important developments in one area of traffic analysis, that of traffic flow, including certain associated queueing phenomena. Aspects of control of traffic networks that are outside the scope of this article can be found in Gazis (1992).

A Kinematical Theory of Traffic Flow

One of the earliest, and most durable, contributions to the understanding of traffic flow was given by Lighthill and Whitham (1955). They viewed the traffic as a special fluid which obeys some basic laws consistent with the physical nature of traffic, such as its unidirectional influence of a vehicle only on the traffic behind it, the constraints on flow imposed by human limitations, etc. The Lighthill-Whitham theory is based on two basic postulates:

1. Traffic is conserved, in the sense that traffic units by and large are neither created nor annihilated; and
2. There is a fundamental relationship between traffic flow and traffic density, resulting from the physical characteristics of the traffic system.

The first postulate is expressed in the relationship

$$\frac{\partial k}{\partial t} + \frac{\partial q}{\partial x} = 0 \tag{1}$$

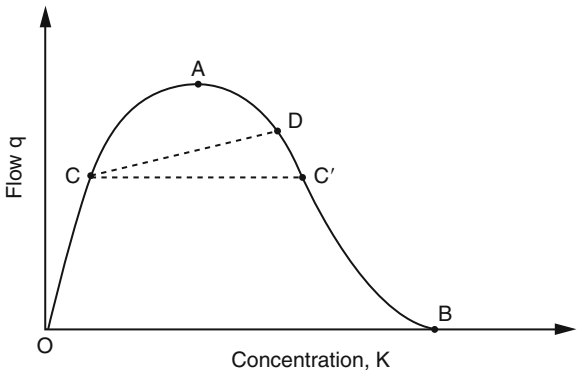
where q is the traffic flow in vehicles per unit of time t, k is the density of traffic in vehicles per unit of distance x, and v is the (average) speed of the traffic fluid. The second postulate is expressed by the relationship

$$q = f(k)$$

between flow q and density k such as that shown in Fig. 1. At zero density, there is zero flow. The flow is also zero at some jam density, k_j, because traffic grinds to a halt as vehicles are packed bumper to bumper. Between these two extremes, traffic flow builds up to a maximum and then decreases down to zero.

A number of interesting properties of traffic can be described on the basis of these two postulates. They relate to observable phenomena such as wave propagation, i.e., the movement along the traffic stream of a transition point corresponding to a change in traffic characteristics, the queueing caused by an obstruction of the traffic movement, etc.

Wave Propagation — Traffic moving at a steady-state flow rate q_1 and density k_1 may shift to a different flow rate q_2, and a corresponding density k_2, by a change in roadway quality, obstruction, or other external influence. When this happens, vehicles situated in a transition region undergo maneuvers adjusting their speed and inter-vehicle spacing, and



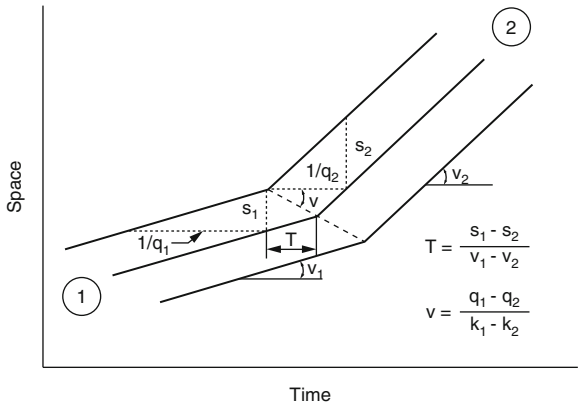
Traffic Analysis, Fig. 1 Flow vs. concentration relationship

this transition region generally moves either forward or backward in space depending on the nature of the change. The adjustments of speed and spacing are gradual, but for the purpose of deriving the characteristics of wave propagation may be assumed abrupt as suggested by Lighthill and Whitham (1955). This assumption leads to the conclusion that a change from one steady-state flow condition to another is associated with a "shock wave," an expression that pervades the traffic engineering literature.

The shock wave marks the transition from one speed to another, and moves always backwards with respect to the traffic stream, since vehicles exert an influence only on vehicles behind them. (The influence of an occasional tailgating vehicle pushing the vehicle in front is ignored as an unimportant aberration). The speed of movement of the shock wave along a roadway, may be obtained on the basis of Eq. (1), and is given by

$$v = \frac{q_1 - q_2}{k_1 - k_2} \tag{3}$$

It should be pointed out that the result given in Eq. (3) depends only on the postulate of conservation of traffic, and is totally independent of any specific relationship between flow and concentration, or even on the existence of such a relationship. It results from kinematical considerations shown in Fig. 2. The transition from one steady state flow situation to another results in a propagation of the change of the corresponding speed along the roadway. The phase velocity of this propagation depends only on the values of the initial and final pairs of flow, q, and



Traffic Analysis, Fig. 2 Transition from one steady-state-flow situation to another

concentration, k, and is given by Eq. (3). If, in addition, a relationship between flow and concentration is assumed (Fig. 1), different domains of traffic quality, and corresponding characteristics of wave propagation, can be defined as follows:

1. The range from zero flow at zero density to maximum flow (Section OA, Fig. 1) corresponds to relatively uncongested traffic flow. A small increase in density in this domain moves forward along the roadway;
2. The range from maximum flow to zero flow at "jam density" (Section AB, Fig. 1) corresponds to relatively congested, stop-and-go traffic. A small increase of density in this domain moves backwards along the roadway; and
3. Any transition from one steady state flow to another (as from point C to point D, Fig. 1) is associated with a wave propagation given by the slope of segment CD.

Queueing — Queueing may be caused by a reduction in roadway capacity at a fixed point on the roadway, or by an obstruction causing traffic to shift from the uncongested to the congested branches of the (q, k) curve, even without reduction in flow rate, (line CC′ in Fig. 1). The rate of growth of the queue can be estimated using the same methodology described above. For example, a total obstruction of flow q and density k causes a queue formation, with the tail-end of the queue moving backwards along the roadway with speed equal to

$$v = \frac{q}{k_j - k} \tag{4}$$

Additional results from the kinematic treatment of traffic — An extensive literature exists on applications of the Lighthill-Whitham model to various traffic phenomena. A word of caution is appropriate with regard to such applications. The Lighthill-Whitham model describes well only transitions from one steady state to another. Any attempt to apply the model to a sequence of traffic maneuvers that do not allow enough relaxation time between changes of speeds violates the basic spirit of the model.

An interesting extension of the above kinematical treatment of traffic was applied by Gazis and Herman (1992) for the treatment of a moving obstruction such as that caused by a vehicle moving more slowly than the other vehicles in the traffic stream. The character of this " moving bottleneck" is different from that of a fixed bottleneck, and the Gazis-Herman treatment derives the characteristic queueing behavior associated with it. Gazis and Herman obtain a description of the queueing caused by a slow vehicle on a two-lane highway. Both lanes are affected by such a vehicle, one by direct trapping of vehicles behind the slow one, and the other by interference from vehicles escaping from the queue behind this vehicle. The result is that queueing takes place in both lanes in the vicinity of the slow vehicle, with the affected vehicles moving at an average speed only marginally higher than that of the slow one, until they come abreast of this slow vehicle and are able to escape at their normal speed. Gazis and Herman also propose an explanation of the phenomenon of a phantom bottleneck, the seemingly unexplainable regions of congestion that drivers often traverse. Some of them may be caused by a moving bottleneck caused by a vehicle that slows down temporarily and then resumes its normal speed; for example, a heavily loaded truck temporarily slowing down along an uphill portion of the roadway. The Gazis-Herman treatment provides a rational way of estimating the minimum allowable speed on a highway, which would not affect its throughput.

A Boltzmann-like Model of Traffic Flow

In 1959, Prigogine suggested a model of traffic flow founded on statistical mechanics, analogous to the Boltzmann model of gases (Prigogine 1961). The Prigogine model was subsequently developed extensively by Herman, Prigogine and their

collaborators (Prigogine, Herman and Anderson 1962, 1965; Prigogine and Herman 1971). They considered a stream of traffic as an ensemble of units associated with certain statistical properties. In particular, a vehicle was associated with a desired speed which it would follow as long as it was not constrained by another vehicle in front with a lower desired speed.

Thus, traffic is described in terms of a probability density for the speed, v, of an individual car, $f(x, v, t)$. This density may vary as a function of time, t, and a coordinate x along the highway. The basic equation for this function f is assumed to be

$$\frac{\partial f}{\partial t} + \upsilon \frac{\partial f}{\partial x} = \left(\frac{\partial f}{\partial t}\right)_{relaxation} + \left(\frac{\partial f}{\partial t}\right)_{interaction} \tag{5}$$

The first term of the righthand side of Eq. (5) is a consequence of the fact that $f(x, v, t)$ differs from some desired speed distribution $f^0(v)$. A car tries to "relax" to its desired speed as soon as it finds an opportunity to do so. The second term of the righthand side corresponds to the slowing down of a fast vehicle by a slow one. True to his tradition as a leading expert in statistical mechanics, Prigogine frequently referred to this second term as the collision term — a rather unsettling choice of words in this context!

The form for these two terms was chosen for mathematical convenience and plausibility, leading to the equation

$$\frac{\partial f}{\partial t} + \upsilon \frac{\partial f}{\partial x} = \frac{f - f_0}{\tau} + (1 - p)k(V - \upsilon)f \tag{6}$$

where τ is a characteristic relaxation time, p is the probability of a car's passing another car, and V is the average speed of the stream of traffic. The second term of the right-hand side of Eq. (6) corresponds to the interaction term, and tends to zero at very light traffic concentration when the probability of passing is close to unity, in which case the relaxation term is dominant. If, in addition, a highway with constant properties along its length is assumed, then $\partial f/\partial x = 0$ and the solution of Eq. (6) is

$$f(\upsilon, t) = f^0(\upsilon) + \left[f(\upsilon, 0) - f^0(\upsilon)\right] e^{-t/\tau} \tag{7}$$

If interested only in solutions of Eq. (6) that are independent of time and space, then the lefthand side of this equation is zero. The equation may then be solved to yield an equation of state whose general form, for small values of the concentration, corresponds to an approximately linear increase of flow with concentration, e.g.,

$$q = V^0 k \tag{8}$$

where V^0 is the average of the desired speed. As k increases, the flow q falls below the straight line (8) due to the increasing influence of interactions.

In the range of high concentrations, q is independent of f^0 and depends only on τ and p, according to the equation

$$q = \frac{1}{\tau(1 - p)}. \tag{9}$$

The complete solution of Eq. (6) for steady-state flow, independent of time and space, is shown in Fig. 3. For any given f^0, the flow q rises with k, reaches a maximum, and then decreases until it intersects a curve corresponding to Eq. (9). This curve may be viewed as a universal curve of collective flow, characterized by high densities and very little passing. One very realistic feature of this theory is the fact that it predicts probable stoppage of some vehicles in the domain of collective flow, in agreement with the common experience of stop-and-go traffic at high concentrations.

It is appropriate to make an observation concerning the linkage of the Herman-Prigogine and Lighthill-Whitham theories in the range of very high densities. Since traffic at those densities is of a stop-and-go nature, it is not really steady-state traffic in the sense of being associated with constant speed and density. Rather, it is associated with alternating states of following slow platoons and escaping from them. Given this fact, it becomes clear that one should not try to apply the Lighthill-Whitham method in describing shock waves and wave propagation involving transitions into this domain of traffic movement, since the L-W theory describes well only clean transitions between two steady-state situations.

Herman and Prigogine (1979), together with several collaborators, went on to use the results of their model to develop a two-fluid approach to town traffic. This approach postulates that traffic in towns is a mixture of

two fluids, one that moves and one that is stopped. Any individual vehicle traverses a network in a stop-and-go fashion, moving part of the time and being stopped part of the time. The quality of service in a particular urban network can be described in terms of two parameters that can be determined by circulating a test vehicle through the network and measuring the percentages of time during which the vehicle is moving, or is being stopped. Thus the two-fluid model yields a simple description of the system-wide traffic quality in congested urban networks. It allows comparison between different urban networks, and it offers the potential of identifying important elements of the network, related to its geometry or control features, which may be targeted for improvement of the service quality.

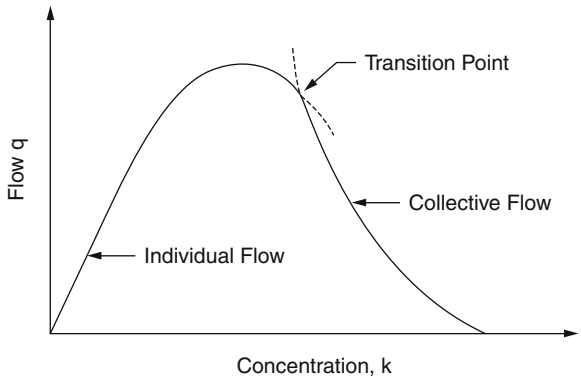
A Car-following Theory of Traffic Flow

Reuschel (1950) and Pipes (1953) proposed models to describe the detailed motion of cars proceeding close together in a single lane. This microscopic, car-following theory of traffic flow was extensively developed by Herman et al. (1959). The theory is based on the fact that when drivers do not have the freedom to pass a vehicle in front, they follow it in a way that is controlled by the overriding need to avoid coinciding with the leader in space and time. In trying to achieve this reasonable objective, drivers react to a limited set of inputs. The postulate of the car-following theory, confirmed by experiments, was that drivers reacted mostly to the relative speed between their car and that of the one in front. Experiments showed a high correlation between the acceleration of a car and its speed relative to that of a leader, after a time-lag of the order of 1 second. This led to the linear car-following model

$$\frac{d^2x_n(t+T)}{dt^2} = \lambda\left[\frac{dx_{n-1}(t)}{dt} - \frac{dx_n(t)}{dt}\right] \qquad (10)$$

in which n denotes the position of a car in a line of cars (a platoon), λ is a constant gain factor, T is the reaction time-lag, and x_n is the position of the nth car on the highway.

This model was used to investigate the stability of a traffic platoon when a perturbation in its movement is introduced. The movement of the platoon is said to be locally stable if the amplitude of a perturbation, for any given car in the platoon, decreases in time. It is asymptotically stable if the amplitude of the perturbation decreases as it propagates upstream. The value of the product λT is the determinant of stability or instability, local or asymptotic. When $\lambda T < 1/e$, a perturbation is damped exponentially as it is passed on to the following car, signifying a very stable situation. For λT between the values of $1/e$ and $\pi/2$, the perturbation produces oscillations of decreasing amplitude between pairs of cars, signifying still a locally stable situation. For $\lambda T > \pi/2$, a perturbation produces oscillations of increasing amplitude, signifying a locally unstable situation.



Traffic Analysis, Fig. 3 Flow vs. concentration relationship according to the Boltzmann-like model of traffic flow

With regard to asymptotic stability, the dividing line is at $\lambda T = 1/2$. For values of λT below 1/2, the amplitude of a perturbation decreases as it propagates backwards; for values of λT greater than 1/2, it increases. This means that between $1/e(\sim 0.368)$ and 1/2 is a situation that is locally stable but asymptotically unstable. Any pair of cars in a platoon is able to absorb a perturbation, but it amplifies it as it passes it backwards, until the perturbation is so large that it causes a collision.

The linear car-following model may be satisfactory in describing fluctuations around a steady-state, constant speed situation. It cannot be expected to describe equally well transitions from one steady state to another involving large changes of speed. For this reason, Gazis et al. (1961) proposed a nonlinear model in which the gain factor is not constant but depends on the speed of the follower and the relative spacing between leader and follower according to the relationship

$$\lambda = \frac{[v_n(t+T)]^l}{[x_{n+1}(t) - x_n(t)]^m} \tag{11}$$

where c is a constant, $v = dx/dt$ is the speed, and (l, m) are integer exponents identifying particular nonlinear models.

Various values of pairs (l, m) were used to define car-following models and investigate their predictions concerning transitions between one steady-state flow situation and another. Integrating over time Eq. (10), with λ described by Eq. (11), leads to the functional relationship between changes of speed and concentration. Together with appropriate boundary conditions, for example the condition of zero speed at jam density, bumper-to-bumper concentration, one can then obtain a phenomenological relationship between flow and concentration such as that shown in Fig. 1. Various pairs (l, m) have been used which yielded quite plausible relationships, consistent with observations.

The preceding discussion outlines most of the key contributions in the car-following treatment of traffic flow. Additional studies have been contributed by Gazis (1965) within the framework of control theory to account for physical constraints on the system, such as limited acceleration or deceleration capability of cars.

Concluding Remarks

As is the case for every scientific endeavor, much can be done to improve the theories of traffic analysis. For example, car-following theories ignore interaction of cars with other than the car just in front, whereas there is evidence that drivers are very much conscious of happenings several cars in front of them, and this consciousness tends to improve the stability of traffic. Another observation that must be made about virtually all traffic models described here is that they effectively correspond to flat, straight, and infinitely long highways. It is clear that the geometry of highways, including curves and inclination, has a strong effect on the behavior of traffic. A systematic study of such effects would greatly advance understanding of traffic movement, and produce necessary tools for future improvements in traffic management.

The analytical description of traffic flow has already had a profound influence on traffic engineering practice, and the advent of activities in the area of Intelligent Transportation Systems (ITS) points to an increasing reliance on analytical investigations of traffic systems toward improvement of their operation. One needs an improved understanding, and an improved analytical description of traffic phenomena, such as the onset of congestion, queueing, and inter-vehicle signal propagation, in order to create the theoretical under-pinning toward the use of high technology for the improvement of traffic systems, which is the central thrust of ITS. Some improvement will come from direct application of analytical results. For example, the development of automatic highways will undoubtedly draw from knowledge based on car-following models. Other improvements may come from the improved understanding of traffic phenomena that traffic analysis provides, leading to improved heuristic schemes for the control and optimization of traffic systems.

See

- ▶ Network Optimization
- ▶ Networks of Queues
- ▶ Queueing Theory
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Anderson, R. L., Herman, R., & Prigogine, I. (1962). On the statistical distribution function theory of traffic flow. *Operations Research, 10*, 180–196.

Ardekani, S. A., & Herman, R. (1985). A comparison of the quality of traffic service in downtown networks of various cities around the world. *Traffic Engineering and Control, 26*, 574–581.

Ardekani, S. A., & Herman, R. (1987). Urban network-wide traffic variables and their relations. *Transportation Science, 21*, 1–16.

Bick, J. H., & Newell, G. F. (1960). A continuum model for two-directional traffic flow. *Quarterly of Applied Mathematics, 18*, 191–204.

Chandler, R. E., Herman, R., & Montroll, E. W. (1958). Traffic dynamics: Studies in car-following. *Operations Research, 6*, 165–184.

Chang, M.-F., & Herman, R. (1981). Trip time versus stop time and fuel consumption characteristics in cities. *Transportation Science, 15*, 183–209.

Edie, L. C., & Foote, R. S. (1960). Effect of shock waves on tunnel traffic flow. *Proceedings of Highway Research Board, 39*, 492–505.

Edie, L. C., Herman, R., & Lam, T. N. (1980). Observed multilane speed distribution and the kinetic theory of vehicular traffic. *Transportation Science, 14*, 55–76.
Foster, J. (1962). An investigation of the hydrodynamic model for traffic flow with particular reference to the effect of various speed-density relationships. *Proceedings of Australian Road Research Board, 1*, 229–257.
Gazis, D. C. (1965). Control problems in automobile traffic. *Proceedings of IBM scientific symposium on control theory and applications*, *IBM Yorktown Heights, New York*, 171–185.
Gazis, D. C. (1992). Traffic modelling and control: Store and forward approach. In M. Papageorgiou (Ed.), *Concise encyclopedia on traffic and transportation* (pp. 278–284). New York: Pergamon Press.
Gazis, D. C., & Herman, R. (1992). The moving and phantom bottlenecks. *Transportation Science, 6*, 223–229.
Gazis, D. C., Herman, R., & Potts, R. B. (1959). Car-following theory of steady-state traffic flow. *Operations Research, 7*, 499–505.
Gazis, D. C., Herman, R., & Rothery, R. W. (1961). Nonlinear follow-the-leader models of traffic flow. *Operations Research, 9*, 546–567.
Greenberg, H. (1959). An analysis of traffic flow. *Operations Research, 7*, 79–85.
Herman, R., & Potts, R. B. (1961). Single-lane traffic theory and experiment. In Herman, R. (Ed.), *Proceedings of the 1st international symposium on the theory of traffic flow*. Elsevier, 120–146.
Herman, R., & Ardekani, S. A. (1984). Characterizing traffic conditions in urban areas. *Transportation Science, 18*, 101–140.
Herman, R., Montroll, E. W., Potts, R. B., & Rothery, R. W. (1959). Traffic dynamics: Analysis of stability in car following. *Operations Research, 7*, 86–106.
Herman, R., & Prigogine, I. (1979). A two-fluid approach to town traffic. *Science, 204*, 148–151.
Leutzbach, W. (1967). Testing the applicability of the theory of continuity on traffic flow at bottle-necks. In Edie, L. C., Herman, R., & Rothery, R. W. (Eds.), *Proceedings of the 3rd international symposium on theory of traffic flow*. Elsevier, 1–13.
Lighthill, M. J., & Whitham, G. B. (1955). On kinematic waves: II. A theory of traffic flow on long crowded roads. *Proceedings of Royal Society (London), A229*, 317–345.
Makigami, Y., Newell, G. F., & Rothery, R. W. (1971). Three-dimensional representations of traffic flow. *Transportation Science, 5*, 302–313.
Newell, G. F. (1965). Instability in dense highway traffic, a review. In Almond, J. (Ed.), *Proceedings of the 2nd international symposium on theory of traffic flow*. OECD, 73–83.
Newell, G. F. (1991). *A simplified theory of kinematic waves*. Research report UCB-ITS-RR-91-12, University of California at Berkeley.
Pipes, L. A. (1953). An operational analysis of traffic dynamics. *Journal of Applied Physics, 24*, 274–281.
Prigogine, I. (1961). A Boltzmann-like approach to the statistical theory of traffic flow. In Herman, R. (Ed.), *Proceedings of the 1st international symposium on the theory of traffic flow*. Elsevier, 158–164.
Prigogine, I., & Andrews, F. C. (1960). A Boltzmann-like approach for traffic flow. *Operations Research, 8*, 789–797.
Prigogine, I., & Herman, R. (1971). *Kinetic theory of vehicular traffic*. New York: American Elsevier.
Prigogine, I., Herman, R., & Anderson, R. L. (1965). Further developments in the Boltzmann-like theory of traffic flow. In Almond, J. (Ed.), *Proceedings of the 2nd international symposium on the theory of traffic flow*. OECD, 129–138.
Prigogine, I., Herman, R., & Anderson, R. L. (1962). On individual and collective flow. *Académie Royale de Belgique — Bulletin de la Classe des Sciences, 48*, 792–804.
Prigogine, I., Resibois, P., Herman, R., & Anderson, R. L. (1962). On a generalized Boltzmann-like approach for traffic flow. *Académie Royale de Belgique — Bulletin de la Classe des Sciences, 48*, 805–814.
Reuschel, A. (1950). Fahrzeugbewegungen in der Kolonne bei gleichfoermig beschleunigtem oder verzoegertem Leitfahrzeug. *Zeit. d. oesterreichischen Ing. u. Arch. Vereins, 95*, 73–77.
Underwood, R. T. (1962). Some aspects of the theory of traffic flow. *Proceedings of Australian Road Research Board, 1*, 35.
Underwood, R. T. (1964). Traffic flow models. *Traffic Engineering and Control, 5*, 699–701.

Traffic Equations

In a queueing network, the set of linear equations that results from balancing flow into each node with the flow out. These traffic equations are derived by recognizing that the total input seen at a node comes from summing the flow of new arrivals from outside the network with the flow of arrivals that are due to departures from service completions at nodes within the network:

$$\lambda_i = \gamma_i + \sum_j \lambda_j r_{ij}$$

where λ_i is the total input flow rate seen at node i, γ_i is the external input rate to node i, r_{ij} is the probability that a service completion at node i is routed to node j, and the summation is taken over all nodes in the network.

See

▶ Conservation of Flow
▶ Networks of Queues
▶ Queueing Theory

Traffic Intensity

The average load offered to each server in a queueing system.

See

- ▶ Offered Load
- ▶ Queueing Theory

Traffic Process

A stochastic point or marked point process representing the flow of customers on the arcs of a queueing network. Marks represent some aspect of the customer or the state of the network and the points represent the epoch of the event.

See

- ▶ Arrival Process
- ▶ Departure Process
- ▶ Input Process
- ▶ Networks of Queues
- ▶ Output Process

Transfer Function

- ▶ Time Series Analysis

Transient Analysis

The time-dependent solution of a stochastic system (such as a queueing network), as contrasted with a steady-state solution.

See

- ▶ Queueing Theory

Transition Function

A function describing the transition probabilities of a Markov process $\{X(t),\ t\ \in T\}$ into a subset A of the state space as $p(s, x; t, A) = \Pr\{X(t)\ \in\ A|X(s) = x\}$, for state x and times $s < t$ in the time domain T.

See

- ▶ Markov Chains
- ▶ Markov Processes

Transition Matrix

The matrix of (single-step) stationary transition probabilities of a Markov chain $\{X_n\}$, $\boldsymbol{P} = [p_{ij}]$, where $p_{ij} = \Pr\{X_{n+1} = j|X_n = i\}$ is the conditional probability that the chain moves to state j from state i in one step.

See

- ▶ Markov Chains
- ▶ Markov Processes

Transition Probabilities

The conditional probabilities describing the movement from state to state of a Markov process $\{X(t), t\ \in T\}$. In general, the transition probabilities are written as $\Pr\{X(t)\ \in\ A|X(s) = x\}$ for times $s < t$ in the time domain T and state x and event (set) A in the state space. For a homogeneous discrete-time Markov chain (DTMC) $\{X_n, \text{n} \geq 0\}$, the stationary transition probabilities are $\Pr\{X_{n+1} = j|X_n = i\} = p_{ij}$, for states i and j in the space state.

See

- ▶ Markov Chains
- ▶ Markov Processes

Transportation Problem

A linear-programming problem of the following form is called a transportation problem:

$$\text{Minimize} \sum_i \sum_j c_{ij} x_{ij}$$

subject to

$$\sum_j x_{ij} = a_j \quad i = 1, \ldots, m \quad \text{(origins/supply)}$$

$$\sum_i x_{ij} = b_j \quad j = 1, \ldots, n \quad \text{(destinations/demand)}$$

$$x_{ij} \geq 0.$$

The variables $\{x_{ij}\}$ represent a shipment of a homogeneous product from origin i to destination j, where the $\{a_i\}$ are the amounts of the product to be shipped from the origins i, and the $\{b_j\}$ are the amounts demanded by the destinations j. The form presented here assumes $\sum_i a_i = \sum_j b_j$, but the problem can also be formulated with the origin constraints as $\geq$ inequalities and the destination constraints as $\leq$ inequalities, without the restriction that the total supply equal the total demand. It can be shown that if the $\{a_i\}$ and $\{b_j\}$ are integers, than an optimal basic feasible solution exists that is all integer. The transportation problem is a special network problem whose network representation is called a bipartite graph. The special case with $m = n$ and all $\{a_i\}$ and $\{b_j\}$ equal to 1 is the assignment problem. A transportation problem can be solved by direct application of the simplex method, but due to its mathematical structure, the problem can be solved by an efficient modification of the simplex method called the transportation (primal-dual) simplex method. It can also be solved by specialized network algorithms.

See

- ▶ Assignment Problem
- ▶ Network Optimization
- ▶ Northwest-Corner Solution
- ▶ Transportation Simplex (Primal-Dual) Method
- ▶ Unbalanced Transportation Problem

Transportation Problem Paradox

Some transportation problems exhibit the paradox that an optimal solution can be improved if the total amount of units shipped is more than the total amount shipped by the optimal solution. In other words, one can ship more for less.

Transportation Simplex (Primal-Dual) Method

The dual problem to the primal equation form of the transportation problem can be stated as follows:

$$\text{Maximize} \sum_i a_i u_i + \sum_j b_j v_j$$

subject to

$$u_i + v_j \leq c_{ij} \quad \text{for all } (i, j).$$

Here the $(m + n)$ set of dual variables u_i and v_j are unrestricted (free) variables. Note that the primal has a redundant equation due to the equality of the total supply and demand. Thus, a feasible basis matrix to the transportation problem is of dimension $(m + n - 1) \times (m + n - 1)$. It can be shown that any feasible basis matrix can be arranged into a triangular form. For a given basis, the simplex method requires that the corresponding dual constraints must hold at equality, i.e., $u_i + v_j = c_{ij}$ for all variables x_{ij} in the basis. This $(m + n - 1) \times (m + n)$ set of dual equations can be reduced to an $(m + n - 1) \times (m + n - 1)$ system by arbitrarily setting one of the dual variables, say $u_1 = 0$. This corresponds to removing, as a redundant constraint, the first equation of the transportation problem. The resulting dual square set of equations also has a triangular form that allows for the efficient calculation of the $\{u_i\}$ and $\{v_j\}$ that correspond to the current basic solution. These values of u_i and v_j are used to calculate the $(u_i + v_j)$ terms for the nonbasic

variables, and if each one is less than or equal to its corresponding c_{ij}, then by duality theory and complementary slackness, the current basis is optimal. If the latter condition does not hold, the usual simplex criterion is used to select a variable to enter the basis and a new basic feasible solution is generated by simple adjustments to the flows in the network that describe the current basic feasible solution. This network is a tree that connects all origins and destinations, and the addition of the new variable (or arc to the tree) enables the new solution to be calculated readily. This primal-dual process is repeated until an optimal solution is found. Such a solution exists because the transportation problem always has feasible solutions and the solution set is bounded.

See

- ▶ Network Optimization
- ▶ Transportation Problem

Transposition Theorems

Transposition theorems deal with disjoint alternatives of solvability of linear systems. For example, Stiemke's transposition theorem is the following: For a matrix $\boldsymbol{A} \neq \boldsymbol{0}$, the following statements are equivalent: (1) $\boldsymbol{Ax} = \boldsymbol{0}$, $\boldsymbol{x} > \boldsymbol{0}$, has no solution, and (2) $\boldsymbol{\mu A} \leq \boldsymbol{0}$, $\boldsymbol{\mu A} \neq \boldsymbol{0}$ has a solution.

See

- ▶ Farkas' Lemma
- ▶ Gordan's Theorem
- ▶ Strong Duality Theorem
- ▶ Theorem of Alternatives

Transshipment Problem

- ▶ Minimum-Cost Network-Flow Problem
- ▶ Network Optimization

Traveling Salesman Problem

Karla L. Hoffman[1], Manfred Padberg[2] and Giovanni Rinaldi[3]
[1]George Mason University, Fairfax, VA, USA
[2]New York University, New York, NY, USA
[3]CNR – Istituto di Analisi dei Sistemi ed Informatica (IASI), Rome, Italy

Introduction

The traveling salesman problem (TSP) has commanded much attention from mathematicians and computer scientists specifically because it is so easy to describe and so difficult to solve. The problem can simply be stated as: if a traveling salesman wishes to visit exactly once each of a list of m cities (where the cost of traveling from city i to city j is c_{ij}) and then return to the home city, what is the least costly route the traveling salesman can take? A complete historical development of this and related problems can be found in Hoffman and Wolfe (1985), Applegate et al. (2006), and Cook (2011).

The importance of the TSP is that it is representative of a larger class of problems known as combinatorial optimization problems. The TSP problem belongs in the class of such problems known as *NP*-complete. Specifically, if one can find an efficient (i.e., polynomial-time) algorithm for the traveling salesman problem, then efficient algorithms could be found for all other problems in the *NP*-complete class. To date, however, no one has found a polynomial-time algorithm for the TSP. Does that mean that it is impossible to solve *any* large instances of such problems? To the contrary, nowadays many practical optimization problems of truly large scale are solved to optimality routinely. From 1992 to 2006, Concorde, a software created by D. Applegate, R.E. Bixby, V. Chvátal, and W.J. Cook (Applegate et al. 1995, 2006), solved (among many others) a traveling salesman problem that models the production of printed circuit boards having 7,397 holes (cities), a problem over the 13,509 largest cities in the U.S., one over the 24,978 cities of Sweden, and, finally, a 85,900 city problem arising from a VLSI

application. So, although the question of what it is that makes a problem difficult may remain open, the computational record of specific instances of TSP problems coming from practical applications is optimistic.

How are such problems tackled? Obviously, one cannot consider a brute-force approach. For example, for a 16-city traveling salesman problem, there are 653,837,184,000 distinct routes that would need to be evaluated. Rather than enumerating all possibilities, successful algorithms for solving the TSP problem eliminate most of the routes without ever explicitly considering them.

Formulations

The first step to solving instances of large TSPs must be to find a good mathematical formulation of the problem. In the case of the traveling salesman problem, the mathematical structure is a graph where each city is denoted by a point (or node) and lines are drawn connecting every two nodes (called arcs or edges). Associated with every line is a distance (or cost). When the salesman can get from every city to every other city directly, then the graph is said to be complete. A round-trip (route) of the cities corresponds to some subset of the lines, and is called a tour or a Hamiltonian cycle in graph theory. The length of a tour is the sum of the lengths of the lines in the round-trip.

Depending upon whether or not the direction in which an edge of the graph is traversed matters, one distinguishes the asymmetric from the symmetric traveling salesman problem. To formulate the asymmetric TSP on m cities, one introduces zero-one variables

$$x_{ij} = \begin{cases} 1 \text{ if the edge } i \to j \text{ is in the tour} \\ 0 \text{ otherwise} \end{cases}$$

and, given the fact that every node of the graph must have exactly one edge pointing towards it and one pointing away from it, one obtains the classic assignment problem. These constraints alone are not enough since this formulation would allow subtours, i.e., it would allow disjoint loops to occur. For this reason, a proper formulation of the asymmetric traveling salesman problem must remove these subtours from consideration by the addition of subtour-elimination constraints. The problem then becomes

$$\begin{aligned} \min \quad & \sum_{j=1}^{m} \sum_{i=1}^{m} c_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{j=1}^{m} x_{ij} = 1 \quad \text{for } i = 1, \ldots, m \\ & \sum_{j=1}^{m} x_{ij} = 1 \quad \text{for } j = 1, \ldots, m \\ & \sum_{i \in K} \sum_{j \in K} x_{ij} \leq |K| - 1 \quad \text{for all } K \subset \{1, \ldots, m\} \end{aligned}$$

where K is any nonempty proper subset of the cities $1, \ldots, m$. The cost c_{ij} is allowed to be different from the cost c_{ji}. Note that there are $m(m-1)$ 0–1 variables in this formulation.

To formulate the symmetric traveling salesman problem, one notes that the direction traversed is immaterial, so that $c_{ij} = c_{ji}$. Since direction does not now matter, one can consider the graph where there is only one arc (undirected) between every two nodes. Thus, let $x_j \in \{0,1\}$ be the decision variable where j runs through all edges E of the undirected graph and c_j is the cost of traveling that edge. To find a tour in this graph, one must select a subset of edges such that every node is contained in exactly two of the edges selected. Thus, the problem can be formulated as a 2-matching problem in a graph having $m(m-1)/2$ 0–1 variables, that is, half of the number of the previous formulation. As in the asymmetric case, subtours must be eliminated through subtour elimination constraints. The problem can therefore be formulated as

$$\begin{aligned} \min \quad & (1/2) \sum_{j=1}^{m} \sum_{k \in J_{(j)}} c_k x_k \\ \text{s.t.} \quad & \sum_{k \in J_{(j)}} x_k = 2 \quad \text{for all } j = 1, \ldots, m \\ & \sum_{j \in E(K)} x_j \leq |K| - 1 \quad \text{for all } K \subset \{1, \ldots, m\} \\ & x_j = 0 \text{ or } 1 \quad \text{for all } j \in E, \end{aligned}$$

where $J(j)$ is the set of all undirected edges connected to node j and $E(K)$ is the subset of all undirected edges

connecting the cities in any proper, nonempty subset K of all cities. Of course, the symmetric problem is a special case of the asymmetric one, but practical experience has shown that algorithms for the asymmetric problem perform, in general, badly on symmetric problems. Thus, the latter need a special formulation and solution treatment. In addition, as an ATSP instance can be easily turned into a symmetric one with twice the number of nodes, any algorithm for STSP can be used to solve an ATSP.

Algorithms

Exact approaches to solving such problems require algorithms that generate both a lower bound and an upper bound on the true minimum value of the problem instance. Any round-trip tour that goes through every city exactly once is a feasible solution with a given cost that cannot be smaller than the minimum cost tour. Algorithms that construct feasible solutions, and thus upper bounds for the optimum value, are called heuristics. These solution strategies produce answers but often without any quality guarantee as to how far off they may be from the optimal answer. Heuristic algorithms that find a feasible solution in a single attempt are called constructive heuristics, while algorithms that iteratively modify and try to improve some given starting solution are called improvement heuristics. When the solution one obtains is dependent on the initial starting point of the algorithm, the same algorithm can be used multiple times from various (random) starting points. Often, if one needs a solution quickly, one may settle for a well-designed heuristic algorithm that has been shown empirically to find near-optimal tours to many TSP problems. Research by Golden and Stewart (1985), Jünger, Reinelt and Rinaldi (1994), Johnson and McGeoch (2002), and Applegate et al. (2006) describes algorithms that find solutions to extremely large TSPs (problems with hundreds of thousands, or even millions of variables) to within 1 or 2% of optimality in very reasonable times. The heuristic algorithm of Lin and Kernighan appears so far to be the most effective in term of solution quality, in particular with the variant proposed by Helsgaun (2000), which was able to find, for the first time, the optimal solution (although without a quality guarantee) of several instances of TSPLIB, a well known library of TSP problems described in Reinelt (1991). For genetic algorithmic approaches to the TSP, see Potvin (1996); for simulated annealing approaches see Aarts, Korst and Laarhoven (1988); for neural net approaches, see Potvin (1993); for tabu search approaches, see Fiechter (1990); and for a very effective evolutionary algorithm, *see* Nagata (2006). Probabilistic analysis of heuristics are discussed in Karp and Steele (1985); performance guarantees for heuristics are given in Johnson and Papadimitriou (1985) and Arora (2002), where an amazing result concerning the polynomial-time approximability is described for Euclidean TSP instances (where the nodes are points in the plane and the traveling costs are the Euclidean distances between the points). For an analysis of the heuristics for the ATSP, see Johnson et al. (2002).

In order to know about the closeness of the upper bound to the optimum value, one must also know a lower bound on the optimum value. If the upper and lower bound coincide, a proof of optimality is achieved. If not, a conservative estimate of the true relative error of the upper bound is provided by the difference of the upper and the lower bound divided by the lower bound. Thus, one needs both upper and lower bounding techniques to find provably optimal solutions to hard combinatorial problems or even to obtain solutions meeting a quality guarantee.

So how does one obtain and improve the lower bound? A relaxation of an optimization problem is another optimization problem whose set of feasible solutions properly contains all feasible solution of the original problem and whose objective function value is less than or equal to the true objective function value for points feasible to the original problem. Thus, the true problem is replaced by one with a larger feasible region but that is more easily solvable. This relaxation is continually refined so as to tighten the feasible region so that it more closely represents the true problem. The standard technique for obtaining lower bounds on the TSP problem is to use a relaxation that is easier to solve than the original problem. These relaxations can have either discrete or continuous feasible sets. Several relaxations have been considered for the TSP. Among them are the n-path relaxation, the assignment relaxation, the 2-matching relaxation, the 1-tree relaxation, and the linear programming relaxation. For randomly generated asymmetric TSPs, problems having up to 7,500 cities

have been solved, in the early 1990s, using an assignment relaxation which adds subtours within a branch and bound framework and which uses an upper bounding heuristic based on subtour patching, (Miller and Pekny 1991). For the symmetric TSP, the 1-tree relaxation and the 2-matching relaxations have been most successful. These relaxations have been embedded into a branch-and-bound framework.

The process of finding constraints that are violated by a given relaxation is called a cutting plane technique and all successes for large TSP problems have used cutting planes to continuously tighten the formulation of the problem. To obtain a tight relaxation the inequalities utilized as cutting planes in many computational approaches to the TSP are often facet-defining inequalities.

One of the simplest classes of cuts that have been shown to define facets of the underlying TSP polytope is the subtour elimination cut. Besides these constraints, comb inequalities, clique tree inequalities, path, wheelbarrow and bicycle inequalities, ladder, crown, domino and many other inequalities have also been shown to define facets of this polytope. The underlying theory of facet generation for the symmetric traveling salesman problem is provided in Grötschel and Padberg (1985), Jünger, Reinelt and Rinaldi (1994) *and* Naddef (2002); analogous results for the ATSP polytope are provided in Balas and Fischetti (2002). The algorithmic descriptions of how these inequalities are used in cutting plane approaches are discussed in Padberg and Rinaldi (1991), in Jünger, Reinelt and Rinaldi (1994), *and in* Applegate et al. (2006) where it is also shown how the polynomial-time equivalence between optimization and separation can be turned into a powerful algorithmic tool to generate inequalities not necessarily belonging to one of the known types.

Cutting plane procedures can then be embedded into a tree search in an algorithmic framework referred to as branch and cut and proposed in Padberg and Rinaldi (1991), where it is shown how such approach made it possible to solve some still unsolved instances of sizes up to 2,392 nodes. Some of the largest TSP problems solved have used parallel processing to assist in the search for optimality. This is the case of the software Concorde, where all the known algorithmic ideas for the TSP (and many new ones) have been carefully implemented. With this code, Applegate et al. (2006) managed to solve all problems of the TSPLIB to optimality; for the largest one, of 85,900 nodes, they used 96 workstations for a total of 139 years of CPU time.

As understanding of the underlying mathematical structure of the TSP problem improves, and with the continuing advancement in computer technology, it is likely that many difficult and important combinatorial optimization problems will be solved using a combination of cutting plane generation procedures, heuristics, variable fixing through logical implications and reduced costs, and tree search.

Applications

One might ask, however, whether the TSP problem is important enough to have received all of the attention it has. Much of the attention that the problem has received is because it is a relatively simple problem to describe and yet a difficult (from a complexity viewpoint) optimization problem to solve. However, there are important cases of practical problems that can be formulated as TSP problems and many other problems are generalizations of this problem. Besides the drilling of printed circuits boards described above, problems having the TSP structure occur in the analysis of the structure of crystals (Bland and Shallcross 1987), in the overhauling of gas turbine engines (Pante et al. 1987), in material handling in a warehouse (Ratliff and Rosenthal 1981), in cutting stock problems (Garfinkel, 1977), in the clustering of data arrays (Lenstra and Rinooy Kan 1975), in the sequencing of jobs on a single machine (Gilmore and Gomory 1964), in the assignment of routes for planes of a specified fleet (Boland et al. 1994) and in genome sequencing (Ben-Dor and Chor 1997; Ben-Dor et al. 2000). Related variations on the traveling salesman problem include the resource-constrained traveling salesman problem, which has applications in scheduling with an aggregate deadline (Pekny and Miller 1991). This paper also shows how the prize collecting traveling salesman problem (Balas 2002) and the orienteering problem (Golden et al. 1987; Fischetti et al. 2002) are special cases of the resource constrained TSP. Most importantly, the traveling salesman problem often comes up as a subproblem in more complex combinatorial problems, perhaps

the best-known application being the vehicle routing problem. This is the problem of determining for a fleet of vehicles which customers should be served by each vehicle and in what order each vehicle should visit the customers assigned to it. For relevant surveys, see Christofides (1985), Fisher (1987), and the book *The Vehicle Routing Problem*, edited by Toth and Vigo (2001).

Concluding Remarks

The seminal paper on the TSP is Dantzig, Fulkerson and Johnson (1954). Books by Lawler et al. (1985), Reinelt (1994) and Gutin and Punnen (2002), and the survey and annotated bibliography by Jünger, Reinelt and Rinaldi (1994, 1997), summarize most of the research up through 2002 and provide extensive references. For a deep understanding of how algorithms for TSP work, see the book by Applegate et al. (2006), which besides providing a wide overview on TSP history and on its applications, also gives a detailed description of how all the components of the Concorde software are built: a valuable source for algorithm designers. Finally, the book by Cook (2011) is for a more general audience, requiring almost no mathematical background to read, but very nicely and completely describing the TSP from several interesting viewpoints. The computer program Concorde, the TSPLIB, and many other sources of information on the TSP are available electronically at a Web site that can be easily located through Web search.

See

- ▶ Assignment Problem
- ▶ Branch and Bound
- ▶ Chinese Postman Problem
- ▶ Integer and Combinatorial Optimization
- ▶ Combinatorics
- ▶ Computational Complexity
- ▶ Graph Theory
- ▶ Heuristics
- ▶ Linear Programming
- ▶ Network
- ▶ *NP*, *NP*-Complete, *NP*-Hard
- ▶ Tabu Search

References

Aarts, E. H. L., Korst, J. H. M., & Laarhoven, P. J. M. (1988). A quantitative analysis of the simulated annealing algorithm: A case study for the traveling salesman problem. *Journal of Statistical Physics, 50*, 189–206.

Applegate, D., Bixby, R. E., Chvátal, V., & Cook, W. (1995). *Finding cuts in the TSP(A preliminary report) DIMACS* (Technical Report 95–05). New Brunswick, USA: Rutgers University.

Applegate, D., Bixby, R. E., Chvátal, V., & Cook, W. (2006). *The traveling salesman problem: A computational study*. Princeton: Princeton University Press.

Arora, S. (2002). Approximation algorithms for geometric TSP. In G. Gutin & A. P. Punnen (Eds.), *The traveling salesman problem and its variations* (pp. 207–222). Dordrecht, The Netherlands: Kluwer.

Balas, E. (2002). The prize collecting traveling salesman problem and its applications. In G. Gutin & A. P. Punnen (Eds.), *The traveling salesman problem and its variations* (pp. 663–696). Dordrecht, The Netherlands: Kluwer.

Balas, E., & Fischetti, M. (2002). Polyhedral theory for the asymmetric traveling salesman problem. In G. Gutin & A. P. Punnen (Eds.), *The traveling salesman problem and its variations* (pp. 117–168). Dordrecht, The Netherlands: Kluwer.

Ben-Dor, A., & Chor, B. (1997). On constructing radiation hybrid maps. *Journal of Computational Biology, 4*, 517–533.

Ben-Dor, A., Chor, B., & Pelleg, D. (2000). RHO-radiation hybrid ordering. *Genome Research, 10*, 365–378.

Bland, R. E., & Shallcross, D. F. (1987). *Large traveling salesman problem arising from experiments in X-ray crystallography: A preliminary report on computation* (Technical Report No. 730). Ithaca, New York: School of OR/ IE, Cornell University.

Burkard, R. E., Deineko, V. G., van Dal, R., van der Veen, J. A. A., & Woeginger, G. J. (1998). Well-solvable cases of the traveling salesman problem: A survey. *SIAM Review, 40*, 496–546.

Cook, W. (2011). *In pursuit of the salesman: Mathematics at the limits of computation*. Princeton: Princeton University Press.

Dantzig, G. B., Fulkerson, D. R., & Johnson, S. M. (1954). Solution of a large-scale traveling salesman problem. *Operations Research, 2*, 393–410.

Fiechter, C. N. (1990). *A parallel tabu search algorithm for large scale traveling salesman problems* (Working Paper 90/1). Switzerland: Department of Mathematics, Ecole Polytechnique Federale de Lausanne.

Fisher, M. L. (1988). Lagrangian optimization algorithms for vehicle routing problems. In G. K. Rand (Ed.), *Operational research '87*, pp. 635–649.

Golden, B. L., Levy, L., & Vohra, R. (1987). The orienteering problem. *Naval Research Logistics, 34*, 307–318.

Golden, B. L., & Stewart, W. R. (1985). Empirical analysis of heuristics. In E. L. Lawler, J. K. Lenstra, A. H. G. Rinooy Kan, & D. B. Shmoys (Eds.), *The traveling salesman problem* (pp. 207–250). Chichester: John Wiley.

Grötschel, M., & Holland, O. (1991). Solution of large scale symmetric traveling salesman problems. *Mathematical Programming, 51*, 141–202.

Grötschel, M., & Padberg, M. W. (1985). Polyhedral theory. In E. L. Lawler, J. K. Lenstra, A. H. G. Rinooy Kan, & D. B. Shmoys (Eds.), *The traveling salesman problem* (pp. 251–306). Chichester: John Wiley.
Gutin, G., & Punnen, A. P. (Eds.). (2002). *The traveling salesman problem and its variations*. Dordrecht, The Netherlands: Kluwer.
Helsgun, K. (2000). An effective implementation of the Lin-Kernighan traveling salesman heuristic. *European Journal of Operational Research, 126*, 106–130.
Hoffman, A. J., & Wolfe, P. (1985). History. In E. L. Lawler, J. K. Lenstra, A. H. G. Rinooy Kan, & D. B. Shmoys (Eds.), *The traveling salesman problem* (pp. 1–16). Chichester: John Wiley.
Johnson, D. S., & McGeoch, L. A. (2002). Experimental analysis of heuristics for the STSP. In G. Gutin & A. P. Punnen (Eds.), *The traveling salesman problem and its variations* (pp. 369–444). Dordrecht: Kluwer.
Johnson, D. S., & Papadimitriou, C. H. (1985). Performance guarantees for heuristics. In E. L. Lawler, J. K. Lenstra, A. H. G. Rinooy Kan, & D. B. Shmoys (Eds.), *The traveling salesman problem* (pp. 145–180). Chichester: John Wiley.
Johnson, D. S., Gutin, G., McGeoch, L. A., Yeo, A., Zhang, W., & Zverovitch, A. (2002). Experimental analysis of heuristics for the ATSP. In G. Gutin & A. P. Punnen (Eds.), *The traveling salesman problem and its variations* (pp. 485–488). Dordrecht, The Netherlands: Kluwer.
Jünger, M., Reinelt, G., & Rinaldi, G. (1994). The traveling salesman problem. In M. Ball, T. Magnanti, C. Monma, & G. Nemhauser (Eds.), *Handbook on operations research and the management sciences* (pp. 225–330). Amsterdam: North Holland.
Jünger, M., Reinelt, G., & Rinaldi, G. (1997). The traveling salesman problem. In M. Dell'Amico, F. Maffioli, & S. Martello (Eds.), *Annotated bibliographies in combinatorial optimization* (pp. 199–221). New York: Wiley.
Karp, R., & Steele, J. M. (1985). Probabilistic analysis of heuristics. In E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, & D. B. Shmoys (Eds.), *The traveling salesman problem* (pp. 181–205). Chichester: John Wiley.
Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G., & Shmoys, D. B. (Eds.). (1985). *The traveling salesman problem*. Chichester, UK: John Wiley.
Miller, D., & Pekny, J. (1991). Exact solution of large asymmetric traveling salesman problems. *Science, 251*, 754–761.
Naddef, D. (2002). Polyhedral theory and branch-and-cut algorithm for the symmetric TSP. In G. Gutin & A. P. Punnen (Eds.), *The traveling salesman problem and its variations* (pp. 21–116). Dordrecht, The Netherlands: Kluwer.
Nagata, Y. (2006). New EAX crossover for large TSP instances. *Lecture Notes in Computer Science, 4193*, 372–381.
Padberg, M. W., & Grötschel, M. (1985). Polyhedral computations. In E. L. Lawler, J. K. Lenstra, A. H. G. Rinooy Kan, & D. B. Shmoys (Eds.), *The traveling salesman problem* (pp. 307–360). Chichester: John Wiley.
Padberg, M. W., & Rinaldi, G. (1991). A branch and cut algorithm for the resolution of large-scale symmetric traveling salesmen problems. *SIAM Review, 33*, 60–100.
Potvin, J. V. (1993). The traveling salesman problem: A neural network perspective. *INFORMS Journal on Computing, 5*, 328–348.
Potvin, J. V. (1996). Genetic algorithms for the traveling salesman problem. *Annals of Operations Research, 63*, 339–370.
Ratliff, H. D., & Rosenthal, A. S. (1981). *Order-picking in a rectangular warehouse: A solvable case for the traveling salesman problem* (PDRC Report Series No. 81–10). Atlanta: Georgia Institute of Technology.
Reinelt, G. (1991). TSPLIB–A traveling salesman library. *ORSA Journal on Computing, 3*, 376–384.
Reinelt, G. (1994). *The traveling salesman: Computational solutions for TSP applications*. Berlin: Springer-Verlag.
Toth, P., & Vigo, D. (2001). *The vehicle routing problem*. Philadelphia: SIAM.

Tree

In a network, a tree is a subnetwork (graph) that has no cycles and connects all nodes of a subnetwork, that is, a unique path exists between each node. A tree that connects all n nodes of a network is called a spanning tree and has $(n - 1)$ arcs.

See

► Minimum Spanning Tree Problem
► Network Optimization

Triangular Matrix

A square matrix $\boldsymbol{A} = (a_{ij})$ such that either all the elements a_{ij} above the diagonal are 0 or all the elements below the diagonal are 0. The former is called a lower triangular matrix and the latter an upper triangular matrix.

Trim Problem

Problem of determining how rolls or sheets of material should be cut to minimize the amount of wasted material (trim) while meeting the demand for different sizes of cuts. The problem originally arose in the context of cutting large rolls of newsprint into desired smaller sizes. The trim problem can be formulated and solved as a linear or

integer program. It was the problem that motivated column generation procedures.

See

- ▶ Column Generation
- ▶ Cutting Stock Problems

Trivial Solution

For the homogeneous linear equations $\boldsymbol{Ax} = 0$, the solution $\boldsymbol{x} = 0$ is called a trivial solution.

See

- ▶ Nontrivial Solution
- ▶ Null Space

Truck Dispatching

The dynamic assignment of trucks (drivers) to loads and/or customers.

See

- ▶ Logistics and Supply Chain Management
- ▶ Vehicle Routing

Truckload (TL) Shipment

A shipment weighing at least the minimum weight to qualify for a TL-size rate reduction.

See

- ▶ Logistics and Supply Chain Management

TS

- ▶ Tabu Search

TSP

- ▶ Traveling Salesman Problem

Tucker Tableau

A reduced simplex tableau of a linear-programming problem that considers the tableau as representation of both the primal and dual problems.

Two-Phase Simplex Method

Any version of the simplex method that requires the finding of a first basic feasible solution using artificial variables (Phase I) and then the finding of an optimal feasible solution (Phase II).

See

- ▶ Artificial Variables
- ▶ Phase I Procedure
- ▶ Phase II Procedure

U

Unary *NP*-Complete (*NP*-Hard)

► Computational Complexity

Unbalanced Transportation Problem

A transportation problem in which the total amount to be shipped (supply) is not equal to the total demand. The unbalanced problem can be stated as a standard transportation problem by the addition of a fictitious destination when the supply is greater than the demand, or by adding a fictitious origin if the demand is greater than the supply. In the first case, the demand at the fictitious destination is the difference between the total supply and total demand, while in the second case, the supply at the fictitious origin is the difference between the total demand and total supply.

See

► Transportation Problem

Unbounded Optimal Solution

A solution to a constrained optimization problem in which the objective function value can be shown to increase (or decrease) without bound on the feasible region. A real-world problem whose mathematical model exhibits an unbounded optimal solution must have an incorrect formulation.

Unconstrained Optimization

Ariela Sofer
George Mason University, Fairfax, VA, USA

Introduction

Unconstrained optimization is concerned with finding the minimizing or maximizing points of a nonlinear function, where the variables are free to take on any value. Unconstrained optimization problems occur in a wide range of applications in science and engineering. A rich source of unconstrained optimization problems are data-fitting problems, in which some model function with unknown parameters is fitted to data, using some criterion of best fit. This criterion may be the minimum sum of squared errors, or the maximum of a likelihood or entropy function. Unconstrained problems also arise from constrained optimization problems, since these are often solved by solving a sequence of unconstrained problems.

In mathematical terms, an unconstrained minimization problem can be written in the form

$$\text{minimize } f(\boldsymbol{x}),$$

where $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ is a vector of unrestricted variables in the n-dimensional space $\Re^n$. Ideally, one would like to find a global minimizer of the function, i.e., a point $\boldsymbol{x}^*$ that yields the lowest value of f. Such a solution satisfies

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x}.$$

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

If the inequality above holds strictly, i.e., $f(\boldsymbol{x}^*) < f(\boldsymbol{x})$ for all $\boldsymbol{x}$, then $\boldsymbol{x}^*$ is a strict global minimizer.

In many cases, finding a global minimizer is extremely difficult. For this reason, most algorithms attempt only to find a local minimizer of the function, i.e., a point $\boldsymbol{x}^*$ that satisfies $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all $\boldsymbol{x}$ in some neighborhood of $\boldsymbol{x}^*$. If the objective f is a convex function (see the next section), a local minimizer will also be a global minimizer; however, for nonconvex functions this property does not generally hold.

There is no inherent difference between minimization problems and maximization problems, since maximizing f can be accomplished by minimizing $-f$ and then multiplying the optimal objective values by -1. For this reason it is sufficient to focus only on unconstrained minimization problems.

Background

Much of the research in unconstrained optimization has focused on functions with continuous derivatives. Throughout this discussion it is assumed that the objective function f is twice-continuously differentiable (i.e., its second partial derivatives exist and are continuous). The gradient of f at $\boldsymbol{x}$, denoted by $\nabla f(\boldsymbol{x})$, is the vector of first partial derivatives $\partial f(\boldsymbol{x})/\partial x_j$, and the Hessian of f at $\boldsymbol{x}$, denoted by $\nabla^2 f(\boldsymbol{x})$, is the matrix of second partial derivatives $\partial^2 f(\boldsymbol{x})/\partial x_i \partial x_j$. When f is twice-continuously differentiable, the Hessian matrix is symmetric.

If there is a single fundamental tool in optimization of differentiable functions, it is the Taylor series, which provides an approximation to the function in a neighborhood of a point. The Taylor series is used in the derivation of the optimality conditions, in the development of solution methods and in analysis of their convergence.

Let $\bar{\boldsymbol{x}}$ be a given point, and suppose that $\boldsymbol{p}$ is some direction in $\Re^n$. The first-order Taylor series expansion of f at $\bar{\boldsymbol{x}}$ is

$$f(\bar{\boldsymbol{x}} + \boldsymbol{p}) = f(\bar{\boldsymbol{x}}) + \boldsymbol{p}^T \nabla f(\bar{\boldsymbol{x}}) + O(\|\boldsymbol{p}\|)^2,$$

where $O(q)$ indicates a term that goes to zero at least as fast as q does. Ignoring the last term in the expansion leads to a linear approximation to f in a neighborhood of $\bar{x}$; the error will be of order $O(\|\boldsymbol{p}\|)^2$. Similarly, the second-order Taylor series expansion of f is given by

$$f(\bar{\boldsymbol{x}} + \boldsymbol{p}) = f(\bar{\boldsymbol{x}}) + \boldsymbol{p}^T \nabla f(\bar{\boldsymbol{x}}) + \tfrac{1}{2}\boldsymbol{p}^T \nabla^2 f(\bar{\boldsymbol{x}})\boldsymbol{p} + O(\|\boldsymbol{p}\|)^3.$$

Ignoring the last term in this expansion leads to a quadratic approximation to f, with an error of order $O(\|\boldsymbol{p}\|)^3$. This approximation is referred to as the quadratic model.

The quantity $\boldsymbol{p}^T \nabla f(\bar{\boldsymbol{x}})$ is called the directional derivative of f along $\boldsymbol{p}$ at $\bar{\boldsymbol{x}}$. If it is negative, then p is termed a direction of descent. A small step $\varepsilon > 0$ taken in such a direction will lead to a point with a lower objective value: $f(\bar{\boldsymbol{x}} + \varepsilon \boldsymbol{p}) < f(\bar{\boldsymbol{x}})$. The quantity $\boldsymbol{p}^T \nabla^2 f(\bar{\boldsymbol{x}})\boldsymbol{p}$ is called the curvature of f along $\boldsymbol{p}$. If the curvature is positive, the function is locally convex along the direction $\boldsymbol{p}$ at $\bar{\boldsymbol{x}}$.

Convexity of the objective function is a desirable property in unconstrained minimization. Geometrically, it means that the function is locally convex in every direction. The formal definition does not actually require the function to be differentiable. A function f is defined to be convex if it satisfies

$$f(\alpha \boldsymbol{x} + (1-\alpha)\boldsymbol{y}) \leq \alpha f(\boldsymbol{x}) + (1-\alpha) f(\boldsymbol{y})$$

for all $0 \leq \alpha \leq 1$ and for all $\boldsymbol{x}, \boldsymbol{y}$. The function is strictly convex if this inequality is strict (for $0 < \alpha < 1$ and $\boldsymbol{x} \neq \boldsymbol{y}$). A function f is concave if $-f$ is convex.

If f is twice continuously differentiable function, f is convex if and only if its Hessian $\nabla^2 f(\boldsymbol{x})$ is positive semidefinite for all $\boldsymbol{x}$. This means that $\boldsymbol{p}^T \nabla^2 f(\boldsymbol{x})\boldsymbol{p} \geq 0$ for all $\boldsymbol{p}$, so the function is locally convex along every direction $\boldsymbol{p}$. If the Hessian $\nabla^2 f(\boldsymbol{x})$ is positive definite for all x, then the function f is strictly convex. However the Hessian of a strictly convex function need not be positive definite everywhere, as is demonstrated by $f(x) = x^4$ at the origin $x = 0$.

Convexity is an attractive property since any local minimizer of an unconstrained convex function is also a global minimizer of the function. Furthermore, any local minimizer of an unconstrained strictly convex function is also the unique global minimizer of the function.

Optimality Conditions

Using the Taylor series approximation, it is possible to derive conditions that must be satisfied by a local

minimizer $\boldsymbol{x}^*$ of f. The conditions state that the function must have zero slope and nonnegative curvature along any direction at $\boldsymbol{x}^*$, which are summarized in the necessary conditions. The first-order necessary condition states that the gradient at $\boldsymbol{x}^*$ must vanish, so that

$$\nabla f(\boldsymbol{x}^*) = 0.$$

The second-order necessary condition states the Hessian must be positive semidefinite, so that

$$\boldsymbol{p}^T \nabla^2 f(\bar{\boldsymbol{x}}) \boldsymbol{p} \ \forall \boldsymbol{p}.$$

(In the case of a local maximizer, the Hessian must be negative semidefinite).

To illustrate these conditions, consider the two-dimensional function $f(\boldsymbol{x}) = x_1^2 + x_2^2$. The function attains its minimum at $x^* = (0,0)^T$. The gradient of f is $\nabla f(\boldsymbol{x}) = (2x_1, 2x_2)^T$, and indeed vanishes at $\boldsymbol{x}^*$; the Hessian at $\boldsymbol{x}^*$ is twice the identity matrix, and hence is positive definite. Thus, the necessary conditions for a minimizer are satisfied at $\boldsymbol{x}^*$.

A point at which the gradient is equal to zero is called a stationary point. Although such a point may be a local minimizer, it may also be a local maximizer, or neither of the above (in such case it is called a saddle point). As an example, $\boldsymbol{x}^* = (0,0)^T$ is a stationary point of the functions $f_1(\boldsymbol{x}) = -x_1^2 - x_2^2$ and $f_2(\boldsymbol{x}) = x_1^2 - x_2^2$; it is a local maximizer of f_1, and a saddle point for f_2.

It is possible to develop a condition that guarantees that a stationary point is a local minimizer. The second-order sufficiency condition states that if

$$\nabla f(\boldsymbol{x}^*) = 0, \text{and } \boldsymbol{p}^T \nabla^2 f(\bar{\boldsymbol{x}}) \boldsymbol{p} > 0 \ \forall \boldsymbol{p},$$

then $\boldsymbol{x}^*$ is a strict local minimizer of f.

As an example, consider the quadratic function

$$f(\boldsymbol{x}) = \tfrac{1}{2} \boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} + \boldsymbol{c}^T \boldsymbol{x},$$

where $\boldsymbol{Q}$ is a symmetric invertible matrix. Any stationary point must satisfy $\nabla f(\boldsymbol{x}) = \boldsymbol{Q}\boldsymbol{x} + \boldsymbol{c} = 0$. Since $\boldsymbol{Q}$ is invertible there is a unique solution $\boldsymbol{x}^* = -\boldsymbol{Q}^{-1}\boldsymbol{c}$. The point $\boldsymbol{x}^*$ is a strict local (and global)minimizer of f if $\boldsymbol{Q}$ is positive definite, and a strict local (and global) maximizer of f if $\boldsymbol{Q}$ is negative definite. It is a saddle point if $\boldsymbol{Q}$ is indefinite.

Methods

The vast majority of algorithms for unconstrained minimization are iterative descent methods. At each iteration, a direction of descent (called the search direction) is computed at the current solution estimate $\boldsymbol{x}_k$; a step is then taken from $\boldsymbol{x}_k$ alongthe search direction, to obtain a new point a new point $\boldsymbol{x}_{k+1}$ such that $f(\boldsymbol{x}_{k+1}) < f(\boldsymbol{x}_k)$. The process is repeated till some test for convergence is satisfied. The effectiveness of an algorithm is, of course, dramatically affected by the choice of the search direction. A key question, of course, is how to obtain a good search direction. The underlying idea of most methods, is to compute a direction that minimizes some local approximation to the function. Typically, this local model is obtained from the Taylor series.

In Newton's method, the search direction at the current point $\boldsymbol{x}_k$ is the vector $\boldsymbol{p}_k$ that minimizes the local quadratic model:

$$\underset{\boldsymbol{p}}{\text{minimize}} \ f(\boldsymbol{x}_k) + \boldsymbol{p}^T \nabla f(\boldsymbol{x}_k) + \boldsymbol{p}^T \nabla^2 f(\boldsymbol{x}_k) \boldsymbol{p}.$$

If the Hessian $\nabla^2 f(\boldsymbol{x}_k)$ is positive definite, the minimizer of the quadratic model is the solution to the linear system of equations

$$\nabla^2 f(\boldsymbol{x}_k) \boldsymbol{p} = -\nabla f(\boldsymbol{x}_k),$$

known as the Newton equations. The resulting iteration takes the form

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{p}_k,$$

where $\boldsymbol{p}_k$ is the solution to the Newton equations.

If the initial point $\boldsymbol{x}_0$ is sufficiently close to a local minimizer $\boldsymbol{x}^*$, and if $\nabla^2 f(\boldsymbol{x}^*)$ is positive definite, then under mild conditions the iterates generated by Newton's method converge to $\boldsymbol{x}^*$. Furthermore, the rate of convergence is quadratic. This means that for large k, the error at an iteration is proportional to the square of the error in the previous iteration:

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq \gamma \|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2$$

for some positive constant γ. Roughly this means that towards the end, the number of significant digits in the iterates double at each iteration. The mild conditions

mentioned are requirements that ensure the Hessian matrix does not fluctuate wildly. A commonly used condition is Lipschitz continuity of the Hessian, namely that

$$\left\|\nabla^2 f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{y})\right\| \leq L\|\boldsymbol{x}-\boldsymbol{y}\|$$

for all $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\Re^n$ and some finite constant L.

The rapid convergence of Newton's method near the solution makes it an extremely attractive method, and indeed, the method can be highly effective. However, the algorithm may fail if started from an initial point that is not sufficiently close to a minimizer. Why? First, if the Hessian $\nabla^2 f(\boldsymbol{x}_k)$ is not positive definite, the Newton direction may not be a descent direction, and if the Hessian is singular, the method is not even defined. Second, even if $\boldsymbol{p}_k$ is a descent direction, there is no guarantee that $f(\boldsymbol{x}_{k+1})$ will actually be lower than $f(\boldsymbol{x}_k)$. Thus, modifications to the basic Newton method are required to guarantee that the method will converge regardless of the starting point.

There are two major approaches to guarantee global convergence (convergence from any initial point): line search methods and trust-region methods. Both approaches use the basic Newton method near the solution to exploit its rapid local convergence property. But they differ in the strategies they employ to guarantee convergence when far from the solution. Both approaches insist, however, on using a descent direction at each iteration.

Line search methods update the new estimate of the solution as $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k$, where the step length α_k is a positive scalar chosen so that $f(\boldsymbol{x}_{k+1}) < f(\boldsymbol{x}_k)$. Ideally, this step length would be chosen to minimize $f(\boldsymbol{x}_k + \alpha \boldsymbol{p}_k)$ with respect to α. However, finding such a step length is too time consuming. A more practical approach is to use a step length that approximately minimizes f along $\boldsymbol{p}_k$. One commonly used condition (known as the Wolfe condition) is to accept a trial step α_k if

$$\left|\boldsymbol{p}_k^T \nabla f(\boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k)\right| \leq \theta \left|\boldsymbol{p}_k^T \nabla f(\boldsymbol{x}_k)\right|,$$

where θ is some scalar satisfying $0 < \theta < 1$, i.e., if a step of length α_k taken along $\boldsymbol{p}_k$ yields a substantial decrease in the magnitude of the directional derivative. This condition alone cannot guarantee convergence, since it does not guarantee decrease in the objective value. It is therefore common to impose an additional sufficient decrease condition on α_k:

$$f(\boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k) \leq f(\boldsymbol{x}_k) + \eta \alpha_k \boldsymbol{p}_k^T \nabla f(\boldsymbol{x}_k)$$

where $0 < \eta < 1$. This is known as the Armijo condition. If the Wolfe and Armijo conditions are used in tandem, and if $\eta < \theta$, then under appropriate conditions, global convergence of the algorithm is guaranteed.

Line search versions of Newton's method must also incorporate some strategy to handle the case when the Hessian is not positive definite. One standard technique is to modify the Hessian matrix by a diagonal matrix, denoted $\boldsymbol{E}_k$, whose diagonal componentsare large enough to ensure that the modified Hessian is indeed positive definite. The modified Newton direction is then computed as the solution to the system

$$\boldsymbol{B}_k \boldsymbol{p}_k = -\nabla f(\boldsymbol{x}_k),$$

where

$$\boldsymbol{B}_k = \nabla^2 f(\boldsymbol{x}_k) + \boldsymbol{E}_k.$$

The approach generates descent directions and can overcome the numerical difficulties associated with near-singular Hessians.

Trust region methods differ from line search methods in that they determine a priori the maximum length of the search direction, say Δ. The direction is taken as the minimizer of the quadratic model, whose length does not exceed Δ for the trial step $\boldsymbol{p}_k$:

$$\underset{\boldsymbol{p}}{\text{minimize}}\ q(\boldsymbol{p}) = f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^T \boldsymbol{p} + \tfrac{1}{2}\boldsymbol{p}^T \nabla^2 f(\boldsymbol{x}_k)\boldsymbol{p}$$
$$\text{subject to } \|\boldsymbol{p}\| \leq \Delta.$$

The motivation for this approach is that the quadratic model obtained from the Taylor series gives an adequate fit to the function for points that are close to $\boldsymbol{x}_k$, but may not give an adequate fit for points far away.

The length Δ is the radius of the trust region, the region in which the quadratic model is trusted. It is adjusted from iteration to iteration, based on the agreement between the function f and the quadratic model. It is increased if the agreement is considered

to be good, and decreased if it is considered to be poor. The criterion used for determining this is the value of

$$\rho = \frac{f(\boldsymbol{x}_k) - f(\boldsymbol{x}_k + \boldsymbol{p}_k)}{f(\boldsymbol{x}_k) - q(\boldsymbol{p}_k)},$$

which is the ratio of actual reduction in the function value to that predicted by the quadratic model. If this ratio is large, it is assumed that the quadratic model can be trusted in a wider region and Δ is increased. If it is small, the quadratic model is deemed inadequate (hence the model cannot be trusted) and Δ is decreased.

Global convergence is achieved under mild conditions. If f is twice continuously differentiable and the set $\{\boldsymbol{x} : f(\boldsymbol{x}) \leq f(\boldsymbol{x}_0)\}$ is bounded, the method converges to a stationary point. In practice, however, the solution to the trust region problem is relatively expensive and often only approximate solutions are attempted.

Modified Newton's methods (both line-search and trust-region variants), are effective for solving small-or moderate-sized problems. As the number of variables increases, however, the cost of each iteration can become prohibitive. The solution of the $n \times n$ system of Newton equations is expensive, on the order of n^3 arithmetic operations. Furthermore, computation of the n^2 second partial derivatives can also be expensive and is prone to human error. Thus, the benefits of fast local convergence are offset by the high costs of each iteration.

Some remedies for these concerns are possible. For example, it is possible to automate the derivative calculations Also, many large problems have sparse Hessian matrices, and special numerical linear algebra techniques for sparse matrices can reduce the storage and computational costs of using Newton's method.

Another alternative is algorithms that compromise on Newton's method by using first derivative only to compute an approximate Newton direction. The driving motivation in these algorithms is to reduce the expensive cost per iteration of Newton's method while retaining reasonably good convergence rates. While one can no longer expect a compromise on Newton's method to achieve a quadratic rate of convergence, it is still possible to achieve superlinear convergence, where

$$\lim_{k \to \infty} \frac{\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}_k - \boldsymbol{x}^*\|} = 0.$$

To achieve this, the search direction must approach the Newton search direction in the limit, as the solution is approached. Specifically, if $\nabla^2 f$ is Lipschitz continuous, and if the sequence $\{\boldsymbol{x}_k\}$ generated by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{p}_k.$$

converges to $\boldsymbol{x}^*$, where $\nabla^2 f(\boldsymbol{x}^*)$ is positive definite, then $\{\boldsymbol{x}_k\}$ converges to $\boldsymbol{x}^*$ superlinearly and $\nabla f(\boldsymbol{x}^*) = 0$ if and only if

$$\lim_{k \to \infty} \frac{\|\boldsymbol{p}_k - \boldsymbol{p}_k^N\|}{\|\boldsymbol{p}_k\|} = 0,$$

where $\boldsymbol{p}_k^N$ is the Newton direction at $\boldsymbol{x}_k$.

These results motivate a general Newton-type framework for unconstrained optimization algorithms, which attempt to find an approximate Newton direction. The search direction is obtained by solving the system

$$\boldsymbol{B}_k \boldsymbol{p}_k = -\nabla f(\boldsymbol{x}_k),$$

where $\boldsymbol{B}_k$ is a positive definite approximation to the Hessian.

The simplest of all such approximations sets $\boldsymbol{B}_k$ to be the identity matrix. The resulting search direction is $\boldsymbol{p}_k = -\nabla f(\boldsymbol{x}_k)$. A line search is needed, so that the resulting iterates are $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha \nabla f(\boldsymbol{x}_k)$. The method is known as the steepest descent method. (Technically it is not a Newton-type method since $\boldsymbol{B}_k$ is usually a poor approximation to the Hessian).

The method is simple, requires only one derivative calculation, does not require the computation of second derivatives, does not require that a system of linear equations be solved to compute the search direction, and does not require matrix storage. So in every way it reduces the costs of Newton's method—at least, the costs per iteration.

On the negative side, it has a slower rate of convergence than Newton's method, only converging at a linear rate, so that

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq c \|\boldsymbol{x}_k - \boldsymbol{x}^*\|.$$

The trouble is that the constant c can be very close to 1, so that the improvements from iteration to iteration can be imperceptible. In fact, even for the simple quadratic function $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x}$, using an

exact line search, it can be shown that the improvements in the objective value from one iteration to the next are of order

$$1 - \left[\frac{\kappa(\boldsymbol{Q}) - 1}{\kappa(\boldsymbol{Q}) + 1}\right]^2,$$

where $\kappa(\boldsymbol{Q})$ is the condition number of $\boldsymbol{Q}$. If $\kappa(\boldsymbol{Q})$ is large (close to 1, as is often the case in practice), convergence is so slow that $\boldsymbol{x}_{k+1} - \boldsymbol{x}_k$ is below the precision of computer arithmetic and the method fails. As a result, even though the costs per iteration are low, the overall costs of solving the optimization problem are high. The method is usually not a viable option for large difficult problems.

A more successful Newton-type family of methods are the quasi-Newton methods. Quasi-Newton methods are a class of methods that are motivated by Newton's method but avoid the expense of computing second derivatives. The search direction is obtained by solving the system

$$\boldsymbol{B}_k \boldsymbol{p} = -\nabla f(\boldsymbol{x}_k),$$

where $\boldsymbol{B}_k$ is an approximation to the Hessian $\nabla^2 f(\boldsymbol{x}_k)$. The matrix $\boldsymbol{B}_{k+1}$ is updated from $\boldsymbol{B}_k$ using gradient information from previous iterations.

In the one-dimensional case, quasi-Newton methods simply replace the second derivative by the slope of the secant line to the first derivative:

$$f''(\boldsymbol{x}_{k+1}) \approx \frac{f'(\boldsymbol{x}_{k+1}) - f'(\boldsymbol{x}_k)}{\boldsymbol{x}_{k+1} - \boldsymbol{x}_k},$$

so that

$$f''(\boldsymbol{x}_{k+1})(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) \approx f'(\boldsymbol{x}_{k+1}) - f'(\boldsymbol{x}_k).$$

To make $\boldsymbol{B}_{k+1}$ resemble the Hessian in the n-dimensional case requires that it satisfy the secant condition

$$\boldsymbol{B}_{k+1} \boldsymbol{s}_k = \boldsymbol{y}_k,$$

where

$$\boldsymbol{s}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k \text{ and } \boldsymbol{y}_k = \nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k).$$

The secant condition, however, does not uniquely define the matrix $\boldsymbol{B}_k$, so numerous methods have been proposed for updating $\boldsymbol{B}_{k+1}$ from $\boldsymbol{B}_k$. The most popular—and arguably most successful—of these methods has been the BFGS (Broyden, Fletcher, Goldfarb, Shanno) method, for which the update formula for the Hessian approximation is given by

$$\boldsymbol{B}_{k+1} = \boldsymbol{B}_k - \frac{(\boldsymbol{B}_k \boldsymbol{s}_k)(\boldsymbol{B}_k \boldsymbol{s}_k)^T}{\boldsymbol{s}_k^T \boldsymbol{B}_k \boldsymbol{s}_k} + \frac{\boldsymbol{y}_k \boldsymbol{y}_k^T}{\boldsymbol{y}_k^T \boldsymbol{s}_k}.$$

The matrix $\boldsymbol{B}_0$ is usually set to the identity matrix. Under appropriate conditions on the line search, it is possible to guarantee that if the BFGS method is applied to a bounded strictly convex function, then the BFGS method converges superlinearly to the unique global minimizer.

Quasi-Newton methods have been successful at solving a wide variety of practical problems, and are perhaps the most widely used methods for nonlinear optimization. However, their storage requirements and iteration costs can make them less suited for problems that have many variables. Limited memory quasi-Newton methods are a modification to quasi-Newton methods that require much less storage and much lower arithmetic costs per iteration. Rather than store the matrix $\boldsymbol{B}_k$, they store a few vectors that provide the information to store a matrix close to $\boldsymbol{B}_k$.

Another class of methods suitable for large problems are truncated-Newton methods. These methods are a compromise on Newton's method. They obtain the search direction by finding an approximate solution to the Newton equations, using some iterative method such as the conjugate-gradient method. The iterative method is stopped before the exact solution has been found, hence the name of the method. The methods do not require explicit computation of the Hessian, and only require the storage of a few vectors. They have been used successfully to solve problems with large number of variables.

Concluding Remarks

The theory and methods of unconstrained optimization are discussed in extensive detail in Dennis and Schnabel (1983), Gill, Murray and Wright (1981), Nash and Sofer (1996), Griva, Nash, and Sofer (2008), Nesterov (2004), and Nocedal and Wright (2006). For a guide to software for numerical

optimization, see Moré and Wright (1993) and the online NEOS Optimization Software Guide. This article focused on methods for computing local optima of differentiable functions; for a survey of methods for optimizing nondifferentiable functions, see Lemarechal (1989), and for a survey of derivative-free methods, see Kolda, Lewis and Torczon (2003). A survey of methods for global optimization is given in Rinnooy Kan and Timmer (1989) and in Horst, Pardalos and Thoai (2000).

See

- ▶ Global Optimization
- ▶ Linear Programming
- ▶ Mathematical Programming
- ▶ Nonlinear Programming

References

Dennis, J. E., & Schnabel, R. B. (1983). *Numerical methods for unconstrained optimization and nonlinear equations*. Englewood Cliffs, NJ: Prentice Hall.

Gill, P. E., Murray, W., & Wright, M. H. (1981). *Practical optimization*. New York: Academic Press.

Griva, I., Nash, S. G., & Sofer, A. (2008). *Linear and nonlinear optimization* (2nd ed.). Philadelphia: SIAM Books.

Horst, R., Pardalos, P. M. & Thoai, N. V. (2000). *Introduction to global optimization* (2nd ed.). Norwell: Kluwer Academic.

Kolda, T. G., Lewis, R. M., & Torczon, V. (2003). Optimization by direct search: New perspectives on some classical and modern methods. *SIAM Review, 45*(3), 385–482.

Lemarechal, C. (1989). Nondifferentiable optimization. In G.L. Nemhauser, A. H. G. Rinnooy Kan, & M. J. Todd (Eds.), *Handbooks in operations research and management science, vol. 1, Optimization* (Chap. VII, pp. 529–572). Amsterdam: Elsevier.

Moré, J. J., & Wright, S. J. (1993). *Optimization software guide*. Philadelphia: SIAM.

Nash, S. G., & Sofer, A. (1996). *Linear and nonlinear programming*. New York: McGraw-Hill.

Nesterov, Y. (2004). *Introductory lectures on convex optimization: A basic course*. Norwell, MA: Kluwer Academic.

Nocedal, J., & Wright, S. J. (2006). *Numerical optimization* (2nd ed.). New York: Springer.

Unconstrained Solution

A solution that is independent or free of constraints.

Uncontrollable Variables

In a decision problem, variables and other elements of a decision problem that are not under the control of the decision maker.

See

- ▶ Decision Maker (DM)
- ▶ Decision Problem
- ▶ Mathematical Model

Underachievement Variable

A nonnegative variable in a goal-programming problem constraint that measures how much the left-hand side of the constraint is less than the right-hand side.

See

- ▶ Goal Programming

Underdetermined System of Linear Equations

An $m \times n$ system of linear equations $\boldsymbol{Ax} = \boldsymbol{b}$ in which $m < n$. Such systems may have an infinite number of solutions or be inconsistent. The equation form of a linear-programming problem is under-determined.

Undirected Arc

In a network, an arc where flow can go in either direction.

Unimodular Matrix

An $m \times n$ matrix $\boldsymbol{A}$ such that any nonsingular square matrix formed by columns of $\boldsymbol{A}$ has a determinant value equal to +1 or −1. The matrix of the transportation problem is unimodular.

Unique Solution

The optimal solution to an optimization problem that has one and only one optimal solution.

See

► Multiple Optimal Solutions

Unrestricted Variable

A variable that can take on any value.

See

► Free Variable

Unsymmetric Primal-Dual Problems

The two linear-programming problems with the following:

Primal

$$\text{Minimize} \quad \boldsymbol{c}^T\boldsymbol{x}$$
$$\text{subject to} \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \quad \boldsymbol{x} \geq \boldsymbol{0}$$

Dual

$$\text{Maximize} \quad \boldsymbol{b}^T\boldsymbol{y}$$
$$\text{subject to} \quad \boldsymbol{A}^T\boldsymbol{y} \leq \boldsymbol{c}$$

Note that the variables of the dual problem are unrestricted.

See

► Strong Duality Theorem
► Symmetric Primal-Dual Problems

Upper-Bounded Problems

► Generalized Upper-Bounded (GUB) Problem
► Simple Upper-bounded Problem (SUB)

Urban Services

Kenneth Chelst
Wayne State University, Detroit, MI, USA

Introduction

Urban services cover a broad range of activities. These include sanitation and water systems; street cleaning to remove trash, snow, and ice; public housing; urban transportation systems; libraries (Reisman and Xu 1994); public health clinics; and other local government services. This article describes representative applications of operations research that improve the efficiency of these services. Emergency services such as police, fire, and emergency medical services are covered elsewhere in this volume.

Routing

One planning function common to many of the services listed above is the need to design efficient vehicle routes that minimize the cost of the service, primarily through minimization of travel time (Laporte 2009). Garbage collectors travel up and down streets, stopping in front of each house to pick up trash. Street sweepers move along curbsides sweeping up garbage while avoiding parked cars. Snowplows and salt trucks make their way through snow-covered arteries preparing them for smooth and safe traffic flow. This last example is the most complicated form of routing because highways and wide streets will require multiple passes over the same route.

This class of routing problems is much more complicated than designing routes to deliver mail (see Chinese Postman Problem). In an urban context, route planning can involve a large number of vehicles,

one- and two-way streets, and opportunities for U-turns. The commonality of problem structure has led to the development of GeoRoute, a software program with separate modules to address multiple decisions that involve traversing the streets of a region.

Snow Clearance

Langevin et al. (2006, 2007) wrote a four-part comprehensive review of operations research models and decision support systems designed for the planning and management of snow clearance and removal. They begin with an overview of issues associated with each class of decisions: spreading chemicals or sand, snowplowing, and snow removal. Planning for road clearance and snow removal involves a number of interrelated decisions. The building block for all decisions is the establishment of service levels for each road segment within the region. The service level defines the degree to which snow and ice will be removed from the road surface and over what time frame. Regression models and cost-benefit analysis have been used to establish service levels.

Strategic decisions include determining the size of the fleet of vehicles and the location of chemical storage depots and snow disposal sites. Related decisions involve the selection of de-icing agent and the process for disposing of snow. A linear mixed-integer programming model has been proposed as an integrated approach to decisions regarding depot location and sector design.

Operational decisions focus on developing efficient routes designed to meet service levels within specified time constraints. The planning of routes begins by dividing a region into sectors to be serviced by one or more vehicles. The sectors are generally designed to balance workloads and to be efficient in terms of travel to and from de-icing storage facilities and snow disposal sites. Once sectors have been defined, routes within each sector are specified.

Salt spreaders are typically on the road in the early stages of a snowstorm with the goal of keeping roads free of ice as much as possible before massive amounts of snow overwhelm the road network. Snowplows move in next to clear larger amounts of snow. This, too, may start before the storm has ended. The routing of chemical dispersers and snowplows is complicated by the need for real-time decisions. Routes need to be adjusted as weather and road conditions evolve during the course of the snowstorm. These routes may be further complicated by accidents blocking access to specific routes. The growing availability of routine real-time road conditions and GPS on vehicles makes the routing process more dynamic. Lastly, after large storms, snow blowers hit the roads accompanied by trucks that carry the snow to disposal sites. Planning snow removal routes involves two dimensions, with allocation of both snow blowers and snow removers. There needs to be a steady stream of trucks accompanying the snow blower.

A wide variety of OR models and software systems have been developed to address different decisions within this broad problem context. Some of the systems are descriptive simulations linked to mapping software that enable decision makers to evaluate the impact of their decisions on road clearance, capital investment, and operating costs. Others are constructive: they design routes, recommend storage locations, and establish work-shift schedules. Some attempt to place many of the decisions into a complex, large-scale optimization model. In almost every instance, these tools are embedded in user-friendly decision support systems that enable the decision maker to use his experience to make informed decisions that take into account local conditions.

Solid Waste Collection and Disposal

Solid waste management begins with citizens disposing of their garbage in bins to be emptied by trucks routed through streets and neighborhoods. If the city recycles, there will be multiple receptacles to be emptied. Both constrained p-median and set-covering models have been developed to optimize the locations of these bins (Devotta et al. 2008). The next task is to devise efficient routes (Beltrami and Bodin 1974). The garbage collection problem has an added complexity due to randomness in the volume of trash. As routes are planned, it is not possible to predict exactly how many stops a truck will make before it reaches capacity A related strategic decision is the size of the

collection trucks. This decision becomes more complex with disparate recyclables. In this case, the storage compartment must be divided to accommodate different types and amounts of recyclables. Once the garbage has been collected, it is transported either directly to a final disposal site, a transfer site, or an intermediate processing facility. Classical location models have been used to identify both transfer sites and end points (Eiselt 2007).

In New York City, operations research has had a broad impact on the sanitation department that goes beyond route planning. Workload forecasting models were developed to plan personnel needs, reduce overtime, shift vacations to off-peak seasons, and plan for the hiring of new personnel. Analyses were carried out to assess the impact of increasing the capacity of trucks. A simulation model was developed to understand the impact of even one illegally parked car on the effectiveness of a street sweeper. This information was used to help coordinate efforts with the traffic enforcement division. One innovation involved the creation of Project Scorecard, a program designed to sample 6,000 blocks each month to track how dirty the streets were and not just how much garbage was collected. The OR group also carried out a variety of studies to determine the impact of different regulations for separation of trash to facilitate recycling. In summary, operations research has fundamentally changed the way New York City makes decisions about street cleaning and the way the sanitation department manages its resources (Riccio et al. 1986).

Hazardous wastes management is much broader than just routing garbage collection trucks and location of facilities. There are a number of policy decisions that relate to what to recycle and how much responsibility to place on the individual. There is also the strategic decision of whether to send the waste to a landfill or an incineration facility. These issues are discussed in more detail in the article "Environmental Systems Analysis."

Public Housing

Operations research has much to offer urban services that do not involve routing or collection issues. In many urban environments, cities build and rent subsidized housing to the poor and the elderly. One of the first issues addressed by the Local Government Operational Unit of Reading, England, was ranking applicants for the 100,000 housing units owned by the City of Manchester. A housing points scheme was developed that captured the perspectives of housing department officials on relative need. Through the use of paired comparisons, they were able to answer questions such as whether an applicant with a medical problem should be given more points than one living in crowded conditions (Ricthie et al. 1996).

In the U.S., operations researchers have used queueing theory to evaluate two alternative tenant assignment policies, namely, first available unit versus priority assignment. They evaluated mean waiting times and the impact of assignment policies on racial integration. In a second study on the redevelopment of a housing project in East Boston, researchers used integer programming to plan the sequential relocation of housing tenants (Kaplan and Berman 1988).

In the 1990s, the U.S. began changing its preferred model for public housing. Instead of concentrating the poor in specialized housing units, the government initiated the Federal Housing Choice Voucher Program. Vouchers are used at the discretion of the recipients to rent apartments in the neighborhoods of their choice. These subsidies enable the poor to live closer to job opportunities and benefit from a higher communal standard of living, such as better schools and lower crime rates. The program also affords greater opportunity for racial integration on a modest scale. One disadvantage, however, is that recipients are separated from their core community and support system.

Operations researchers have developed models to address both policy and personal decisions. At the policy level, they have developed a multi-objective model to determine pareto optimal solutions for locating clusters of subsidized rental units in neighborhoods throughout a county or metropolitan region. The model explicitly accounts for multiple perspectives: those of renters of subsidized housing, nearby residents, and employers. An objective function assigns a weight to each perspective. The model includes several constraints that limit the number of units assigned to any one area.

With regard to personal decisions, OR models are applied to the decision of where to rent. A decision support system was developed for the Pittsburgh

Housing Authority to facilitate this process, designed to be used both by renters and housing authority counselors. The system applies the concepts of value-focused thinking to help the renter determine his multiple objectives. It is linked to a comprehensive GIS that includes numerous communal measures as well as listings of housing unit availability. This system helps the counselor and renter rank-order the renter's preferred housing choices and presents him with detailed neighborhood information. The system is also designed to help landlords who are interested in contributing units to the subsidized housing rental pool (Johnson 2005).

Urban Transportation Services

The issues surrounding the delivery of urban mass transit services have been studied from a broad range of disciplines. Economists have led the study of the relationship between fare structure (price) and demand. Urban and regional planners have researched the role of mass transit in urban and regional development. Statisticians have tackled the complex problem of estimating the origin–destination matrix that is critical in planning to meet route specific demand. Civil engineers have made both road and mass transit transportation planning a major component of their discipline and have often used operations research models in their studies or teamed with operations researchers. The journal *Transportation Science* is a focal point for reporting the latest research in this and related fields. In this review, the discussion is limited to the use of OR models.

Transportation services can be viewed from three perspectives, the passenger, the crew, and the infrastructure needed to provide the service (e.g., vehicles and facilities). The passenger is interested in traveling from point A to point B in the most cost- and time-efficient ways. The journey begins with travel from home or work to the bus stop or train station. Set-covering models have been used to increase accessibility to the nearest bus station (Murray 2003).

The design of transit routes and the scheduled frequency of trains or buses (e.g., headway) are the key management decisions that influence passenger experiences (Szeto and Wu 2011). Probabilistic models such as simulation in general and queueing models in particular have been developed to estimate passenger waiting times for both rail and bus services under a variety of operational strategies. In Australia, a software system called BUDI is used to address the issue of bus dispatching (Forbes et al. 1994).

It is common in bus transit for several buses to arrive at a particular stop within a relatively short period of time, followed by a relatively long wait until the next group of buses. Early research explored a range of static policies to address this phenomenon (Larson and Odoni 1981). The increased availability of real-time data on the location of each bus, however, has led to dynamic policies that recommend changes in speed so as to maintain a consistent headway between buses (Daganzo and Pilachowski 2011).

The elderly and handicapped have difficulty using mass transit to meet their travel needs. Taxis are an expensive alternative. Dial-a-Ride mini-bus systems fill the gap by picking up passengers upon request from multiple points and delivering them to different locations. These routes are constrained by time windows. Dynamic programming, clustering, and specially designed heuristic algorithms have been developed and used to efficiently manage the complex dispatching operation associated with Dial-a-Ride Cordeau and Laporte 2007).

Demand for transportation services varies significantly by time of day. Personnel and vehicle schedules must adjust accordingly to be cost-effective. Operations researchers have addressed this issue of manpower and rotation scheduling with mathematical programming models as well as HASTUS, a software package used to develop schedules for both personnel and vehicles (Blais et al. 1990). The Italian Railway Corporation has worked with the Italian Operational Research Society to sponsor university competitions for the design of effective heuristics. Random absences of personnel produce an added burden on managing an already complex system. Probabilistic models have been developed to help transportation managers pool resources to fill in unanticipated personnel shortages.

Garages and crew rosters are an important element of any municipal bus system (Ball et al. 1984; Caprara et al. 1998). Buses and drivers start and end their shifts at garages, and most maintenance occurs in these facilities. The decision as to the number and location of these garages has been analyzed by applying iteratively a minimum cost network flow model. A related question,

common to all capital-intensive systems, involves the maintenance of capital equipment. This issue falls within the broad range of operations research methods that model reliability, optimal maintenance, and replacement strategies. One statistical study of 2,000 buses in Montreal analyzed the relationship between inspection and breakdowns and suggested that the optimal inspection policy be changed from 5,000 kilometers to 8,000. Multi-criteria decision models have been used to develop component maintenance policies that focus not only on total maintenance cost but also transit vehicle availability and component reliability (Goplalaswamy et al. 1993).

Other Services

In an urban environment, one common problem that cuts across a broad range of both government and non-government services is how many facilities to build and where they should be located. Classic facility location models, both capacitated and uncapacitated, have been applied to address this decision in cities around the world. The urban setting often requires the organization of specialized delivery services that involve the scheduling and routing of multiple vehicles, usually with time constraints. One specific application area has been the delivery of meals (Johnson et al. 2002). Traveling salesman-based routing models have been used to develop and maintain efficient routes for a Meal-on-Wheels program that provides regular service to the homebound elderly. Another application is the delivery of home-care services to AIDS patients in Rome (De Angelis 1998). In general, OR models can help manage the delivery of a variety of services to the homes of an increasingly elderly urban population (Eveborn et al. 2009).

The role of operations researchers is not limited to model development; it also includes evaluation studies, discussion of performance evaluation, and concerns over equity. However, OR's overall impact on planning and managing urban services in cities worldwide has been extremely limited compared to its potential. The primary barriers to greater use are (a) an unfamiliarity among urban leaders regarding the potential of OR models to improve efficiency, (b) the limited availability of trained OR professionals in city government, and (c) a no-profit incentive or lack of accountability to drive the search for continuous improvement.

See

- ▶ Chinese Postman Problem
- ▶ Crime and Justice
- ▶ Emergency Services
- ▶ Environmental Systems Analysis
- ▶ Facility Location
- ▶ Libraries
- ▶ Location Analysis
- ▶ Manpower Planning
- ▶ Network
- ▶ Transportation Problem
- ▶ Traveling Salesman Problem
- ▶ Vehicle Routing

References

Ball, M., Assad, A., Bodin, L., Golden, B., & Spielberg, F. (1984). Garage location for an urban mass transit system. *Transportation Science, 18*, 56–75.

Beltrami, E. M., & Bodin, L. (1974). Networks and vehicle routing for municipal waste collection. *Networks, 4*, 65–94.

Blais, J. Y., Lamont, J., & Rousseau, J. M. (1990). The HAUSTUS vehicle and manpower scheduling system at the societe de transport de la communate urbaine de Montreal. *Interfaces, 20*(1), 26–42.

Caprara, A., Toth, P., Vigo, D., & Fischetti, M. (1998). Modeling and solving the crew rostering problem. *OR, 46*, 820–830.

Cordeau, J. F., & Laporte, G. (2007). The dial-a-ride problem: Models and algorithms. *Annals of Operations Research, 153*, 29–46.

Daganzo, C. F., & Pilachowski, J. (2011). Reducing bunching with bus-to-bus cooperation. *Transportation Research Part B, 45*, 267–277.

De Angelis, V. (1998). Planning home assistance for AIDS patients in the city of Rome, Italy. *Interfaces, 28*(3), 75–83.

Devotta, S., Vijay, R., Gautam, A., Kalamdhad, A., & Gupta, A. (2008). GIS-based locational analysis of collection bins in municipal solid waste management systems. *Journal of Environmental Engineering and Science, 7*, 39–43.

Eiselt, H. A. (2007). Locating landfills – optimization vs. Reality. *EJOR, 179*, 1040–1049.

Eveborn, P., Rönnqvist, M., Einarsdóttir, H., Eklund, M., Lidén, K., & Almroth, M. (2009). Operations research improves quality and efficiency in home care. *Interfaces, 39*(1), 18–34.

Forbes, M. A., Holt, J. N., Kilby, P. J., & Watts, A. M. (1994). BUDI: A software system for bus dispatching. *JORS, 45*, 497–508.

Goplalaswamy, V., Rice, J. A., & Miller, F. G. (1993). Transit vehicle component maintenance policy via multiple criteria decision making methods. *JORS, 44*, 37–50.

Johnson, M. P. (2005). Spatial decision support for assisted housing mobility counseling. *Decision Support Systems, 41*, 296–312.

Johnson, M. P., Gorr, W. L., & Roehrig, S. F. (2002). Location / allocation / routing for home-delivered meals provision. *International Journal of Industrial Engineering, 9*, 45–56.

Kaplan, E., & Berman, O. (1988). OR hits the heights: Relocation planning at the orient heights housing project. *Interfaces, 18*(6), 14–22.
Langevin, A., Perrier, N., & Campbell, J. F. (2006). A survey of models and algorithms for winter road maintenance. Part I: System design for spreading and plowing. *Computers and Operations Research, 33*, 209–238.
Langevin, A., Perrier, N., & Campbell, J. F. (2007). A survey of models and algorithms for winter road maintenance. Part IV: Vehicle routing and fleet sizing for plowing and snow disposal. *Computers and Operations Research, 34*, 258–294.
Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science, 43*, 408–416.
Larson, R. C., & Odoni, A. R. (1981). *Urban operations research*. Englewood Cliffs, NJ: Prentice Hall.
Murray, A. T. (2003). A coverage model for improving public transit system accessibility and expanding access. *Annals of Operations Research, 123*, 143–156.
Perrier, N., Langevin, A., & Campbell, J. F. (2007). A survey of models and algorithms for winter road maintenance. Part III: Vehicle routing and depot location for spreading. *Computers and Operations Research, 34*, 211–257.
Reisman, A., & Xu, X. (1994). Operations research in libraries: A review of 25 years of activity. *OR, 42*, 34–40.
Riccio, L. J., Miller, J., & Litke, A. (1986). Polishing the big apple, How management science has helped make New York streets cleaner. *Interfaces, 16*(1), 83–88.
Ricthie, C., Taket, A., & Bryant, J. (1996). *Community works – 26 case studies showing community operational research in action*. Sheffield Hallam University, UK: Pavic Publications.
Szeto, W. Y., & Wu, Y. (2011). A simultaneous bus route design and frequency setting problem for Tin shui Wai, Hong Kong. *EJOR, 209*, 141–155.

Utility Function

▶ Multiobjective Programming
▶ Utility Theory

Utility Theory

Peter Fishburn
AT&T Bell Laboratories, Murray Hill, NJ, USA

Introduction

Utility theory is the systematic study of preference structures and ways to represent preferences quantitatively. The objects on which preferences are defined could be potential outcomes of a decision, decision alternatives, individual or family consumption bundles in a fixed time period, time streams of net profits, investment portfolios, the entrees on a restaurant menu, or just about anything else. The preferences themselves are usually those of an individual, but are sometimes attributed to groups or organizations.

Let A denote the set of objects on which preferences are defined and let $\succcurlyeq$ be a binary relation on A, that is, a set of ordered pairs (x, y) of objects in A. When (x, y) is a member of $\succcurlyeq$, it is customary to write $x \succcurlyeq y$ and to say that x is at least as preferred as y. If $x \succcurlyeq y$ and not $(y \succcurlyeq x)$, then x is (strictly) preferred to y; if $x \succcurlyeq y$ and $y \succcurlyeq x$ then x and y are equally preferred, or are indifferent; if neither $x \succcurlyeq y$ nor $y \succcurlyeq x$ then x and y are preferentially incomparable. Strict preference and indifference are denoted by $x \succ y$ and $x \sim y$, respectively.

Utility theory typically regards the preference relation $\succcurlyeq$ on A as deterministic and interprets $x \succcurlyeq y$ as: if you have title to y you would be willing to trade it for title to x. There are also notions of uncertain or probabilistic preference that will not be described here. An excellent introduction to probabilistic preference and stochastic utility is provided by Luce and Suppes (1965).

Two book collections offer a broad overview of utility theory. Page (1968) contains historical essays, including an English translation of a 1738 paper by Daniel Bernoulli that introduced expected utility, a philosophical piece from 1823 by Jeremy Bentham that popularized the term utility, an excerpt from the game theory classic by John von Neumann and Oskar Morgenstern in 1944 that placed expected utility on a firm axiomatic foundation, and an economist's account by George Stigler of the development of utility theory from 1776 to 1915. The collection by Eatwell, Milgate and Newman (1990) covers many facets of utility theory, including several that are areas of contemporary research.

Distinguishing Features

There are numerous specific theories of utility. Each is distinguished by three features: (1) the structure of A; (2) the assumptions made about the properties of $\succcurlyeq$ on A; and (3) the quantitative representation that reflects $(A, \succcurlyeq)$ in a numerical structure.

Assumptions for feature 1 are structural assumptions, and those for feature 2 are preference

axioms. Together they are used to deduce the quantitative representation of feature 3. The representation's numerical functions are often called utility functions. Other real-valued functions, including probability distributions and threshold functions, also occur in representations.

An important adjunct of a utility representation is a description of the class of all functions that satisfy the representation. This is the representation's uniqueness structure. Some representations have very demanding uniqueness structures; others allow great latitude for their utility functions.

Two examples illustrate these ideas. First, let $A = \{\text{beef, chicken, fish, lamb}\}$ in regard to entrees for dinner. Assume for feature 2 that $\succ$ on A is a linear (strict) order, which, for all x, y and z in A, means that

$$\begin{aligned}&\succ \text{ is irreflexive: not } (x \succ x)\\ &\succ \text{ is complete: } x \neq y \rightarrow (x \succ y \text{ or } y \succ x)\\ &\succ \text{ is transitive: } (x \succ y \text{ and } y \succ z) \rightarrow x \succ z.\end{aligned}$$

One realization of $\succ$ on A is [beef $\succ$ lamb $\succ$ chicken $\succ$ fish]. This ordering is represented by a utility function u on A which assigns a number $u(x)$ to each x in A such that $[u(\text{beef}) > u(\text{lamb}) > u(\text{chicken}) > u(\text{fish})]$. There is great latitude for u. Every real-valued function on A whose $>$ ordering mirrors $>$ is a suitable utility function for the representation.

Second, let $A = [0, M]^3$, $0 < M$, the set of all triples (x_1, x_2, x_3) with $0 \leq x_i \leq M$ for each i. Interpret x_i as the income an individual earns in year i hence. One representation for feature 3 is the additive utility model

$$(x_1,x_2,x_3) \succcurlyeq (y_1,y_2,y_3) \rightarrow \sum_{i=1}^{3} u_i(x_i) \geq \sum_{i=1}^{3} u_i(y_i),$$

where each u_i is an increasing and continuous real-valued function on $[0, M]$. This requires that $\succcurlyeq$ on A be a weak order, which, for all x, y and z in A, means that

$$\begin{aligned}&\succcurlyeq \text{ is strongly connected : } x \succcurlyeq y \text{ or } y \succcurlyeq x\\ &\succcurlyeq \text{ is transitive : } (x \succcurlyeq y \text{ and } y \succcurlyeq z) \rightarrow x \succcurlyeq z.\end{aligned}$$

Another axiom that concerns additivity says that if two triples have identical incomes in a given year, then $\succcurlyeq$ between them remains unchanged if the identical income is changed, e.g.,

$$(x_1,x_2,x_3) \succcurlyeq (x_1,y_2,y_3) \rightarrow (y_1,x_2,x_3) \succcurlyeq (y_1,y_2,y_3).$$

Other axioms relate to monotonicity of utility in income and to continuity of each utility function.

The preceding model has a very tight uniqueness structure. In particular, when u_1, u_2 and u_3 satisfy the representation, then so do v_1, v_2 and v_3 in place of u_1, u_2 and u_3 respectively if and only if there are real numbers $\alpha > 0$ and β_1, β_2 and β_3 so that, for all m in $[0, M]$,

$$v_i(m) = \alpha u_i(m) + \beta_i, \quad i = 1, 2, 3.$$

Hence, except for an origin and unit, each u_i is unique.

The next few sections describe utility theories according to a three-part classification that mixes feature 1 with extra mathematical interpretations:

- certainty: there is no explicit use of chance or uncertainty;
- chance: chance in the form of numerical probabilities appears in A, but unquantified uncertainty is excluded;
- uncertainty: outcomes of decisions depend explicitly on uncertain events with not-yet-quantified probabilities.

Differences among classes can be illustrated by an object (m_1, m_2, m_3, m_4) in which each m_i is an amount of money. If the object describes a four-year income stream, the certainty designation applies. If the object is a gamble or risky prospect that pays off m_1, m_2, m_3 or m_4, each with probability 1/4, then chance applies. And if m_1 through m_4 are the amounts won for each dollar bet on your favorite horse in tomorrow's big race when the horse wins, places, shows and finishes out of the money, respectively, then uncertainty applies.

A differentiator for feature 2 is the extent to which preferences are transitive. The most restrictive case occurs when $\succcurlyeq$ is a weak order. Then each of $\succcurlyeq$, $\succ$ and $\sim$ is transitive. A more flexible case arises when $\succ$ but not $\sim$ is assumed transitive. Intransitive indifference is illustrated by a sequence of indifference comparisons $x_1 \sim x_2$, $x_2 \sim x_3, \ldots, x_{n-1} \sim x_n$ between similar objects, the first of which is definitely preferred to the last ($x_1 \succ x_n$). The most flexible case occurs when neither $\succ$ nor $\sim$ is assumed transitive. This allows

preference cycles, such as $x \succ y \succ z \succ x$. Fishburn (1991) provides access to the nontransitive preference literature.

Certainty

A basic theorem of utility theory for $(A, \succcurlyeq)$ says that a real number $u(x)$ can be assigned to each object in A so that, for all x and y in A,

$$x \succcurlyeq y \leftrightarrow u(x) \geq u(y),$$

if and only if $\succcurlyeq$ on A is a weak order and there is a countable (finite or denumerable) subset B of A such that, whenever $x \succ y$, some z in B satisfies $x \succcurlyeq z \succcurlyeq y$. A relaxation of this ordinal utility representation that accommodates intransitive indifference and thresholds for preference is

$$x \succ y \leftrightarrow u(x) > u(y) + \sigma(y),$$

where $\sigma(y) \geq 0$ for each y. This representation assigns a utility interval $[u(x), u(x) + \sigma(x)]$ to each x and has x preferred to y if and only if the right end of y's interval is less than the left end of x's interval. One of its preference axioms is

$$(x \succ a \text{ and } y \succ b) \rightarrow (x \succ b \text{ or } y \succ a).$$

A popular structure for preference theory formulates A as a subset of n-tuples $(x_1, \ldots, x_n)$, $(y_1, \ldots, y_n), \ldots$, in $\boldsymbol{X}_1 \times \boldsymbol{X}_2 \times \cdots \times \boldsymbol{X}_n$. Index i for $\boldsymbol{X}_i$ could refer to an attribute of objects in A or a time period. This product structure gives rise to special forms for the utility function u of the preceding paragraph, including the additive decomposition

$$u(x_{1,\ldots,}x_n) = \sum_{i=1}^{n} u_i(x_i)$$

in which u_i is a marginal utility function for the ith attribute or time period. A generalization that does not presume transitivity but retains additivity is

$$(x_1, \ldots, x_n) \succcurlyeq (y_1, \ldots, y_n) \leftrightarrow \sum_{i=1}^{n} \varphi_i(x_i, y_i) \geq 0,$$

where φ_i is defined on $\boldsymbol{X}_i \times \boldsymbol{X}_i$ and has $\varphi_i(x_i, x_i) = 0$.

Fishburn (1970, 1991), Keeney and Raiffa (1993) and Wakker (1989) have extensive coverage of the preceding topics.

Chance

The primary structure for chance takes A as a set of probability distributions on an outcome set X. For p in A, $p(x)$ is the probability that risky prospect p will yield outcome x. It is usually assumed as part of feature 1 that A is closed under convex combinations: if p and q are in A and $0 < \lambda < 1$, then $\lambda p + (1 - \lambda)q$ is also in A.

Two common preference axioms for $(A, \succcurlyeq)$ are weak order and the independence condition

$$p \succ q \rightarrow \lambda p + (1 - \lambda)r \succ \lambda q + (1 - \lambda)r$$

whenever p, q and r are in A and $0 < \lambda < 1$. When an Archimedean axiom is added to weak order and independence, the existence of a von Neumann-Morgenstern linear utility function u on A can be established. It has $p \succcurlyeq q \leftrightarrow u(p) \geq u(q)$ along with the linearity property

$$u(\lambda p + (1 - \lambda)q) = \lambda u(p) + (1 + \lambda)u(q),$$

and is unique except for origin and unit, i.e., unique up to transformations $\alpha u + \beta$ with $\alpha > 0$.

If A includes all distributions with finite support and $u(x)$ is defined as $u(p)$ when $p(x) = 1$, then linearity implies the expected-utility form

$$u(p) = \sum_{x} p(x)u(x)$$

for each finite-support distribution. Additional axioms are needed to obtain $u(p) = \int u(x)\, dp(x)$ for general probability measures.

Three variations on the expected-utility theme involve risk attitudes such as risk aversion when outcomes are monetary (Raiffa 1968; Wakker 1989), multiattribute expected utility when $\boldsymbol{X} = \boldsymbol{X}_1 \times \boldsymbol{X}_2 \times \cdots \times \boldsymbol{X}_n$, including additive and multiplicative decompositions of $u(x_1, \ldots, x_n)$ (Fishburn 1970; Keeney and Raiffa 1993; Wakker 1989), and generalizations of expected utility that relax one or more of its axioms (Fishburn 1988). A representation

that does not presume transitivity and substantially weakens the independence condition is $p \succcurlyeq q \leftrightarrow \varphi(p, q) \geq 0$, where φ is skew symmetric [$\varphi(p, q) + \varphi(q, p) = 0$] and linear separately in each argument.

Uncertainty

The main structure for uncertainty (Savage 1954) takes A as the set of functions $f, g, \ldots$, called acts from a set S of states into an outcome set X. If you choose f and state s occurs, your outcome is $f(s)$. It is presumed that one and only one state will occur, that you are uncertain which it will be, and that your chosen act will not affect its occurrence.

Savage's axioms (see also Fishburn 1970) for $(A, \succcurlyeq)$, which include weak order and independence assumptions, imply the existence of a bounded utility function u on X and a probability measure π on the set of all subsets of S such that, for all acts f and g,

$$f \succcurlyeq g \leftrightarrow \int_S u(f(s))d\pi(s) \geq \int_S u(g(s))d\pi(s).$$

Moreover, u is unique except for origin and unit, and π is unique.

Deduced probabilities in Savage's model are personal or subjective probabilities. The model itself is a subjective expected utility representation. The art of applying it to real-world problems is known as decision analysis (Raiffa 1968). Multiattribute and/ or time-stream outcomes occur in most applications.

Many other utility theories have been proposed for structures similar to Savage's. One strain relaxes his model by assuming monotonicity [$A \subseteq B \rightarrow \pi(A) \leq \pi(B)$] but not necessarily additivity [$\pi(A \cup B) = \pi(A) + \pi(B)$ when A and B are disjoint] for subjective probability. Another retains Savage's properties for π but relaxes transitivity to obtain

$$f \succcurlyeq g \leftrightarrow \int_S \varphi(f(s), g(s))d\pi(s) \geq 0,$$

with φ skew symmetric on $\boldsymbol{X} \times \boldsymbol{X}$. See Fishburn (1988) and Wakker (1989) for further details and references.

See

- Choice Theory
- Decision Analysis
- Game Theory
- Preference Theory

References

Eatwell, J., Milgate, M., & Newman, P. (Eds.). (1990). *Utility and probability*. London: Macmillan.

Fishburn, P. C. (1970). *Utility theory for decision making*. New York: Wiley.

Fishburn, P. C. (1988). *Nonlinear preference and utility theory*. Baltimore: The Johns Hopkins University Press.

Fishburn, P. C. (1991). Nontransitive preferences in decision theory. *Journal of Risk and Uncertainty, 4*, 113–134.

Keeney, R. L., & Raiffa, H. (1993). *Decisions with multiple objectives: Preferences and value trade-offs*. New York: Cambridge University Press.

Luce, R. D., & Suppes, P. (1965). Preference, utility and subjective probability. In R. D. Luce, R. R. Bush, & E. Galanter (Eds.), *Handbook of mathematical psychology, III* (pp. 249–410). New York: Wiley.

Page, A. N. (Ed.). (1968). *Utility theory: A book of readings*. New York: Wiley.

Raiffa, H. (1968). *Decision analysis: Introductory lectures on choice under uncertainty*. Reading, MA: Addison-Wesley.

Savage, L. J. (1954). *The foundations of statistics*. New York: Wiley.

Wakker, P. P. (1989). *Additive representations of preferences*. Dordrecht, The Netherlands: Kluwer.

V

Vacation Model

A queueing model where the server(s) will periodically stop serving a pool of customers for some period – take a "vacation" – before resuming service. (Note that during the so-called vacation, a server could possibly be serving some other source of customers elsewhere in the system). The vacation policy governs when a server stops service and when service is resumed. For example, one simple policy would be to take a vacation whenever the queue is empty and resume after a fixed period of time.

See

- ► Cyclic Service Discipline
- ► Queueing Theory
- ► Vacation Time

Vacation Time

In vacation models, the time starting from when a server stops serving customers (goes on "vacation") and ending when the server resumes serving customers.

See

- ► Cyclic Service Discipline
- ► Vacation Model

Validation

The process of determining how well a mathematical model of a real-world system conforms to reality for the purposes of the study being undertaken. Two key aspects of validity are face validity and predictive validity. Face validity is based on an examination of the assumptions and data going into the model for logical consistency and the review of the results by experts knowledgeable in the real world situation. Predictive validity is based on examining the model's predictions for events that were not used in building the model.

See

- ► Verification
- ► Verification, Validation, and Testing of Models

Value at Risk

Financial risk measure representing the maximum amount that can be lost over a given horizon with a specified probability, abbreviated as VaR. Mathematically, a quantile (or percentile) of the probability distribution of potential portfolio loss. For continuous probability distributions, VaR at the (1-α) 100% level is the value x such that $\Pr\{L > x\} = \alpha$,

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

where L represents the loss and α is generally 0.05 or 0.01, corresponding to 95% and 99% levels of VaR, respectively.

See

- Financial Engineering

Value Function

In a decision problem, let a be a feasible alternative from the set of all feasible alternatives A. Each alternative is measured against n attributes $(X_1,\ldots, X_n)$. The decision maker's (DM) problem is to choose an alternative $a \in A$ that maximizes the payoff vector of scores $[X_1(a), \ldots, X_n(a)] = \boldsymbol{X}(a)$. The value function is a real-valued, scalar function $v(\cdot)$ with the property that $v(\boldsymbol{X}(a)) > v(\boldsymbol{X}(b))$ if and only if the DM prefers alternative a to alternative b; and $v(\boldsymbol{X}(a)) = v(\boldsymbol{X}(b))$ if and only if the DM is indifferent between alternative a and alternative b. A similar concept can be found in dynamic programming and Markov decision processes.

See

- Approximate Dynamic Programming
- Choice Theory
- Decision Analysis
- Dynamic Programming
- Markov Decision Processes
- Multiple Criteria Decision Making
- Preference Theory
- Utility Theory

VAM

- Vogel's Approximation Method (VAM)

Variance Reduction Techniques in Monte Carlo Methods

Jack P. C. Kleijnen[1], Ad A. N. Ridder[2] and Reuven Y. Rubinstein[3]
[1]Tilburg University, Tilburg, The Netherlands
[2]Vrije University, Amsterdam, The Netherlands
[3]Technion – Israel Institute of Technology, Haifa, Israel

Introduction

Monte Carlo methods are simulation algorithms to estimate a numerical quantity in a statistical model of a real system. These algorithms are executed by computer programs. Variance reduction techniques (VRT) are needed, even though computer speed has been increasing dramatically, ever since the introduction of computers. This increased computer power has stimulated simulation analysts to develop ever more realistic models, so that the net result has not been faster execution of simulation experiments; e.g., some modern simulation models need hours or days for a single 'run' (one replication of one scenario or combination of simulation input values). Moreover there are some simulation models that represent rare events which have extremely small probabilities of occurrence), so even modern computer would take centuries to execute a single run—were it not that special VRT can reduce theses excessively long runtimes to practical magnitudes.

Preliminaries

In this contribution the focus is to estimate a quantity

$$\ell = E(H(\mathbf{Y})), \tag{1}$$

where $H(\mathbf{Y})$ is the performance function driven by an input vector $\mathbf{Y}$ with probability density function $f(\mathbf{y})$. To estimate ℓ through simulation, one generates a random sample $\mathbf{Y}_i$ with $i = 1,\ldots,N$ from $f(\mathbf{y})$, computes the sample function $H(\mathbf{Y}_i)$, and the sample-average estimator

$$\hat{\ell}_N = \frac{1}{N}\sum_{i=1}^{N} H(\mathbf{Y}_i).$$

This is called crude Monte Carlo sampling (CMC). The resulting sample-average estimator is an unbiased estimator for ℓ. Furthermore, as N gets large, laws of large numbers may be invoked (assuming simple conditions) to verify that the sample-average estimator stochastically converges to the actual quantity to be estimated. The efficiency of the estimator is captured by its relative error (RE), i.e., the standard error divided by the mean: $\mathrm{RE} = \sqrt{\mathrm{Var}(\hat{\ell}_\mathrm{N})}/\mathrm{E}(\hat{\ell}_\mathrm{N})$. Applying the Central Limit Theorem, one easily gets that $z_{1-\alpha/2}\mathrm{RE} < \varepsilon$, where $z_{1-\alpha/2}$ is the $(1-\alpha/2)^{th}$ quantile of the standard normal distribution (typically one takes $\alpha = 0.05$ so $z_{1-\alpha/2} = 1.96$) if and only if

$$P(|\frac{\hat{\ell}_N - \ell}{\ell}| < \varepsilon) > 1 - \alpha. \qquad (2)$$

When (2) holds, the estimator is said to be $(1-\alpha, \varepsilon)$ -efficient.

To illustrate, consider the one-dimensional version of (1):

$$\ell = \int h(y) f(y)\, dy.$$

Monte Carlo integration is a good way to estimate the value of the integral when the dimension is much higher than one, but the concept is still the same. Monte Carlo integration has become an important tool in financial engineering for pricing financial products such as options, futures, and swaps (Glasserman 2003). This Monte Carlo estimate samples $Y_1, \ldots, Y_N$ independently from f and calculates

$$\hat{\ell}_N = \frac{1}{N}\sum_{i=1}^{N} h(Y_i).$$

Then $\hat{\ell}_N$ is an unbiased estimator for ℓ, and the standard error is

$$\sqrt{\mathrm{Var}\left(\hat{\ell}_N\right)} = \sqrt{\frac{1}{N}\mathrm{Var}(\mathrm{h}(\mathrm{Y}))} = \sqrt{\frac{1}{N}E\left(h(Y)-\ell\right)^2}$$
$$= \sqrt{\frac{1}{N}\int \left(h(y)-\ell\right)^2 f(y)\, dy}.$$

Hence, the relative error (or efficiency) of the estimator is proportional to $1/\sqrt{N}$. This is a poor efficiency in case of high-dimensional problems where the generation of a single output vector is costly and consumes large computing time and memory. VRT improve efficiency if they indeed require smaller sample sizes. To be more specific, consider again the performance measure (1), and assume that besides the CMC-estimator $\hat{\ell}_N$, a VRT results in another unbiased estimator, denoted $\hat{\ell}^*_N$, also based on a sample of N independent and identical observations. The VRT-estimator is said to be statistically more efficient than the CMC-estimator if

$$\mathrm{Var}(\hat{\ell}^*_\mathrm{N}) < \mathrm{Var}(\hat{\ell}_\mathrm{N}).$$

Then one usually computes the reduction factor for the variance:

$$\frac{\mathrm{Var}(\hat{\ell}_\mathrm{N}) - \mathrm{Var}(\hat{\ell}^*_\mathrm{N})}{\mathrm{Var}(\hat{\ell}_\mathrm{N})} \times 100\%.$$

Notice that this factor does not depend on the sample size N. Suppose that the reduction factor is $100r\%$, so $r = 1 - (\mathrm{Var}(\hat{\ell}^*)/\mathrm{Var}(\hat{\ell}))$, and suppose that $(1-\alpha, \varepsilon)$-efficiency is desired. The required sample size for the CMC-estimator is N, given by $z_{1-\alpha/2}\mathrm{RE} = \varepsilon$, which holds iff

$$\frac{\ell^2\varepsilon^2}{z^2_{1-\alpha/2}} = \mathrm{Var}(\hat{\ell}_\mathrm{N}) = \frac{1}{\mathrm{N}}\mathrm{Var}(\hat{\ell}_1) \Leftrightarrow \mathrm{N}$$
$$= \frac{\mathrm{z}^2_{1-\alpha/2}}{\ell^2\varepsilon^2}\mathrm{Var}(\hat{\ell}_1).$$

The same reasoning holds for the VRT-estimator with a required sample size N^*. Consequently, the reduction in sample size becomes

$$\frac{N - N^*}{N} = \frac{\mathrm{Var}(\hat{\ell}_1) - \mathrm{Var}(\hat{\ell}^*_1)}{\mathrm{Var}(\hat{\ell}_1)} = r,$$

which is the same reduction as for the variance.

Generating samples under a VRT consumes generally more computer time (exceptions are antithetic and common random numbers; see next

section). Thus to make a fair comparison with CMC, the computing time should be incorporated when assessing efficiency improvement. Therefore, denote the required time to compute $\hat{\ell}_N$ by $\mathrm{TM}(\hat{\ell}_\mathrm{N})$. Then the effort of an estimator may be defined to be the product of its variance and its computing time: $\mathrm{EFFORT} = \mathrm{Var} \times \mathrm{TM}(\hat{\ell}_\mathrm{N})$. Notice that the effort does not depend on the sample size, if the computing time of N samples equals N times the computing time of a single sample. Then the estimator $\hat{\ell}^*_N$ is called more efficient than estimator $\hat{\ell}_N$ if the former requires less effort:

$$\mathrm{EFFORT}(\hat{\ell}^*_\mathrm{N}) < \mathrm{EFFORT}(\hat{\ell}_\mathrm{N}).$$

Again, a reduction factor for the effort can be defined, and one can analyze the reduction in computer time needed to obtain $(1-\alpha, \varepsilon)$-efficiency.

Estimating the Probability of Rare Events

An important class of statistical problems assesses probabilities of risky or undesirable events. These problems have become an important issue in many fields; examples are found in reliability systems (system failure), risk management (value-at-risk), financial engineering (credit default), insurance (ruin), and telecommunication (packet loss); see Juneja and Shahabuddin (2006); Rubino and Tuffin (2009). These problems can be denoted in the format of this contribution by assuming that a set A contains all the risky or undesirable input vectors $\mathbf{y}$, so that (1) becomes

$$\ell = P(A) = P(\mathbf{Y} \in A) = E(I_A(\mathbf{Y})),$$

where I_A is the indicator function of the set A (and thus in (1) $H = I_A$). The standard error of the Monte Carlo estimator is easily computed as $\sqrt{\ell(1-\ell)/N}$. Hence, the relative error becomes

$$\mathrm{RE} = \frac{\sqrt{\ell(1-\ell)}}{\ell\sqrt{\mathrm{N}}} = \frac{\sqrt{(1-\ell)}}{\sqrt{\ell \mathrm{N}}}. \qquad (3)$$

This equation implies that the sample size is inverse proportional to the target probability ℓ when requiring a prespecified efficiency; for instance, to obtain (95%, 10%)-efficiency, the sample size should be $N \geq 385(1-\ell)/\ell$. This leads immediately to the main issue of this contribution; namely $\ell << 1$ so A is called a rare event. To illustrate, suppose $\mathbf{Y} = (Y_1, \ldots, Y_n)$, where Y_j $(j = 1, \ldots, n)$ are identically and independently distributed (IID) with finite mean $\mu = E(Y_1)$ and standard deviation $\sigma = \sqrt{\mathrm{Var}(\mathrm{Y}_1)}$. Denote their sum by $S(\mathbf{Y}) = Y_1 + \cdots + Y_n$, and let the rare event be $A = \{S(\mathbf{Y}) > n(\mu + \delta)\}$ for a positive δ. A normal approximation results for $n = 500, \delta = 0.5, \sigma = 1$ that $\ell \approx 2.5\mathrm{E}\text{-}29$. A (95%, 10%)-efficient CMC-estimator would need sample size $N \approx 1.5\mathrm{E}{+}31$; which is impossible to realize. For example, the practical problem might require the daily simulation of a financial product for a period of two years in which a single normal variate needs to be generated per simulated day. Fast algorithms for normal variate generation on standard PCs require about 20 s for E+9 samples. This gives only E+5 vector samples $\mathbf{Y}$ per second. Note that the number of calls of the random number generator (RNG) is at least $N \times n$, which in this numerical example equals 7.5E+ 33; this number is large, but modern RNGs can meet this requirement (L'Ecuyer 2006).

In conclusion, the desired level of efficiency of the CMC estimator for rare event problems requires sample sizes that go far beyond available resources. Hence, researchers have looked for ways to reduce the variance of the estimator as much as possible for the same amount of sampling resources. Traditional VRTs are common random numbers, antithetic variates, control variates, conditioning, stratified sampling and importance sampling (Law 2007; Rubinstein and Kroese 2008). Modern VRTs include splitting techniques, and quasi-Mont Carlo sampling (Asmussen and Glynn 2007; Glasserman 2003).

Antithetic and Common Random Numbers

Consider again the problem of estimating $\ell = E(H(\mathbf{Y}))$ defined in (1). Now let $\mathbf{Y}_1$ and $\mathbf{Y}_2$ be two input samples generated from $f(\mathbf{y})$. Denote $X_i = H(\mathbf{Y}_i)$ with $i = 1, 2$. Then $\hat{\ell} = (X_1 + X_2)/2$ is an unbiased estimator of ℓ with variance

$$\mathrm{Var}(\hat{\ell}) = \frac{1}{4}\left(\mathrm{Var}(\mathrm{X}_1) + \mathrm{Var}(\mathrm{X}_2) + 2\mathrm{Cov}(\mathrm{X}_1, \mathrm{X}_2)\right).$$

If X_1, X_2 would be independent (as is the case in CMC), then $\mathrm{Var}(\hat{\ell})$ would be $\frac{1}{4}(\mathrm{Var}(\mathrm{X}_1) + \mathrm{Var}(\mathrm{X}_2))$. Obviously, variance reduction is obtained if

$\mathrm{Cov}(\mathrm{X}_1, \mathrm{X}_2) < 0$. The usual way to make this covariance negative is as follows. Whenever the uniform random number U is used for a particular purpose (for example, the second service time) in generating $\mathbf{Y}_1$, use the antithetic number $1 - U$ for the same purpose to generate $\mathbf{Y}_2$. Because U and $1 - U$ have correlation coefficient -1, it is to be expected that $\mathrm{Cov}(\mathrm{X}_1, \mathrm{X}_2) < 0$. This can be formalized by the following technical conditions.

(a) The sample vector $\mathbf{Y} = (Y_1, \ldots, Y_n)$ has components Y_j that are one-dimensional, independent random variables with distribution functions F_j that are generated by the inverse transformation method; i.e., $Y_j = F_j^{-1}(U_j)$, for $j = 1, \ldots, n$.

(b) The performance function H is monotone.

Under these conditions, negative correlation can be proved (Rubinstein and Kroese 2008). In condition (a) the inverse transformation requirement can be replaced by the assumption that all Y_j-components are Gaussian: when $Y \sim \mathrm{N}(\mu, \sigma^2)$, then $\tilde{Y} = 2\mu - Y \sim \mathrm{N}(\mu, \sigma^2)$, and clearly Y and $\tilde{Y}$ are negatively correlated. This alternative assumption is typically applied in financial engineering for option pricing (Glasserman 2003).

The method of common random numbers (CRN) is often applied in practice, because simulationists find it natural to compare alternative systems under 'the same circumstances'; for example, they compare different queueing disciplines (such as First-In-First-Out or FIFO, Last-In-First-Out or LIFO, Shortest-Jobs-First or SJF) using the same sampled arrival and service times in the simulation.

To be more specific, let $\mathbf{Y}$ be an input vector for two system performances $E(H_1(\mathbf{Y}))$ and $E(H_2(\mathbf{Y}))$, and the performance quantity of interest is their difference

$$\ell = E(H_1(\mathbf{Y})) - E(H_2(\mathbf{Y})).$$

To estimate ℓ, two choices produce an unbiased estimator:

1. Generate one sequence of IID input vectors $\mathbf{Y}_1, \ldots, \mathbf{Y}_N$, and estimate ℓ by

$$\hat{\ell}_N^{(1)} = \frac{1}{N} \sum_{i=1}^{N} (H_1(\mathbf{Y}_i) - H_2(\mathbf{Y}_i)).$$

2. Generate two independent IID sequences of input vectors $\mathbf{Y}_1^{(1)}, \ldots, \mathbf{Y}_N^{(1)}$, and $\mathbf{Y}_1^{(2)}, \ldots, \mathbf{Y}_N^{(2)}$, and estimate ℓ by

$$\hat{\ell}_N^{(2)} = \frac{1}{N} \sum_{i=1}^{N} H_1(\mathbf{Y}_i^{(1)}) - \frac{1}{N} \sum_{i=1}^{N} H_2(\mathbf{Y}_i^{(2)}).$$

The first method is the CRN method, and is intuitively prefered because it reduces variability:

$$\mathrm{Var}(\hat{\ell}_{\mathrm{N}}^{(1)}) < \mathrm{Var}(\hat{\ell}_{\mathrm{N}}^{(2)}).$$

To prove this inequality, denote $X_i = H_i(\mathbf{Y}_i)$. Then $\hat{\ell} = X_1 - X_2$ is an unbiased estimator of ℓ with variance

$$\mathrm{Var}(\hat{\ell}) = \mathrm{Var}(\mathrm{X}_1) + \mathrm{Var}(\mathrm{X}_2) - 2\mathrm{Cov}(\mathrm{X}_1, \mathrm{X}_2). \quad (4)$$

If X_1 and X_2 are independent (as is the case in the second method), then (4) becomes $\mathrm{Var}(\mathrm{X}_1) + \mathrm{Var}(\mathrm{X}_2)$. Hence, variance reduction is obtained if $\mathrm{Cov}(\mathrm{X}_1, \mathrm{X}_2) > 0$ in (4). This requirement is precisely the opposite of what was needed in antithetic variates. To force the covariance to become positive through CRN, the uniform random number U used for a particular purpose in generating $\mathbf{Y}_1$, is used for the same purpose to generate $\mathbf{Y}_2$. This can be formalized by the technical conditions completely analogous to those for antithetic variates.

CRN is often applied not only because it seems 'fair' but also because CRN is the default in many simulation software systems; e.g., Arena compares different scenarios using the same seed—unless, the programmer explicitly selects different seeds to initialize the various sampling processes (arrival process, service time at work station 1, etc.) for different scenarios. Detailed examples are given in Law (2007), pp. 582–594.

So while the simulation programmers need to invest little extra effort to implement CRN, the comparisons of various scenarios may be expected to be more accurate; i.e., the what-if or sensitivity analysis gives estimators with reduced variances. However, some applications may require estimates of the absolute (instead of the relative) responses; i.e., instead of sensitivity analysis the analysis aims at prediction or interpolation from the observed responses for the scenarios that have already been simulated. In these applications, CRN may give worse predictions; also see Chen, Ankenman, and Nelson (2010).

The analysis of simulation experiments with CRN should go beyond (4), which compares only two scenarios. The simplest extension is to compare a fixed set of (say) k scenarios using (4) combined with the Bonferroni inequality so that the type-I error rate does not exceed (say) α; i.e., in each comparison of two scenarios the value α is replaced by α/m where m denotes the number of comparisons (e.g., if all k scenarios are compared, then $m = k(k-1)/2$). Multiple comparison and ranking techniques are discussed in Chick and Gans (2009).

However, the number of interesting scenarios may be not fixed in advance; e.g., the scenarios differ in one or more quantitative inputs (e.g., arrival speed, number of servers) and the optimal input combination is wanted. In such situations, regression analysis is useful; i.e., the regression model is then a metamodel that enables validation, sensitivity analysis, and optimization of the simulation model; see Kleijnen (2008). The estimated regression coefficients (regression parameters) may have smaller variances if CRN is used—because of arguments based on (4)—except for the intercept (or the 'grand mean' in Analysis of Variance or ANOVA terminology). Consequently, CRN is not attractive in prediction, but it is in sensitivity analysis and optimization.

A better metamodel for prediction may be a Kriging or Gaussian Process model, assuming the scenarios correspond with combinations of quantitative inputs; e.g., the scenarios represent different traffic rates in a queuing simulation. Kriging implies that the correlation between the responses of different scenarios decreases with the distance between the corresponding input combinations; i.e., the Gaussian process is stationary (Kleijnen 2008). In random simulation (unlike deterministic simulation, which is popular in engineering) the Kriging metamodel also requires the estimation of the correlations between the 'intrinsic' noises of different scenarios caused by the use of random numbers U; see Chen et al. (2010).

An important issue in the implementation of Antithetics and CRN is synchronization, which is a controlling mechanism to ensure that the same random variables are generated by the same random numbers from the random number generator. As an example, consider comparing a single-server queue $GI/GI/1$ with a two-server system $GI/GI/2$. The two systems have statistically similar arrivals and service times, but the single server works twice as fast. The performance measure is the expected waiting time per customer (which is conjectured to be less in the two-server system). In a simulation study, the two simulation models with CRN should have the same arrival variates, and the same service-time variates. Suppose that $A_1, A_2, \ldots$ are the consecutive interarrival times in a simulation run of the $GI/GI/1$ model, and $S_1, S_2, \ldots$ are their associated service-time requirements. Then, in the corresponding simulation run of the $GI/GI/2$ model, these same values are used for the consecutive interarrival times, and their associated service times; see Kelton, Sadowski, and Sturrock (2007); Law (2007).

Antithetic and common random numbers can be combined. Their optimal combination is the goal of the Schruben-Margolin strategy; i.e., some blocks of scenarios use CRN, whereas other blocks use antithetic variates, etc.; see Chih (2013).

Control Variates

Suppose that $\hat{\ell}$ is an unbiased estimator of ℓ in the estimation problem (1); for example, C is the arrival time in a queueing simulation. A random variable C is called a control variate for $\hat{\ell}$ if it is correlated with $\hat{\ell}$ and its expectation γ is known. The linear control random variable $\hat{\ell}(\alpha)$ is defined as

$$\hat{\ell}(\alpha) = \hat{\ell} - \alpha(C - \gamma),$$

where α is a scalar parameter. It is easy to prove that the variance of $\hat{\ell}(\alpha)$ is minimized by

$$\alpha^* = -\frac{\mathrm{Cov}(\hat{\ell}, \mathrm{C})}{\mathrm{Var}(\mathrm{C})}.$$

The resulting minimal variance is

$$\mathrm{Var}(\hat{\ell}(\alpha^*)) = \left(1 - \rho^2_{\hat{\ell}C}\right)\mathrm{Var}(\hat{\ell}), \tag{5}$$

where $\rho_{\hat{\ell}C}$ denotes the correlation coefficient between $\hat{\ell}$ and C. Since $\mathrm{Cov}(\hat{\ell}, C)$ is unknown, the optimal control coefficient α^* must be estimated from the simulation. Estimating both $\mathrm{Cov}(\hat{\ell}, C)$ and $\mathrm{Var}(\mathrm{C})$ means that linear regression analysis is applied to estimate α^*. Estimation of α^* implies that the variance reduction becomes smaller than (2) suggests, and that the estimator may become biased.

The method can be easily extended to multiple control variables (Rubinstein and Marcus 1985).

A well-known application of control variates is pricing of Asian options. The payoff of an Asian call option is given by

$$H(\mathbf{Y}) = \max(0, \frac{1}{n}\sum_{j=1}^{n} Y_j - K),$$

where $Y_j = S_{jT/n}$, the expiration date T is discretized into n time units, K is the strike price, and S_t is the asset price at time t, which follows a geometric Brownian motion. Let r be the interest rate; then the price of the option becomes

$$\ell = E\left(e^{-rT}H(\mathbf{Y})\right).$$

A control variate may be $C = e^{-rT}\max(0, S_T - K)$ whose expectation is readily available from the Black-Scholes formula. Alternative control variates are S_T, or $\frac{1}{n}\sum_{j=1}^{n} S_{jT/n}$.

Conditioning

The method of conditional Monte-Carlo is based on the following basic probability formulas. Let X and Z be two arbitrary random variables, then

$$\begin{aligned} E(E(X|Z)) &= E(X) \quad \text{and} \\ \mathrm{Var(X)} &= \mathrm{E(Var(X|Z))} + \mathrm{Var(E(X|Z))}. \end{aligned} \tag{6}$$

Because the last two terms are both nonnegative, variance reduction is obvious:

$$\mathrm{Var(E(X|Z))} \le \mathrm{Var(X)}.$$

The same reasoning holds for the original problem (1), setting $X = H(\mathbf{Y})$. Also Z is allowed to be a vector variable. These formulas are used in a simulation experiment as follows. The vector $\mathbf{Z}$ is simulated, and the conditional expectation $C = E(H(\mathbf{Y})|\mathbf{Z})$ is computed. Repeating this N times gives the conditional Monte-Carlo estimator

$$\hat{\ell}_N^* = \frac{1}{N}\sum_{i=1}^{N} C_i.$$

A typical example is a level-crossing probability of a random number of variables:

$$\ell = P\left(\sum_{j=1}^{R} Y_j > b\right),$$

where $Y_1, Y_2, \ldots$ are IID positive random variables, R is a nonnegative integer-valued random variable, independent of the Y_j variables, and b is some specified constant. Such problems are of interest in insurance risk models for assessing aggregate claim distributions (Glasserman 2003). CMC can be improved by conditioning on the value of R for which level crossing occurs. To be more specific, denote the event of interest by A, so $\ell = E(I_A(\mathbf{Y}))$. Define

$$M = \min\left(r : \sum_{j=1}^{r} Y_j > b\right).$$

Assume that the distribution of Y can be easily sampled, and that the distribution of R is known and numerically available (for instance, Poisson). Then it is easy to generate a value of M. Suppose that $M = m$. Then $E(I_A(\mathbf{Y})|M = m) = P(R \ge m)$, which can be easily computed.

Stratified Sampling

Recall the original estimation problem $\ell = E(H(\mathbf{Y}))$, and its crude Monte Carlo estimator $\hat{\ell}_N$. Suppose now that there is some finite random variable Z taking values from $\{z_1, \ldots, z_m\}$, say, such that

1. the probabilities $p_i = P(Z = z_i)$ are known;
2. for each $i = 1, \ldots, m$, it is easy to sample from the conditional distribution of $\mathbf{Y}$ given $Z = z_i$.

Because

$$\ell = E(E(H(\mathbf{Y}))) = \sum_{i=1}^{m} p_i E(H(\mathbf{Y})|Z = z_i),$$

the stratified sampling estimator of ℓ may be

$$\hat{\ell}_N^* = \sum_{i=1}^{m} p_i \frac{1}{N_i}\sum_{j=1}^{N_i} H(\mathbf{Y}_{ij}),$$

where N_i IID samples $\mathbf{Y}_{i1}, \ldots, \mathbf{Y}_{iN_i}$ are generated from the conditional distribution of $\mathbf{Y}$ given $Z = z_i$, such that $N_1 + \cdots + N_m = N$. Notice that the estimator is unbiased. To assess its variance, denote the conditional variance of the performance estimator by $\sigma_i^2 = \mathrm{Var}(\mathrm{H}(\mathbf{Y})|\mathrm{Z} = \mathrm{z_i})$. The variance of the stratified sampling estimator is then given by

$$\mathrm{Var}(\hat{\ell}_{\mathrm{N}}^*) = \sum_{\mathrm{i}=1}^{\mathrm{m}} \frac{\mathrm{p_i^2}\sigma_{\mathrm{i}}^2}{\mathrm{N_i}}.$$

Because of (6)

$$\mathrm{Var}(\mathrm{H}(\mathbf{Y})) \geq \mathrm{Var}(\mathrm{H}(\mathbf{Y})|\mathrm{Z}) = \sum_{\mathrm{i}=1}^{\mathrm{m}} \mathrm{p_i}\sigma_{\mathrm{i}}^2.$$

Selecting proportional strata sample sizes $N_i = p_i N$ gives variance reduction:

$$\mathrm{Var}(\hat{\ell}_{\mathrm{N}}^*) = \sum_{\mathrm{i}=1}^{\mathrm{m}} \frac{\mathrm{p_i}\sigma_{\mathrm{i}}^2}{\mathrm{N}} \leq \frac{1}{\mathrm{N}}\mathrm{Var}(\mathrm{H}(\mathbf{Y})) = \mathrm{Var}(\hat{\ell}_{\mathrm{N}}).$$

It can be shown that the strata sample sizes N_i that minimize this variance are

$$N_i = N\frac{p_i\sigma_i}{\sum_{j=1}^m p_j\sigma_j};$$

see Rubinstein and Kroese (2008). A practical problem is that the standard deviations σ_i are usually unknown, so these variances are estimated by pilot runs. Stratified sampling is used in financial engineering to get variance reductions in problems such as value-at-risk, and pricing path-dependent options (Glasserman 2003).

Importance Sampling

The idea of importance sampling is explained best in case of estimating the probability of an event A. The underlying sample space is $(\Omega, \mathscr{F})$ for which $A \in \mathscr{F}$, and the probability measure P on this space is given by the specific simulation model. In a simulation experiment for estimating $P(A)$, the CMC estimator would be $\hat{\ell}_N = \sum_{i=1}^N I_A^{(i)}$, where $I_A^{(1)}, \ldots, I_A^{(N)}$ are IID indicator functions of event A generated under P. On average in only one out of $1/P(A)$ generated samples the event A occurs, and thus for rare events (where $P(A)$ is extremely small) this procedure fails. Suppose that there is an alternative probability measure P^* on the same $(\Omega, \mathscr{F})$ such that (i) A occurs much more often, and (ii) P is absolutely continuous with respect to P^*, meaning

$$\forall F \in \mathscr{F} \ : \ P(F) > 0 \ \Rightarrow \ P^*(F) > 0.$$

Then according to the Radon-Nikodym theorem, it holds that there is a measurable function L on Ω such that $\int_F dP = \int_F L\, dP^*$ for all $F \in \mathscr{F}$. The function L is called likelihood ratio and usually written as $L = dP/dP^*$; the alternative probability measure P^* is said to be the importance sampling probability measure, or the change of measure. Thus, by weighting the occurrence I_A of event A with the associated likelihood ratio, simulation under the change of measure yields an unbiased importance sampling estimator

$$\hat{\ell}_N^* = \sum_{i=1}^N L^{(i)} I_A^{(i)}.$$

More importantly, variance reduction is obtained when the change of measure has been chosen properly, as will be explained below. Importance sampling has been applied successfully in a variety of simulation areas, such as stochastic operations research, statistics, Bayesian statistics, econometrics, finance, systems biology; see Rubino and Tuffin (2009). This section will show that the main issue in importance sampling simulation is the question which change of measure to consider. The choice is very much problem dependent, however, and unfortunately, it is difficult to prevent gross misspecification of the change of measure P^*, particularly in multiple dimensions.

Exponential Change of Measure

As an illustration, consider the problem of estimating the level-crossing probability

$$\ell_n = P(A_n) \quad \text{with} \quad \mathrm{A_n} = \{\mathrm{Y_1} + \cdots + \mathrm{Y_n} > \mathrm{na}\}, \tag{7}$$

where $Y_1, \ldots, Y_n$ are IID random variables with finite mean $\mu = E(Y) < a$ and with a light-tailed PDF $f(y, \mathbf{v})$, in which $\mathbf{v}$ denotes a parameter vector, such as mean

and variance of a normal density. It is well-known from Cramér's Theorem that $P(A_n) \to 0$ exponentially fast as $n \to \infty$. Suppose that under the importance sampling probability measure the random variables $Y_1, \ldots, Y_n$ remain IID, but with an exponentially tilted PDF (also called exponentially twisted), with tilting factor t:

$$f_t(y, \mathbf{v}) = \frac{f(y, \mathbf{v}) e^{ty}}{\int f(y, \mathbf{v}) e^{ty}\, dy}.$$

Thus, in the importance sampling simulations the Y_k -samples are generated from $f_t(y, \mathbf{v})$. Because of the IID assumption, the likelihood ratio becomes

$$\begin{aligned} L(Y_1, \ldots, Y_n) &= \prod_{k=1}^{n} \frac{f(Y_k, \mathbf{v})}{f_t(Y_k, \mathbf{v})} \\ &= \exp\left(n\psi(t) - t \sum_{k=1}^{n} Y_k \right), \end{aligned} \tag{8}$$

with $\psi(t) = \log \int f(y, \mathbf{v}) e^{ty}\, dy$. Variance reduction is obtained if

$$\begin{aligned} &\mathrm{Var}_t(\hat{\ell}_{\mathrm{N}}^*) \le \mathrm{Var}(\hat{\ell}_{\mathrm{N}}) \quad \Leftrightarrow \quad \mathrm{Var}_t(\hat{\ell}_1^*) \le \mathrm{Var}(\hat{\ell}_1) \\ &\Leftrightarrow \quad E_t[(\hat{\ell}_1^*)^2] \le E[(\hat{\ell}_1)^2] \\ &\Leftrightarrow \quad E_t[(I_A L(Y_1, \ldots, Y_n))^2] \le E[(I_A)^2]. \end{aligned}$$

Because of (8), it is easy to show that the variance is minimized for $t = (\psi')^{-1}(a)$. In that case the importance sampling estimator is logarithmically efficient (also called asymptotically optimal; see Rubino and Tuffin (2009; Chapter 4)):

$$\lim_{n \to \infty} \frac{\log E_t\left[(\hat{\ell}_N^*)^2 \right]}{\log E_t\left[\hat{\ell}_N^* \right]} = 2,$$

where the subscript t means that the underlying probability is the change of measure. Asymptotic optimality implies that $\mathrm{RE}(\hat{\ell}_{\mathrm{N}}^*)$ grows subexponentially as $n \to \infty$, whereas for CMC the relative error grows exponentially (see (3)).

The Cross-entropy Method

A general heuristic for constructing an importance sampling algorithm is to consider only a parameterized family of changes of measures. Consider again problem (1), with PDF $f = f(\mathbf{y}, \mathbf{v})$ where $\mathbf{v}$ is the parameter vector. Thus, let Θ be all feasible parameter vectors for f. For any $\theta \in \Theta$, the change of measure P_θ induces the (single-run) importance sampling estimator

$$\hat{\ell}_\theta^* = H(\mathbf{Y}) \frac{dP}{dP_\theta}(\mathbf{Y}) = H(\mathbf{Y}) \frac{f(\mathbf{Y}, \mathbf{v})}{f(\mathbf{Y}, \theta)}.$$

The optimal change of measure is found by variance minimization. Since the estimators are unbiased, it suffices to minimize the second moment:

$$\min_{\theta \in \Theta} E_\theta \left[\left(H(\mathbf{Y}) \frac{f(\mathbf{Y}, \mathbf{v})}{f(\mathbf{Y}, \theta)} \right)^2 \right].$$

Generally, this problem is hard. A successful approach is based on cross-entropy minimization as explained in Rubinstein and Kroese (2004). First, consider the optimal change of measure, resulting in a zero-variance estimator:

$$dP^{\mathrm{opt}}(\mathbf{Y}) = \frac{H(\mathbf{Y}) dP(\mathbf{Y})}{\ell}. \tag{9}$$

This change of measure is not implementable as it requires knowledge of the unknown quantity ℓ. The cross-entropy method finds P_θ by minimizing the Kullback–Leibler distance (or cross-entropy) within the class of feasible changes of measure:

$$\min_{\theta \in \Theta} \mathscr{D}(dP^{\mathrm{opt}}, dP_\theta),$$

where the cross-entropy is defined by

$$\begin{aligned} \mathscr{D}(dP^{\mathrm{opt}}, dP_\theta) &= E^{\mathrm{opt}}\left[\log\left(\frac{dP^{\mathrm{opt}}}{dP_\theta}(\mathbf{Y}) \right) \right] \\ &= E_v\left[\frac{dP^{\mathrm{opt}}}{dP}(\mathbf{Y}) \log\left(\frac{dP^{\mathrm{opt}}}{dP_\theta}(\mathbf{Y}) \right) \right]. \end{aligned}$$

Substituting expression (9), and canceling constant terms and factors, the equivalent cross-entropy problem becomes

$$\max_{\theta \in \Theta} E_v[H(\mathbf{Y}) \log dP_\theta(\mathbf{Y})].$$

There are several ways to solve this stochastic optimization problem. The original description of the cross-entropy method for such problems proposes to solve the stochastic counterpart iteratively, see Rubinstein and Kroese (2004). This approach has been applied successfully to a variety of estimation and rare-event problems.

State-dependent Importance Sampling

The importance sampling algorithms described above were based on a static change of measure; i.e., the samples are generated by a fixed alternative statistical law; see (8). In specific problems, such as (7), the static importance sampling algorithm yields an efficient estimator. However, for many problems it is known that efficient estimators require an adaptive or state-dependent importance sampling algorithm (Juneja and Shahabudding 2006). To illustrate this concept, consider again the problem of estimating the level-crossing probability (7). The Y_k-variables are called jumps of a random walk $(S_k)_{k=0}^n$, defined by $S_0 = 0$, and for $k \geq 1$: $S_k = \sum_{j=1}^k Y_j = S_{k-1} + Y_k$. Under a state-dependent change of measure, the next jump Y_{k+1} might be generated from a PDF $f(y|k+1, S_k)$; i.e., it depends on jump time $k+1$ and current state S_k. Hence, under the change of measure, the process $(S_k)_{k=0}^n$ becomes an inhomogeneous Markov chain. Given a generated sequence $Y_1, \ldots, Y_n$, the associated likelihood ratio is

$$L(Y_1, \ldots, Y_n) = \prod_{k=1}^{n} \frac{f(Y_k, \mathbf{v})}{f(Y_k|k, S_{k-1})}.$$

The next question is: Which time-state dependent PDFs should be chosen for this kind of change of measure? The criterion could be (i) variance minimization, (ii) cross-entropy minimization, or (iii) efficiency.

1. A small set of rare-event problems are suited to find so-called zero-variance approximate importance sampling algorithms, notably level-crossing problems with Gaussian jumps, reliability problems, and certain Markov chains problems; see L'Ecuyer et al. (2010).
2. A cross-entropy minimization is applied after each state S_k for determining the PDF of the next jump (Ridder and Taimre 2011). The result is that when the level-crossing at time n can be reached from state S_k just by following the natural drift, no change of measure is applied. Otherwise, the next jump is drawn from an exponentially tilted PDF with tilting factor $t = (\psi')^{-1}((an - S_k)/(n-k))$. This would be the static solution given before when starting at time $k = 0$. This approach gives logarithmic efficiency.
3. The method developed by Dupuis and Wang (2007) considers the rare-event problem as an optimal control problem in a differential game. Applying dynamic programming techniques while using large-deviations expressions, the authors develop logarithmically efficient importance sampling algorithms. This approach works also for rare events in Jackson networks (Dupuis et al. 2007).

Markov Chains

Many practical estimation problems in statistical systems (e.g., reliability, production, inventory, queueing, communications) can be reformulated as a Markov model to estimate a quantity $\ell = P(\mathbf{Y}_T \in \mathscr{F})$. Let $\{\mathbf{Y}_t : t = 0, 1, \ldots\}$ denote a discrete-time Markov chain with a state space $\mathscr{X}$ with transition probabilities $p(\mathbf{x}, \mathbf{y})$; $\mathscr{F} \subset \mathscr{X}$ is a subset of states, and T is a stopping time. A typical example is a system of highly reliable components where the response of interest is the probability of a break down of the system.

Assume that the importance sampling is restricted to alternative probability measures P^* such that the Markov chain property is preserved with transition probabilities $p^*(\mathbf{x}, \mathbf{y})$ satisfying

$$p(\mathbf{x}, \mathbf{y}) > 0 \;\Leftrightarrow\; p^*(\mathbf{x}, \mathbf{y}) > 0.$$

This constraint ensures the absolute continuity condition. Furthermore, assuming that the initial distribution remains unchanged, the likelihood ratio of a simulated path of the chain becomes simply

$$L = \prod_{t=0}^{T-1} \frac{p(\mathbf{Y}_t, \mathbf{Y}_{t+1})}{p^*(\mathbf{Y}_t, \mathbf{Y}_{t+1})}.$$

Thus, it suffices to find the importance-sampling transition-probabilities $p^*(\mathbf{x}, \mathbf{y})$. Considering these probabilities as parameters, the method of cross-entropy is most convenient; Ridder (2010) gives sufficient conditions to guarantee asymptotic

optimality. However, many realistic systems are modeled by Markov chains with millions of transitions, which causes several difficulties: the dimensionality of the parameter space, the danger of degeneracy of the estimation, and numerical underflow in the computations. Several approaches are proposed to reduce the parameter space in the cross-entropy method (de Boer and Nicola 2002; Kaynar and Ridder 2010).

Another approach to importance sampling in Markov chains approximates the zero-variance probability measure P^{opt}. It is known that this P^{opt} implies transition probabilities of the form

$$p^{\text{opt}}(\mathbf{x},\mathbf{y}) = p(\mathbf{x},\mathbf{y})\,\frac{\gamma(\mathbf{y})}{\gamma(\mathbf{x})},$$

where $\gamma(\mathbf{x}) = P(\mathbf{Y}_T \in \mathscr{F} | \mathbf{Y}_0 = \mathbf{x})$. As these quantities are unknown (and in fact the subject of interest), these zero-variance transition probabilities cannot be implemented. However, approximations of the $\gamma(\mathbf{x})$ probabilities may be considered (L'Ecuyer et al. 2010). Under certain conditions this approach leads to strong efficiency of the importance sampling estimator.

Splitting

The splitting method may handle rare-event probability estimation. Unlike importance sampling, the probability laws remain unchanged, but a drift to the rare event is constructed by splitting (cloning) favorable trajectories, and terminating unfavorable trajectories. This idea may be explained as follows. Consider a discrete-time Markov chain $\{Y_t : t = 0, 1, \ldots\}$ on a state space $\mathscr{X}$. Suppose that the chain has a regeneration state or set $\mathbf{0}$, a set of failure states $\mathscr{F}$, and a starting state $\mathbf{y}_0$. The response of interest is the probability that the chain hits $\mathscr{F}$ before $\mathbf{0}$. More formally, if T denotes the stopping time

$$T = \inf\{t : \mathbf{Y}_t \in \mathbf{0} \cup \mathscr{F}\},$$

then

$$\ell = P(\mathbf{Y}_T \in \mathscr{F}).$$

The initial state $\mathbf{y}_0 \notin \mathbf{0} \cup \mathscr{F}$ may have either some initial distribution, or be fixed and known. The assumption is that ℓ is so small that CMC in impractical. Suppose that the state space is partitioned into sets according to

$$\mathscr{X} \supset \mathscr{X}_1 \supset \mathscr{X}_2 \supset \cdots \supset \mathscr{X}_m = \mathscr{F}, \tag{10}$$

with $\mathbf{0} \in \mathscr{X}/\mathscr{X}_1$. Usually these sets are defined through an importance function $\phi : \mathscr{X} \to \mathbb{R}$, such that for each k, $\mathscr{X}_k = \{\mathbf{y} : \phi(\mathbf{y}) \geq L_k\}$ for certain levels $L_1 \leq L_2 \leq \cdots \leq L_m$, with $\phi(\mathbf{0}) = L_0 < L_1$. Now define stopping times T_k and associated events A_k by

$$T_k = \inf\{t : X(t) \in \mathbf{0} \cup \mathscr{X}_k\}; \quad A_k = \{\mathbf{Y}_{T_k} \in \mathscr{X}_k\}.$$

Because of (10), clearly $A_1 \supset A_2 \supset \cdots \supset A_m = A = \{\mathbf{Y}_T \in \mathscr{F}\}$. Thus the rare-event probability $\ell = P(A)$ can be decomposed as a telescoping product:

$$\ell = P(A_1) \prod_{k=2}^{m} P(A_k | A_{k-1}).$$

To estimate ℓ, one might estimate all conditional probabilities $P(A_k|A_{k-1})$ separately (say) by $\hat{\ell}_k$, which gives the product estimator

$$\hat{\ell}^* = \prod_{k=1}^{m} \hat{\ell}_k, \tag{11}$$

where $\hat{\ell}_1$ estimates $P(A_1)$. The splitting method implements the following algorithm for constructing the $\hat{\ell}_k$ estimators in a way that the product estimator is unbiased. In the initial stage ($k = 0$), run N_0 independent trajectories of the chain starting at the initial state $\mathbf{y}_0$. Each trajectory is run until either it enters $\mathscr{X}_1$ or it returns to $\mathbf{0}$, whatever come first. Let R_1 be the number of "successful" trajectories; i.e., trajectories that reach $\mathscr{X}_1$ before $\mathbf{0}$. Then set $\hat{\ell}_1 = R_1/N_0$. Consider stage $k \geq 1$, and suppose that R_k trajectories have entered set $\mathscr{X}_k$ in entrance states $\mathbf{Y}_1^{(k)}, \ldots, \mathbf{Y}_{R_k}^{(k)}$ (not necessarily distinct). Replicate (clone) these states, until a sample of size N_k has been obtained. From each of these states, run a trajectory of the chain, independently of the others. Each trajectory is run until either it enters $\mathscr{X}_{k+1}$ or it returns to $\mathbf{0}$, whatever come first. Let R_{k+1} be the number of successful trajectories, i.e., trajectories that reach $\mathscr{X}_{k+1}$ before $\mathbf{0}$. Then set $\hat{\ell}_{k+1} = R_{k+1}/N_k$. This procedure is continued until all trajectories have entered either $\mathscr{F}$ or returned to $\mathbf{0}$.

This form of the splitting method has attracted a lot of interest (see the reference list in Rubino and Tuffin (2009; Chapter 3)), both from a theoretical point of view analyzing the efficiency, and from a practical point of view describing several applications. The analysis shows that the product estimator (11) is unbiased. Furthermore, the analysis of the efficiency of the splitting technique depends on the implementation of (a) selecting the levels, (b) the splitting (cloning) of successful trajectories, and (c) the termination of unsuccessful trajectories. Generally, the problem of solving these issues optimally is like choosing an optimal change of measure in importance sampling. In fact, Dean and Dupuis (2008) discusses this relationship when the model satisfies a large deviations principle.

Concerning issue (c), the standard splitting technique terminates a trajectory that returns to the regeneration state $\mathbf{0}$, or—in case of an importance function—when the trajectory falls back to level L_0. This approach, however, may be inefficient for trajectories that start already at a high level L_k. Therefore, there are several adaptations such as truncation (L'Ecuyer et al. 2007), RESTART (Villen-Altamirano, and Villen-Altamirano 1994), and Russian roulette principle (Melas 1997).

Concerning issue (b), there are numerous ways to clone a trajectory that has entered the next level, but the two ways implemented mostly are (i) fixed effort, and (ii) fixed splitting. Fixed effort means that the sample sizes N_k are predetermined, and thus each of the R_k entrance states at set $\mathscr{X}_k$ is cloned $c_k = \lfloor N_k/R_k \rfloor$ times. The remaining $N_k \mathrm{mod} R_k$ clones are selected randomly. An alternative is to draw N_k times at random (with replacement) from the R_k available entrance states. Fixed splitting means that the splitting factors c_k are predetermined, and each of the R_k entrance states at set $\mathscr{X}_k$ is cloned c_k times to give sample size $N_k = c_k R_k$.

For a certain class of models, Glasserman et al. (1999) has shown that fixed splitting gives asymptotic optimality (as $\ell \to 0$) when the number of levels $m \approx -\ln(\ell)/2$, with sets $\mathscr{X}_k$ such that $P(A_k|A_{k-1})$ are all equal (namely, roughly equal to e^{-2}) and splitting factors such that $c_k P(A_{k+1}|A_k) = 1$. However, since ℓ and the $P(A_{k+1}|A_k)$ are unknown in practice, this result can only be approximated. Moreover, one should take into account the amount of work or computing time in the analysis; for example, Lagnoux (2006) determines the optimal setting under a budget constraint of the expected total computing time.

Application to Counting

Recently, counting problems have attracted the interest of the theoretical computer science and the operations research communities. A standard counting problem is model counting, or #SAT: how many assignments to boolean variables satisfy a given boolean formula consisting of a conjunction of clauses? The related classical decision problem is: does there exist a true assignment of the formula? Because exact counting is impracticable due to the exponential increase in memory and running times, attention shifted to approximate counting—notably by applying randomized algorithms. In this randomized setting, the counting problem is equivalent to rare event simulation: let $\mathscr{X}^*$ be the set of all solutions of the problem, whose number $|\mathscr{X}^*|$ is unknown and the subject of study. Assume that there is a larger set of points $\mathscr{X} \supset \mathscr{X}^*$ with two properties:

1. the number of points $|\mathscr{X}|$ is known;
2. it is easy to generate uniformly points $\mathbf{x} \in \mathscr{X}$.

Because

$$|\mathscr{X}^*| = \frac{|\mathscr{X}^*|}{|\mathscr{X}|}|\mathscr{X}|,$$

it suffices to estimate

$$\ell = \frac{|\mathscr{X}^*|}{|\mathscr{X}|} = P(\mathbf{U} \in \mathscr{X}^*),$$

where $\mathbf{U}$ is the uniform random vector on $\mathscr{X}$. Typically ℓ is extremely small, and thus rare event techniques are required. Splitting techniques with Markov chain Monte Carlo (MCMC) simulations have been developed in Botev and Kroese (2008) and Rubinstein (2010) to handle such counting problems.

Quasi-Monte Carlo

Suppose that the performance function H in (1) is defined on the d-dimensional unit hypercube $[0,1)^d$,

and the problem is to compute its expectation with respect to the uniform distribution:

$$\ell = E(H(\mathbf{U})) = \int_{[0,1)^d} H(\mathrm{u})\,\mathrm{du}.$$

As was shown in the introduction, the variance of the CMC estimator $\hat{\ell}_{Nm}$ using a sample size $N \times m$ equals $\sigma^2/(N \times m)$, where

$$\sigma^2 = \int_{[0,1)^d} H^2(\mathrm{u})\,\mathrm{du} - \ell^2.$$

Let $P_N = \{\mathbf{u}_1, \ldots, \mathbf{u}_N\} \subset [0,1)^d$ be a deterministic point set that is constructed according to a quasi-Monte Carlo rule with low discrepancy, such as a lattice rule (Korobov), or a digital net (Sobol', Faure, Niederreiter); see Lemieux (2006). The quasi-Monte Carlo approximation of ℓ would be

$$\sum_{j=1}^{N} H(\mathbf{u}_j).$$

This deterministic approach is transformed into Monte Carlo simulation by applying a randomization of the point set. A simple randomization technique is the random shift: generate m IID random vectors $\mathbf{v}_i \in [0,1)^d$, $i = 1, \ldots, m$, and compute the quasi-Monte Carlo approximations

$$\hat{\ell}_i = \sum_{j=1}^{N} H(\mathbf{u}_j + \mathbf{v}_i \mod 1).$$

Then the randomized quasi-Monte Carlo estimator using sample size $N \times m$ is defined by

$$\hat{\ell}^* = \frac{1}{m}\sum_{i=1}^{m} \hat{\ell}_i\,.$$

The scrambling technique is based on permuting the digits of the coordinates u_j.. Other techniques of randomizing quasi-Monte Carlo point sets are less used. The main property is that when the performance function H is sufficiently smooth, these randomized quasi-Monte Carlo methods give considerable variance reduction (Lemieux 2006).

See

- Cross-Entropy Method
- Inverse Transform Method
- Markov Chain Monte Carlo
- Rare Event Simulation
- Regenerative Simulation
- Simulation of Stochastic Discrete-Event Systems

References

Asmussen, S., & Glynn, P. W. (2007). *Stochastic simulation*. New York: Springer-Verlag.

Botev, Z. I., & Kroese, D. P. (2008). An efficient algorithm for rare-event probability estimation, combinatorial optimization, and counting. *Methodology and Computing in Applied Probability, 10*, 471–505.

Chen, X., Ankenman, B., & Nelson, B. L. (2010). *The effects of common random numbers on stochastic kriging metamodels* (Working Paper). Evanston, IL: Department of Industrial Engineering and Management Sciences, Northwestern University.

Chick, S. E., & Gans, N. (2009). Economic analysis of simulation selection problems. *Management Science, 55*, 421–437.

Chih, M. (2013). A more accurate second-order polynomial metamodel using a pseudo-random number assignment strategy. *Journal of the Operational Research Society, 64*, 198–207.

de Boer, P. T., & Nicola, V. F. (2002). Adaptive state-dependent importance sampling simulation of Markovian queueing networks. *European Transactions on Telecommunications, 13*, 303–315.

Dean, T., & Dupuis, P. (2008). Splitting for rare event simulation: A large deviations approach to design and analysis. *Stochastic Processes and their Applications, 119*, 562–587.

Dupuis, P., Sezer, D., & Wang, H. (2007). Dynamic importance sampling for queueing networks. *The Annals of Applied Probability, 17*, 1306–1346.

Dupuis, P., & Wang, H. (2007). Subsolutions of an Isaacs equation and efficient schemes for importance sampling. *Mathematics of Operations Research, 32*, 723–757.

Glasserman, P. (2003). *Monte Carlo methods in financial engineering*. New York: Springer-Verlag.

Glasserman, P., Heidelberger, P., Shahabuddin, P., & Zajic, T. (1999). Multilevel splitting for estimating rare event probabilities. *Operations Research, 47*, 585–600.

Juneja, S. and Shahabuddin, P. (2006). Rare-event simulation techniques: An introduction and recent advances. In S. G. Henderson & B. L. Nelson (Eds.), *Handbook in operations research and management science, vol. 13: Simulation* (Chap. 11, pp. 291–350). Amsterdam: Elsevier.

Kaynar, B., & Ridder, A. (2010). The cross-entropy method with patching for rare-event simulation of large Markov chains. *European Journal of Operational Research, 207*, 1380–1397.

Kelton, W. D., Sadowski, R. P., & Sturrock, D. T. (2007). *Simulation with Arena* (4th ed.). Boston: Mc Graw-Hill.
Kleijnen, J. P. C. (2008). *Design and analysis of simulation experiments*. New York: Springer.
L'Ecuyer, P. (2006). Uniform random number generator. In S. G. Henderson & B. L. Nelson (Eds.), *Handbook in operations research and management science, vol. 13: Simulation* (Chap. 3, pp. 55–81).
L'Ecuyer, P., Blanchet, J. H., Tuffin, B., & Glynn, P. W. (2010). Asymptotic robustness of estimators in rare-event simulation. *ACM Transactions on Modeling and Computer Simulation, 20*(1), Article 6.
L'Ecuyer, P., Demers, V., & Tuffin, B. (2007). Rare events, splitting, and quasi-Monte Carlo. *ACM Transactions on Modeling and Computer Simulation, 17*(2), Article 9.
Lagnoux, A. (2006). Rare event simulation. *Probability in the Engineering and Informational Sciences, 20*, 45–66.
Law, A. M. (2007). *Simulation modeling & analysis* (4th ed.). Boston: McGraw-Hill.
Lemieux, C. (2006). Quasi-random number techniques. In S. G. Henderson & B. L. Nelson (Eds.), *Handbook in operations research and management science, vol. 13: Simulation* (Chap. 12, pp. 351–379).
Melas, V. B. (1997). On the efficiency of the splitting and roulette approach for sensitivity analysis. In *Proceedings of the 1997 Winter simulation conference*, pp. 269–274.
Ridder, A. (2010). Asymptotic optimality of the cross-entropy method for Markov chain problems. *Procedia Computer Science, 1*, 1565–1572.
Ridder, A., & Taimre, T. (2011). State-dependent importance sampling schemes via minimum cross-entropy. *Annals of Operations Research, 189*(1), 357–388.
Rubino, G., & Tuffin, B. (Eds.). (2009). *Rare event simulation using Monte Carlo methods*. Wiley.
Rubinstein, R. Y. (2010). Randomized algorithms with splitting: Why the classic randomized algorithms do not work and how to make them work. *Methodology and Computing in Applied Probability, 12*, 1–50.
Rubinstein, R. Y., & Kroese, D. P. (2004). *The cross-entropy method: A unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning*. New York: Springer.
Rubinstein, R. Y., & Kroese, D. P. (2008). *Simulation and the Monte Carlo method*. New York: Wiley.
Rubinstein, R. Y., & Marcus, R. (1985). Efficiency of multivariate control variables in Monte Carlo simulation. *Operations Research, 33*, 661–667.
Villen-Altamirano, M., & Villen-Altamirano, J. (1994). RESTART: A straightforward method for fast simulation of rare events. In *Proceedings of the 1994 Winter simulation conference*, pp. 282–289.

Vector Maximum Problem

► Multiobjective Programming

Vector Space

A vector *n*-space is a set of vectors or points, each with *n* components, and rules for vector addition and multiplication by real numbers. Euclidean 3-space is a vector space.

Vehicle Routing

Lawrence Bodin
University of Maryland, College Park, MD, USA

Introduction

The traditional point-to-point vehicle routing problem (K-VRP) determines a minimum cost set of routes for a fleet of K identical vehicles where the vehicles service a set of locations and each location has a known demand for service. For these problems, minimum cost can represent the 1) minimum dollars to service the locations or 2) minimum nonproductive operating cost to service the locations or 3) minimum travel distance to service the locations. If the fleet size (number of vehicles in the fleet) is known, then the routes are formed in most cases to minimize the total cost associated with the nonproductive travel time (also known as deadhead travel time) associated with the routes that are formed. If the fleet size is not known, then the routes are formed in order to minimize the total cost of the operation, where the total cost of the operation is a combination of the capital cost associated with the fleet size and the operating cost associated with the deadhead travel time of the fleet.

Generating and Executing Routes

When the routes are generated is an important consideration in solving these vehicle routing problems. In some vehicle routing problems, the routes for the vehicles are formed for a single day and these routes are determined well in advance of when the routes are executed. These problems are called fixed route problems. In the preplanned vehicle routing problems, the vehicles are assumed to carry out different routes daily since the locations to be serviced can differ from one day to the next but all of the

locations to be serviced on a given day are known in advanced. In these cases, the vehicle routing problem is solved daily. Most of the traditional papers on vehicle routing are concerned with variants of the fixed route and preplanned vehicle routing problems. These problems are the focus of this article.

However, there are other classes of vehicle routing problems that can be defined where these classes are based on when the routes are generated. In some cases, the routes begin with a skeleton routes or partial routes formed over the locations that are always serviced on that day. Then, new locations for that day are inserted onto these routes to determine a set of routes for that day. Adding locations to routes as the locations are received by the dispatcher generate a class of routing problems called the real time vehicle routing problem or real time dispatching problem.

Traditional Vehicle Routing Constraints

Traditional vehicle routing constraints include the following:

1. Each route can be no longer than a specified length.
2. The volume or weight on each route can be no larger than a specified amount, called the capacity of the vehicle.
3. Each route begins and ends at the depot so that these problems are called single depot vehicle routing problems.
4. All vehicles are identical so that this vehicle routing problem is called a homogeneous routing problem.

Time Windows

A time window at a location to be serviced is defined as a time interval $[L, U]$ where L is the earliest possible time to begin the service at the location and U is the latest possible time to begin the service at the location. If a location has a hard time window, It is normally assumed that the time window is hard, i.e., the service of the location must begin between L and U. In a route, if the vehicle can arrive at the location before L, the hard time window forces the vehicle to wait until L to begin service at the location. The time that the vehicle must wait before beginning the service at L is called the wait time of the vehicle at the location. Vehicle wait time represents nonproductive time and can be considered a cost to the organization. With hard time windows, it is assumed that it is infeasible to begin to service a location after U. If is desired that the service at a location is to be carried out between L and U but the service can begin before L or after U, then $[L, U]$ is called a soft time window at the location.

Vehicle routing problems then fall into two classes – vehicle routing problems without time windows and vehicle routing problems with time windows. If the locations in a vehicle routing problem have time windows, then the windows can be soft or hard. Further, a location without a time window can be assumed to be a location with a time window where $L = 0$ and U is very large.

Vehicle Routing Literature

Vehicle routing problems form a rich area of research and applications. There are many vehicle routing problems that can be defined, where the analysis of these vehicle routing problems depend upon the conditions placed on the problem. As such, a vast literature on vehicle routing exists. Standard references include Ball (1995a; 1995b), Bodin (1990), Bodin et al. (1983), Golden and Assad (1986, 1988) and Lawler et al. (1985). Three books on vehicle routing are Dror (1999), Hall (1999) and Toth and Vigo (1999). Each of these books and papers include extensive bibliographies.

Practical Vehicle Routing Problems

There are many applications of vehicle routing problems and effective software has been developed for solving many of these applications. Practical vehicle routing problems include the delivery of goods from a depot to a set of locations, residential and containerized sanitation pickup, scheduling of meter readers, scheduling of field maintenance personnel, delivery of newspapers and telephone books, scheduling of fuel deliveries such as propane gas and gasoline, scheduling of paratransit vehicles, and scheduling of pickups and deliveries for courier services.

Some of the more common constraints that can be encountered when attempting to solve practical vehicle routing problems are as follows:

1. The length of each route must be between a prespecified lower and upper bound.
2. Each route begins and ends at the same depot although in some problems, there can be several such depots. This problem is called the multiple depot vehicle routing problem. Further, in some

problems, the route can begin (or end) at a depot and end (or begin) at the last (or first stop) on the route.

Problems have been encountered where vehicles leave a depot (or storage facility), deliver items to several locations, go to a second depot (or storage facility), reload the vehicle, deliver items to several locations, go to a third depot (or storage facility), etc. These problems can involve developing routes over several days.

The rollon-rolloff problem is an example where (i) a tractor leaves a depot, (ii) goes to a location with an empty container, (iii) exchanges the empty container for a full container at the location, (iv) brings the full container to a landfill or storage facility, (v) exchanges the full container for an empty container, (vi) brings the empty container to a location, (vii) exchanges the empty container for a full container at the location, (viii) brings the full container to a landfill or storage facility, (ix) exchanges the full container for an empty container, etc. In the rollon-rolloff problem, there can be multiple landfills and storage facilities and the vehicle capacity is 1 (the tractor can only move one container at a time).

3. There can be several types of vehicles in the fleet where the vehicles can be different in terms of capacity, size of crew, speed, etc. This problem is called the multiple vehicle type routing and scheduling problem.
4. If some of the locations can be serviced by some, but not necessarily all, of the vehicle types and the specification of the vehicle types that can service a location can differ by location, then this problem is called the vehicle/location or vehicle/site dependency routing problem.
5. If demand at a location is not known in advance but can be stochastic, then this problem is called the stochastic vehicle routing and scheduling problem. A variant of the stochastic vehicle routing and scheduling problem, called the inventory routing problem, occurs in the delivery of such items as propane gas and fuel oil. In the case of propane gas or fuel oil, the demand (amount of propane gas or fuel oil needed at the location) is forecasted using a factor such as degree-days.
6. In paratransit, courier delivery and shared cab ride problems, each customer demanding service has a specified pickup location and a specified delivery location. The pickup has to be scheduled on the route before the delivery is scheduled on the route. Some of these problems can have transshipments; that is to say, packages are picked up and brought to a pre-specified drop location where they are unloaded and another vehicle later picks up these packages and makes the deliveries.
7. The vehicle routing problem with backhauling occurs In some vehicle routing problems when there are delivery locations and pickup locations and each route has the restriction that all (or most) of the deliveries are to be carried out before any of the pickups are to be carried out. In this way, the vehicle can be close to empty before it is filled up. An example of this problem is when a vehicle makes several deliveries and then goes to a warehouse where it reloads the vehicle to bring the items that has just been loaded on the vehicle to another warehouse or depot.

   A second example is the local delivery and pickup routes for organizations like Federal Express and UPS. In this example, the drivers make deliveries in the morning, then make later deliveries and early pickups in the midday, and then mostly pickups at the end of the day. These pickups are then returned to the depot where they are processed for delivery. This problem involves both daily scheduled pickups and daily real time dispatching.

Algorithms for Solving Vehicle Routing Problems

Virtually all vehicle routing problems fall into the class of combinatorial optimization problems called *NP*-Hard. A problem is *NP*-Hard if the number of computations needed to solve this problem grows exponentially with a parameter of the problem (Garey and Johnson (1979), Karp (1975), Lenstra and Rinnooy Kan (1981) and Papadimitriou and Stieglitz (1982)). Many of the algorithms for solving vehicle routing problems can be divided into the following three classes – heuristic approaches, metaheuristic approaches and exact procedures.

Heuristic and Metaheuristic Algorithms

Since finding the optimal solution for reasonable size problems and proving that the solution is optimal is difficult, heuristic approaches are generally employed to find a close-to-optimal solution for these problems.

Many of the papers and books in the literature describe various heuristic approaches for solving vehicle routing problems. These approaches are surveyed in Ball (1995a; 1995b), Bodin (1990), Bodin et al. (1983), Golden and Assad (1986, 1988) and Toth and Vigo (1999). Further, metaheuristics such as tabu search and neural networks (Martello et al. 1999; Toth and Vigo 1999) have been applied with some success to finding reasonable solutions to these vehicle routing problems.

A standard heuristic approach for solving many point-to-point vehicle routing and scheduling problems consists of the following steps:

(a) *Specify K, the fleet size*. The fleet size K is generally set by the user or determined by some estimation procedure.

When practical considerations are considered, determining the value of K (the number of vehicles in the fleet) may be very difficult to estimate accurately. As such, the user may not know how many routes to form. To overcome this issue, the user can always repeat the heuristic approach for different values of K and take the best solution.

(b) *Tour Construction or Partitioning of the Locations*. The locations to be serviced are aggregated into K clusters. Some of the approaches for tour construction use single location insertion heuristics and are sequential in nature (one location is assigned to a cluster on each iteration). Other approaches, such as the generalized assignment algorithm, are based on solving a mathematical program and are not sequential in nature.

In some cases, the locations are sequenced as the partitions are formed. If routes and schedules have not been formed while aggregating locations into partitions, then a route and schedule is found over the locations assigned to each of the K partitions, one cluster at a time.

Depending upon the algorithm being implemented and the constraints on the problem, at the conclusion of this step, it is possible to have some locations that are not assigned to routes and/or some of the routes to violate the upper bound on travel time.

(c) *Tour Improvement*. In Tour Improvement, the total travel time of each of the routes is reduced by reordering the locations. To accomplish this, the following is carried out:

(i) The locations on each route are reordered, one route at a time. There are very popular and effective procedures designed for carrying out this exchange process. The number of exchanges of this type is a function of the implementation of the algorithms used and the amount of computer time available.

(ii) Locations are moved between routes and the routes regenerated to reduce their lengths. The results of exchanging locations between routes is not nearly as effective as the within route exchanges described in c-i. Thus, this approach has to be used with some caution because it could use up significant computer time and find few route improvements.

(iii) Unassigned locations are inserted into routes using some of the insertion procedures described in step (b). This approach can be integrated with the approaches in c-i and c-ii.

The tour improvement step continues until no more improvements are found or the time allocated to tour improvement is exhausted.

Many of the tour construction approaches (Step B above) are sequential in nature in that one location is assigned to one of the K routes on each iteration. As such, a bad decision of assigning a customer to a route made at an early step in the tour construction part of the above approach locks in the solution being generated. This bad decision can adversely affect the subsequent assignment of locations to routes.

Moreover, the tour improvement procedures [Step (c) above] can be either too time consuming or not powerful enough to derive a close to optimal solution when starting from an initial assignment of locations to routes that is inferior (these routes are formed in the tour construction approach (Step B above). Despite these caveats, this approach, from a research standpoint and in commercially available software, has served as one of the 'workhorse' procedures for solving vehicle routing problems.

The above approach did not explicitly mention time windows. Adapting this approach to solve a VRP with time windows may not be effective if the time windows are hard and the time window duration D = U-L at some of the locations is narrow (say D < 1 hour).

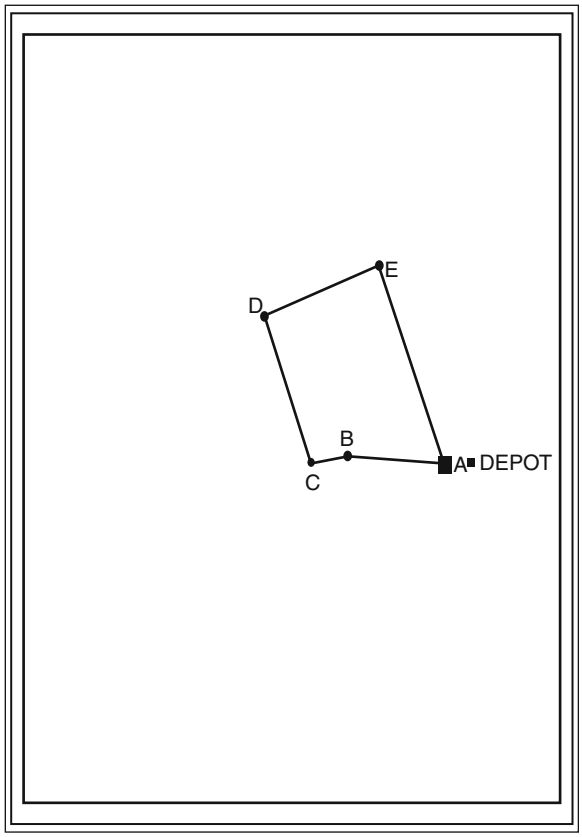
Mathematical Programming Approaches

Mathematical programming approaches have been developed for solving certain vehicle routing problems with at least 150 locations exactly. Bodin, Mingozzi and Maniezzo (1999) survey some of the

more promising approaches for solving vehicle routing exactly. Subsequent papers by Aristide Mingozzi and his colleagues describe effective exact approaches for solving various classes of vehicle routing problems. The results in these papers show that the exact approach taken in these papers get superior results to virtually all of the test problems that have been created for solving different and difficult vehicle routing problems.

Classes of Vehicle Routing Problems

The above discussion has concentrated on solving point-to-point vehicle routing problems (also known as node routing problems). In a point-to-point vehicle routing problem, the locations are distinct points to be serviced and these points are scattered over a region. It is generally assumed that in a node routing problem, the density of the service points is not too great so that the resulting travel path for the vehicle is reasonably independent of the underlying street network. Traditionally, the Euclidean distance (or travel time based on the Euclidean distance or some other metric based on Euclidean disatnace) is used as the deadhead travel time metric when solving point-to-point vehicle routing problems.



Vehicle Routing, Fig. 1 Euclidean distance route over five points

Traveling Salesman Problem

The traveling salesman problem is the one vehicle point-to-point vehicle routing problem. The optimal solution to the traveling salesman problem requires the determination of a minimum deadhead time path that services each location demanding service exactly once. This route begins and ends at the depot. A traveling salesman solution is displayed in Fig. 1. The route in Fig. 1 represents a solution to a symmetric or undirected traveling salesman problem since the travel time between each pair of locations does not depend on the direction of travel. An asymmetric or directed traveling salesman problem occurs when the travel time between each pair of locations can be different.

There has been considerable research in developing efficient and accurate procedures for solving the traveling salesman problem. Helsgaun (Helsgaun 2009 and Applegate et al. 2009) has developed extremely effective approaches for solving the traveling salesman problem with as many as 10,000,000 nodes and has demonstrated that his computationally efficient procedure has found a solution to these very large traveling saleman problems that is within .02% of the optimal solution.

Arc Routing Problems

A second class of routing problems are called arc-routing problems. In an arc-routing problem, the entities to be serviced are the arcs in a network, rather than the individual locations in node routing problems discussed earlier. The methodology for solving arc routing problems is similar to the methodology for solving node routing problems. Matter of fact, a question that often arises is should one (i) convert an arc routing problem to a node routing problem and solve the node routing problem or (ii) convert a node routing problem to an arc routing problem and solve the arc routing problem (assuming that the locations to be serviced are geocoded onto a network). At this time, there is no definitive answer to this question although

algorithms for solving node routing problems tend to allow for additional constraints to be considered as compared to arc routing problems. Dror's book on arc routing problems is devoted totally to the formulation, solution methodologies and practical applications of arc routing problems (Dror 1999). Many of the other references cited earlier also have sections or chapters on arc routing problems.

Chinese Postman Problems

The single vehicle version of an arc routing problem is called the Chinese Postman Problem (given the name by Meigu Guan or Mei-Ko Kwan, see Kwan (1962)). In the Chinese Postman problem, there are streets in a network that require service and the subgraph made up of the streets requiring service is connected. The problem is to develop a travel path (called an Euler path) that services all of the streets requiring service where the deadhead time is minimzed for the additional streets added to the required streets to allow for an Euler path to be found.

A simple approach for solving the undirected Chinese Postman Problem (all arcs in the network are undirected) is to sove a 1-match problem to determined the minimum cost additional arcs to be added into the problem to allow for an Euler path to be found over the subgraph of required arcs and deadhead arcs. In a similar vein, the determination of the minimum deadhead arcs in the directed Chinese Postman Problem is found by solving a transportation problem.

Since the algorithms for optimally solving the 1-match problem and the transportation problems are polynomial, the undirected and directed Chinese Postman problems are not NP-Hard. The Chinese Postman Problem becomes NP-Hard when some of the arcs are directed and others are not directed. In the case where the arcs to be serviced are a subset of all of the arcs in the network, then the problem is not NP-Hard if all the arcs requiring service are either directed or undirected and the network of all streets requiring service is connected.

The famous Swiss mathematician, Leonhard Euler, solved the first undirected Chinese Postman Problem in the famous problem called the Seven Bridges of Koenigsburg (Prussia). Because of this problem, Euler is generally credited with originating Graph Theory (Assad 2007). A special issue of *Networks* was devoted to celebrating Euler's 300th birthday (Golden and Shier 2007). This special issue of *Networks* also contained a delightful article on the present status of the Seven Bridges of Koenigsburg (Gribkovskaia et al. 2007).

Neighborhood Vehicle Routing Problems

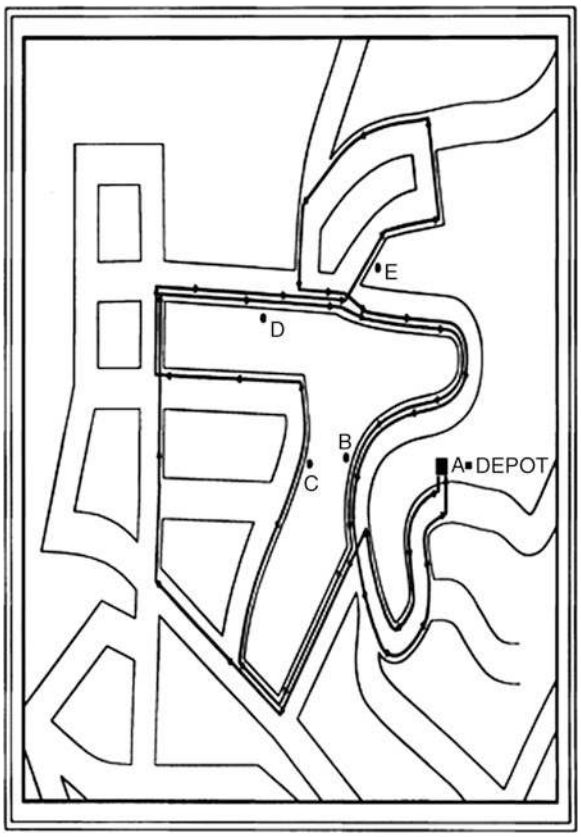
A second class of vehicle routing problems are called neighborhood routing (or arc routing) problems. In a neighborhood vehicle routing problem, the locations are arcs in the underlying network to be serviced. In solving the neighborhood vehicle routing problem, the arcs are partitioned into subsets and, within each subset, the locations are ordered to form a minimum travel time path. Traditionally, when solving neighborhood vehicle routing problems, the shortest travel time path between street segments requiring service is used as the deadhead travel time metric.

The capacitated arc routing problem is another example of a neighborhood routing problem. In the undirected (directed) capacitated arc routing problem, a network is given where every arc in this network is either directed or undirected and the network is connected. Moreover, each arc has a known demand and each vehicle has the same capacity Q. The problem is to break this network down into partitions where each arc is assigned to a partition, the demand in each partition is no greater than Q, a travel path can be found over the arcs in each partition that traverses all of the arcs in that partition and the total deadhead travel time over all of the vehicles is minimized.

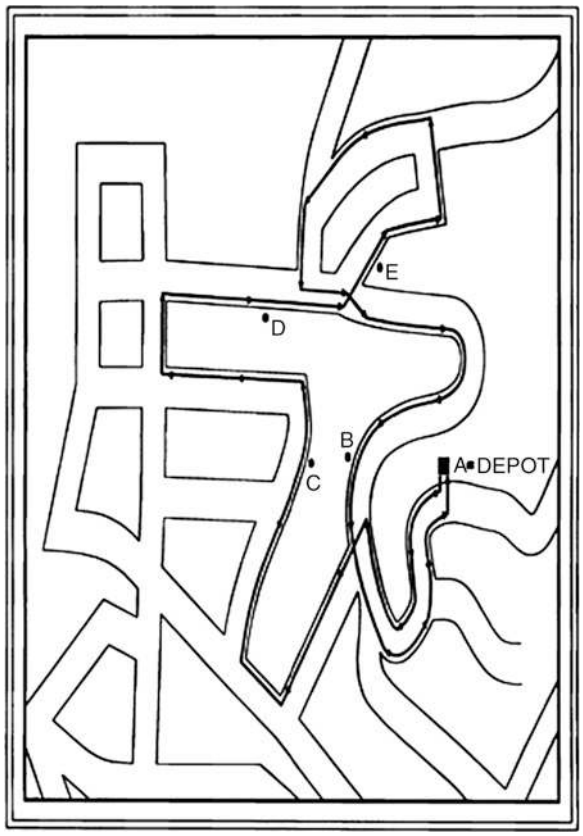
Street Routing and Scheduling Problems

As noted earlier, the original procedures for solving the point-to-point vehicle routing problems assumed that Euclidean distance was used to determine the distance or travel time between the locations to be serviced in a point-to-point vehicle routing problem. This assumption worked reasonably well as long as the locations to be serviced were scattered over a region and not too dense. However, the Euclidean distance assumption may not be realistic when solving practical vehicle routing problems where the locations to be serviced are somewhat dense and/or the street network on which the locations are geocoded have to be taken into consideration.

A five-location traveling salesman solution, A-B-C-D-E-A, using Euclidean distances was displayed in Fig. 1. Assume that location A is the depot in Figs. 2 and 3.



Vehicle Routing, Fig. 2 Actual travel path for Euclidean distance route in Fig. 1. (Notes for Fig. 2: Order of the servicing the locations on the route is maintained as in Fig. 1. No U-turns are allowed. Service is carried out on the location's side of the street without making a left hand turn. To get from location A to location B without U-turns and ensuring that service is carried out on the location's side of the street, the route has to go around several blocks)



Vehicle Routing, Fig. 3 Travel path when shortest paths are computed over the street network. (Notes on Fig. 3: No U-turns are allowed. Service is carried out on the location's side of the street without making a left hand turn)

In Fig. 2, the same traveling salesman solution (A-B-C-D-E-A) is displayed but the locations to be serviced are superimposed on a street network. In this solution, the vehicle would drive past locations and not service these locations and then have to service these locations later in the route, increasing the total travel time of the route.

The solution in Fig. 3 is generated by solving a traveling salesman problem where the shortest travel time path between each pair of locations is used rather than Euclidean distances. In this solution, no U-turns are allowed and the vehicle is forced to traverse each street segment on the right hand side. In this solution, the vehicle always services locations as it drives past these locations.

A special class of practical vehicle routing problems are called street routing and scheduling problems. Street routing and scheduling problems consist of point-to-point routing problems where locations to be serviced are dense and arc routing problems. In street routing problems, the locations to be serviced are located on a digital map and the deadhead travel times are computed as shortest paths rather than Euclidean distances. In arc routing problems, most street segments in the region are to be serviced. Street routing and scheduling problems are described in detail in Bodin, Mingozzi and Maniezzo (1999).

Geographic Information Systems and Digital Street Networks

To solve street routing and scheduling problems requires an accurate digital street networks and

a Geographic Information System (GIS). A digital street network is a street segment by street segment representation of a geographic region. A *GIS* is a system of computer hardware, software and procedures designed to support the capture, management, manipulation, analysis, modeling and display of a digital street network. With an accurate GIS, the user is able to address match the locations to be serviced on the digital street network, to compute the travel times between locations as shortest travel-time paths, and to give accurate street-by-street travel directions for each vehicle route. On the other hand, the Euclidean distance approach for solving vehicle routing problems only gives an ordering of the locations of a route could be derived.

Most commercially available vehicle routing systems use a digital street network and a GIS in their routing and scheduling procedures. The generation of accurate travel paths is essential for having these travel paths accepted by the drivers and management. Users do not want strange turns on their routes and they wish to be able to give accurate locations for all of the locations that they service. In this regard, early procedures for solving street routing and scheduling problems were concerned about adverse turns in their travel paths and developed approaches for reducing U-turns and left-hand turns on their travel paths (assuming vehicles are driven on the right hand side of the street) (McBride (1982)). UPS announced that they had developed the travel paths for their drivers that emphasized right hand turns. As a result, they saved $600 million per year (Farber (2005)).

Concluding Remarks

Due to the increase in the capabilities of computers and the increase in functionality and sophistication of software, it is now possible to derive better solutions to larger (in terms of number of locations to be served) and more varied (in terms of the constraints that can be considered) vehicle routing problems. Moreover, the computer systems that solve these systems have improved graphics, user interfaces and underlying geographic data. Since the cost of distribution is a major cost component of many organizations, computerized vehicle routing systems are becoming a necessary part of an organization's logistics/distribution system.

See

- ▶ Chinese Postman Problem
- ▶ Computational Complexity
- ▶ Geographic Information Systems
- ▶ Graph Theory
- ▶ Heuristics
- ▶ Logistics and Supply Chain Management
- ▶ Metaheuristics
- ▶ Neural Networks
- ▶ Supply Chain Management
- ▶ Tabu Search
- ▶ Traveling Salesman Problem
- ▶ Visualization

References

Applegate, D. L., Bixby, R. E., Chvatal, V., Cook, W., Espinoza, D. G., Goycoolea, M., & Kelsgaun, K. (2009). Certification of an optimal TSP tour through 85,900 cities. *Operations Research Letters, 37*, 11–15.

Assad, A. (2007). Leonhard euler: A brief apreciation. *Networks, 49*, 190–198.

Baldacci, R., Christofides, N., & Mingozzi, A. (2008). An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming Series A, 115*(2), 51–385.

Baldacci, R., & Mingozzi, A. (2008). A unified exact method for solving different classes of vehicle routing problems. *Mathematical Programming Ser. A* http://dx.doi.org/10.1007/s 10107-008-0218-9.

Ball, M., Magnanti, T. L., Monma, C., & Nemhauser, G. L. (Eds.). (1995a). *Network models, handbooks in operations research and management science* (Vol. 7). Amsterdam: North-Holland.

Ball, M., Magnanti, T. L., Monma, C., & Nemhauser, G. L. (Eds.). (1995b). *Network routing, handbooks in operations research and management science* (Vol. 8). Amsterdam: North-Holland.

Bodin, L. (1990). Twenty years of routing and scheduling. *Operations Research, 38*, 571–579.

Bodin, L., Golden, B. L., Assad, A., & Ball, M. (1983). Routing and scheduling of vehicles and crews: The state of the art. *Computers and Operations Research, 10*(2), 63–211.

Bodin, L., & Kursh, S. (1999). A computer-assisted system for the routing and scheduling of street sweepers. *Operations Research, 26*(4), 525–537.

Bodin, L., Mingozzi, A., & Maniezzo, V. (1999). Street routing and scheduling problems. In R. W. Hall (Ed.), *Handbook of transportation science*. Norwell, MA: Kluwer.

Dror, M. (Ed.). (1999). *Arc routing: Theory, solutions, and applications*. Norwell, MA: Kluwer.

Farber, D. (2005). UPS: Driving cost savings by eliminating left-hand turns, ZDNET.com.

Garey, M., & Johnson, D. (1979). *Computer and intractibility: A guide to the theory of NP-completeness*. San Francisco: Freeman Press.
Golden, B. L., & Assad, A. (Eds.). (1986). Special Issue on time windows. American Jl. *Mathematical and Management Sciences*, *6*(3 and 4), 251–399.
Golden, B. L., & Assad, A. (1988). *Vehicle routing: Methods and studies*. Amsterdam: North-Holland.
Golden, B. L., & Shier, D. (Eds.). (2007). Special issue dedicated to Leonhard Euler. Networks, Vol. 49, pp. 189–242.
Gribkovskaia, I., Halskau, O., Sr., & LaPorte, G. (2007). The bridges of Konigsberg – A historical perspective. *Networks, 49*, 199–203.
Hall, R. (Ed.). (1999). *Handbook of transportation science*. Norwell, MA: Kluwer.
Helsgaun, K. (2009). Generak k-opt submoves for the Lin-Kernighan TSP heuristic. *Mathematical Programming Computation*. doi:10. 1007/s12532-009-0004-6.
Karp, R. (1975). On the computational complexity of combinatorial problems. *Networks, 5*, 45–68.
Kwan, M.-K. (1962). Graphic programming using odd or even points. *Chinese Math.*, *1*, 273–277.
Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G., & Shmoys, D. B. (Eds.). (1985). *The traveling salesman problem*. Chichester, UK: John Wiley.
Lenstra, J. K., & Rinnooy Kan, A. (1981). Complexity of vehicle routing and scheduling problems. *Networks, 11*, 221–227.
Martello, S., Osman, I. H., & Roucairol, C. (Eds.). (1999). *Meta-heuristics: Advances and trends in local search paradigms for optimization*. Norwell, MA: Kluwer.
McBride, R. (1982). Controlling left and U-turns in the routing of refuse collection vehicles. *Computers and Operations Research, 9*, 145–152.
Papadimitriou, C., & Stieglitz, K. (1982). *Combinatorial optimization*. Englewood Cliffs, NJ: Prentice Hall.
Toth, P., & Vigo, D. (Eds.). (1999). The vehicle routing problem, SIAM series on discrete mathematics and its applications, Philadelphia.

Vehicle Scheduling

► Vehicle Routing

Verification

For a mathematical model, especially a computer-based one such as a simulation model, verification is the process by which the computational procedure (computer program or software) is checked to determine if it is error free (debugged) and the determination that the model, as represented by the calculations or software, does what the analyst intended. A model is said to be verified if it (the computation) correctly executes the intended calculations.

See

► Validation
► Verification, Validation, and Testing of Models

Verification, Validation, and Testing of Models

Osman Balci
Virginia Polytechnic Institute & State University, Blacksburg, VA, USA

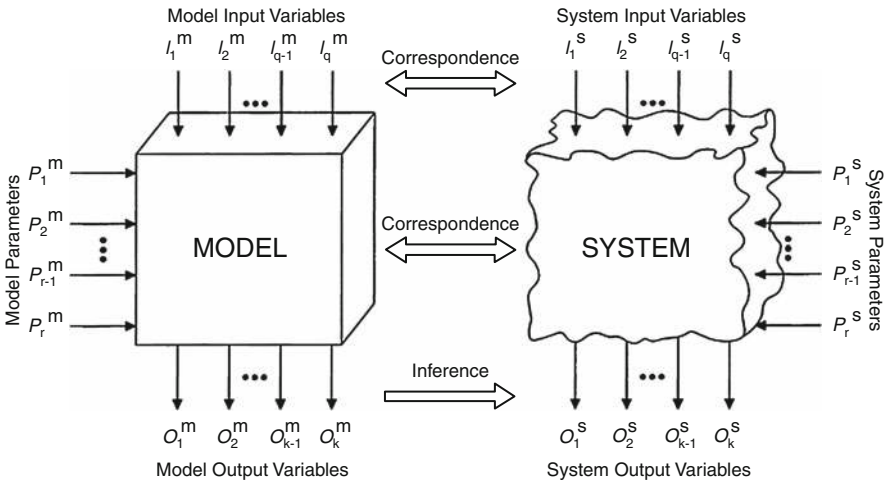
Introduction

Operations research/management science (OR/MS) models lacking a sufficiently accurate representation produce erroneous results that can be catastrophic when making critical decisions based on the model results. Thus, principles and techniques for verification, validation and testing (VV&T) of the OR/MS models are critical for their successful implementation and utilization. After presenting some background information, the principles are introduced and a taxonomical brief overview of the techniques is given.

Background

A model is a representation and an abstraction of anything such as a system, concept, problem, or phenomena. It can have inputs, parameters, and outputs as illustrated in Fig. 1. The term system is used to refer to whatever the model represents.

Model Verification is substantiating that the model is transformed from one form into another, as intended, with sufficient accuracy. Model verification deals with building the model right. The accuracy of transforming a problem formulation into a model specification or the accuracy of converting a model representation in micro flowchart into an executable computer program is evaluated in model verification.



Verification, Validation, and Testing of Models, Fig. 1 Model and system characteristics

Model Validation is substantiating that the model, within its domain of applicability, behaves with satisfactory accuracy consistent with the study objectives. Model validation deals with building the right model. It is conducted by executing/running the model under the same input conditions that drive the system and by comparing model behavior with the system behavior. (Note that a linear programming model is executed and a simulation model is run).

Model Testing is demonstrating that inaccuracies exist in the model or revealing the existence of errors in the model. In model testing, the model is subjected to test data or test cases to see if it functions properly. Test failed implies the failure of the model, not the test. Testing is conducted to perform verification and validation. Some tests are intended to judge the accuracy of model transformation from one form into another (verification). Some tests are devised to evaluate the behavioral accuracy (i.e., validity) of the model. Therefore, the whole process is commonly referred to as model VV&T.

Model VV&T is conducted to prevent occurrences of three major types of errors in OR/MS modeling studies (Balci 1998b): Type I Error is the error of rejecting the model credibility when in fact the model is sufficiently credible; Type II Error is the error of accepting the model credibility when in fact the model is not sufficiently credible; and Type III Error is the error of solving the wrong problem. The probability of committing the Type I Error is called Model Builder's Risk and probability of committing the Type II Error is called Model User's Risk. Committing the Type I error increases the cost of model development. The consequences of committing the Type II and Type III errors may be catastrophic. Therefore, a cost-risk analysis should be conducted whenever possible (Balci and Sargent 1981).

Principles

The principles presented here are established based on the experience described in the published literature and the author's experience (Balci 1998b, 2010). The principles are listed below in no particular order.

Principle 1: The model VV&T must be conducted throughout the entire modeling life cycle starting with problem formulation and culminating with the presentation of model results. The VV&T activities throughout the entire life cycle are intended to reveal and rectify quality deficiencies during the life cycle phase in which they occur.

Principle 2: The outcome of model VV&T should not be considered as a binary variable where the model is absolutely correct or absolutely incorrect. Since a model is an abstraction of an entity, perfect representation is never expected. The outcome of model VV&T should be considered as a degree of credibility on a scale from 0 to 100, where 0 represents absolutely incorrect and 100 represents absolutely correct.

Principle 3: A model is built with respect to the study objectives and its credibility is judged with respect to those objectives. The study objectives dictate how representative the model should be. Sometimes, 60% representation accuracy may be

sufficient; sometimes, 95% accuracy may be required. The adjective "sufficient" must be used in front of the terms such as model credibility, model validity or model accuracy to indicate that the judgment is made with respect to the study objectives.

Principle 4: The model VV&T requires independence to prevent developer's bias. The organization which is contracted to conduct the modeling study is not qualified to perform the final model VV&T (acceptance testing). The sponsor of the modeling study should identify an independent agent to conduct the final model VV&T. To emphasize this principle, VV&T is called independent VV&T or independent V&V (IV&V) by many authors in the literature.

Principle 5: The model VV&T is difficult and requires creativity and insight. Knowledge of the problem domain, expertise in the modeling methodology, and prior modeling and VV&T experience are required. It is not possible for one person to fully understand all aspects of a large and complex model especially if the model is a stochastic one containing hundreds of concurrent activities.

Principle 6: Model credibility can be claimed only for the prescribed conditions for which the model is tested. The accuracy of the input–output transformation of a simulation model is affected by the characteristics of the input conditions. The transformation that works for one set of input conditions may produce absurd output when conducted under another set of input conditions.

Principle 7: Complete model testing is not possible. Exhaustive (complete) testing requires testing the model under all possible inputs. Combinations of feasible values of model input variables can generate millions of logical paths in model execution. Due to time and budgetary constraints, it is impossible to test the accuracy of millions of logical paths. Therefore, in model testing, the purpose is to increase confidence in model credibility as much as dictated by the study objectives rather than trying to show 100% credibility.

Principle 8: The model VV&T must be planned and documented. Testing is not a phase or step in model development life cycle; it is a continuous activity throughout the entire life cycle. The tests should be identified, test data or cases should be prepared, tests should be scheduled, and the whole testing process should be documented. All test data and cases must be preserved for use in model maintenance and regression testing.

Principle 9: Type I, II and III errors must be prevented. Committing a Type I Error unnecessarily increases the cost of model development. The consequences of Type II and Type III Errors can be catastrophic especially when critical decisions are made on the basis of model results. Committing a Type III Error implies solving the wrong problem and causes the study results to be irrelevant.

Principle 10: Errors should be detected as early as possible in the life cycle of a modeling study. Correcting errors detected in later phases of the life cycle is much more expensive. Some vital errors may not be detectable in later phases resulting in the occurrence of Type II or Type III error.

Principle 11: Multiple response problems must be recognized and resolved properly. The validity of a model with two or more output variables (responses) cannot be tested by comparing the corresponding model and system output variables one at a time, that is, O_1^m versus O_1^s, O_2^m versus O_2^s, etc. as shown in Fig. 1. A multivariate statistical procedure must be used to incorporate the correlations among the output variables in the comparison.

Principle 12: Double validation problem must be recognized and resolved properly. If data can be collected on both system input and output, model validation can be conducted by comparing model and system outputs obtained by running the model with the same input data that drives the system. Determination of the same is yet another validation problem within model validation. Therefore, this is called the double validation problem.

Principle 13: Successfully testing each submodel (module) does not imply overall model credibility. The credibility of each submodel is judged to be sufficient with some error that is acceptable with respect to the study objectives. Each submodel may be found to be sufficiently credible, but this does not imply that the whole model is sufficiently credible. The allowable errors for the submodels may accumulate to be unacceptable for the whole model. Therefore, the whole model must be tested even if each submodel is found to be sufficiently credible.

Principle 14: Model validity does not guarantee the credibility and acceptability of modeling study results. Model validity is a necessary but not a sufficient condition for the credibility and acceptability of

model results. Model validity is assessed with respect to the modeling study objectives by comparing the model with the system as it is defined. If the study objectives are incorrectly identified and/or the system is improperly defined, the model results will be invalid; however, the model may still be found to be sufficiently valid by comparing it with the improperly defined system and with respect to the incorrectly identified objectives.

Principle 15: Formulated problem accuracy greatly affects the acceptability and credibility of model results. If the problem is formulated incorrectly, no matter how excellent the problem solution is, the modeling study results will be irrelevant.

Techniques

Figure 2 shows a taxonomy that classifies more than 77 VV&T techniques into four primary categories: informal, static, dynamic, and formal (Balci 1998a). The use of mathematical and logic formalism by the techniques in each primary category increases from informal to formal from left to right. Likewise, the complexity also increases as the primary category becomes more formal. The categories and techniques in each category are briefly described below (Balci 1998b).

Informal Techniques

These techniques are among the most commonly used ones. They are called informal because the tools and approaches used rely heavily on human reasoning and subjectivity without stringent mathematical formalism. The informal label does not imply any lack of structure and formal guidelines for the use of the techniques.

Audit is undertaken to assess how adequately the modeling study is conducted with respect to established plans, policies, procedures, standards and guidelines. The audit also seeks to establish trace-ability within the modeling study.

Desk Checking (also known as Self-Inspection) is the process of thoroughly examining one's work to ensure correctness, completeness, consistency and unambiguity. It is considered to be the very first step in VV&T and is particularly useful for the early stages of development.

Documentation Checking is conducted to ensure correctness, completeness, consistency, and unambiguity of all model documentation and to justify that all documentation is up-to-date with respect to model logic specification.

In Face Validation, the project team members, potential users of the model, people knowledgeable about the system under study, based on their estimates and intuition, subjectively compare model and system behaviors under identical input conditions and judge whether the model and its results are reasonable.

Inspections are conducted by a team of four to six members for any model development phase such as model requirements specification, detailed model design, or model code. An inspection goes through five distinct phases: overview, preparation, inspection, rework and follow-up.

Reviews are conducted in a similar manner as the inspections and walkthroughs except in the way the team members are selected. The review team also involves managers. The review is intended to give management and study sponsors evidence that the model development process is being conducted according to stated study objectives.

Turing Test is based on the expert knowledge of people about the system under study. The experts are presented with two sets of output data obtained, one from the model and one from the system, under the same input conditions. Without identifying which one is which, the experts are asked to differentiate between the two. If they succeed, they are asked how they were able to do it. Their response provides valuable information for model validation.

Walkthroughs are conducted by a team composed of a coordinator, model developer and three to six other members. Except the model developer, all other members should not be directly involved in the development effort.

Static Techniques

These techniques are concerned with accuracy assessment on the basis of characteristics of the static model design and source code. Static techniques do not require machine execution of the model, but mental execution can be used.

Cause-Effect Graphing assists model correctness assessment by addressing the question of "what

Verification, Validation, and Testing of Models, Fig. 2 A taxonomy of model VV&T techniques

causes what in the model representation?" It is performed by first identifying causes and effects in the system being modeled and by examining if they are accurately reflected in the model specification.

Calling Structure Analysis is used to assess model accuracy by identifying who calls whom and who is called by whom. The who could be a module, procedure, subroutine, function, or a method in an object-oriented model.

Concurrent Process Analysis is especially useful for parallel and distributed models. Model accuracy is assessed by analyzing the overlap or concurrency of model components executed in parallel or as distributed. Such analysis can reveal synchronization problems such as deadlocks.

Control Flow Analysis requires the development of a graph of the model where conditional branches and model junctions are represented by nodes and the model segments between such nodes are represented by links. A node of the model graph usually represents a logical junction where the flow of control changes, while an edge represents towards which junction it changes. This technique examines sequences of control transfers and is useful for identifying incorrect or inefficient constructs within model representation.

State Transition Analysis requires the identification of a finite number of states the model execution goes through. A state transition diagram is created showing how the model transitions from one state to another. Model accuracy is assessed by analyzing the conditions under which a state change occurs.

Data Dependency Analysis involves the determination of what variables depend on what other variables. For parallel and distributed models, the data dependency knowledge is critical for assessing the accuracy of process synchronization.

Data Flow Analysis is used to assess model accuracy with respect to the use of model variables. It can be used to detect undefined or unreferenced variables and, when aided by model instrumentation, can track minimum and maximum variable values, data dependencies and data transformations during model execution. It is also useful in detecting inconsistencies in data structure declaration and improper linkages among submodels.

In Fault/Failure Analysis, fault implies incorrect model component and failure implies incorrect behavior of a model component. The analysis uses model input–output transformation descriptions to identify how the model might logically fail. The model design specification is examined to determine if any failure-mode possibilities could logically occur and in what context and under what conditions. Such model examinations often lead to identification of model defects.

Model Interface Analysis is conducted to examine the (sub)model-to-(sub)model interface and determine if the interface structure and behavior are sufficiently accurate.

User Interface Analysis is conducted to examine the user-model interface and determine if it is human engineered so as to prevent occurrences of errors during the user's interactions with the model. It is also used to assess how accurately the interface is integrated with the model. This technique is particularly useful for accuracy assessment of interactive models used for training purposes.

Semantic Analysis is conducted by the model's programming language compiler and attempts to determine the modeler's intent in writing the code. The compiler informs the modeler about what is specified in the source code so that the modeler can verify that the true intent is accurately reflected. The compiler generates a wealth of information to help the modeler determine if the true intent is accurately translated into the executable code.

Structural Analysis is used to examine the model structure and to determine if it adheres to structured principles. It is conducted by constructing a control flow graph of the model structure and examining the graph for anomalies, such as multiple entry and exit points, excessive levels of nesting within a structure and questionable practices such as the use of unconditional branches (i.e., GOTOs).

Symbolic Evaluation is used to assess model accuracy by exercising the model using symbolic values rather than actual data values for input. It is performed by feeding symbolic inputs into the (sub) model and producing expressions for the output which are derived from the transformation of the symbolic data along model execution paths.

Syntax Analysis is carried on by the model's programming language compiler to assure that the mechanics of the language are applied correctly.

Traceability Assessment is used to match, one to one, the elements of one form of the model to another. For example, the elements of the model requirements specification are matched one to one to the elements of the model design specification. Unmatched elements may reveal either unfulfilled requirements or unintended design functions.

Dynamic Techniques

These techniques require model execution and are intended for evaluating the model based on its execution behavior. Most dynamic VV&T techniques require model instrumentation. The insertion of additional code (probes or stubs) into the executable model for the purpose of collecting information about model behavior during execution is called model instrumentation. Probe locations are determined manually or automatically based on static analysis of model structure. Automated instrumentation is accomplished by a preprocessor that analyzes the model static structure (usually via graph-based analysis) and inserts probes at appropriate places. Dynamic VV&T techniques are usually applied using the following three steps. In Step 1, the executable model is instrumented. In Step 2, the instrumented model is executed, and in Step 3, the model output is analyzed and dynamic model behavior is evaluated.

Acceptance Testing is conducted either by the model sponsor or an independent contractor hired by the sponsor after the model is officially delivered and before the sponsor officially accepts the delivery. The model is operationally tested by using the actual hardware and actual data to determine whether all requirements specified in the legal con-tract are satisfied.

Alpha Testing refers to the operational testing of the alpha version of the complete model at an in-house site which is not involved with the model development.

In Assertion Checking, an assertion is a statement that should hold true as the model executes. Assertion checking is a verification technique used to check what is happening against what the modeler assumes is happening so as to guard model execution against potential errors. The assertions are placed in various parts of the model to monitor model execution. They can be inserted to hold true globally—for the whole model; regionally—for some submodels; locally—within a submodel; or at entry and exit of a submodel.

Beta Testing refers to the operational testing of the beta version of the complete model at a "beta" user site under realistic field conditions.

Bottom-up Testing is used in conjunction with bottom-up model development strategy under which model construction starts with the submodels at the leaf nodes and culminates with the submodels at the highest level.

Comparison Testing (also known as back-to-back testing) may be used when more than one version of a model representing the same system is available for testing. All versions of the model built to represent exactly the same system are run with the same input data and the model outputs are compared. Differences in the outputs reveal problems with model accuracy.

Authorization Testing is used to test how accurately and properly different levels of access authorization are implemented in the model and how properly they comply with the established rules and regulations.

Performance Testing is used to test whether (a) all performance characteristics are measured and evaluated with sufficient accuracy, and (b) all established performance requirements are satisfied.

Security Testing is used to test whether all security procedures are correctly and properly implemented in conducting a classified experiment with the model.

Standards Testing is used to substantiate that the model is developed with respect to the required standards, procedures, and guidelines.

Debugging is an iterative process the purpose of which is to uncover errors or misconceptions that cause the model's failure and to define and carry out the model changes that correct the errors. This iterative process consists of four steps. In Step 1, the model is tested revealing the existence of errors (bugs). Given the detected errors, the cause of each error is determined in Step 2. In Step 3, the model changes believed to be required for correcting the detected errors are identified. The identified model changes are carried out in Step 4. Step 1 is re-executed right after Step 4 to ensure successful modification because a change correcting an error may create another one. This iterative process continues until no errors are identified in Step 1 after sufficient testing.

Execution Monitoring is used to reveal errors by examining low-level information about activities and events that take place during model execution.

Execution Profiling is used to reveal errors by examining high-level information (profiles) about activities and events that take place during model execution.

Execution Tracing is used to reveal errors by watching the line-by-line execution of a model.

Fault/Failure Insertion Testing is used to insert a kind of fault (incorrect model component) or a kind of failure (incorrect behavior of a model component) into the model and observe whether the model produces the invalid behavior as expected. Unexplained behavior may reveal errors in model representation.

Field Testing places the model in an operational situation for the purpose of collecting as much information as possible for model validation.

Functional Testing (also known as Black-Box Testing) is used to assess the accuracy of model input–output transformation. It is applied by feeding inputs (test data) to the model and evaluating the corresponding outputs. The concern is how accurately the model transforms a given set of input data into a set of output data.

Graphical Comparisons is a subjective, inelegant and heuristic, yet quite practical approach especially useful as a preliminary approach to model VV&T. The graphs of values of model variables over time are compared with the graphs of values of system

variables to investigate characteristics such as similarities in periodicities, skewness, number and location of inflection points, logarithmic rise and linearity, phase shift, trend lines and exponential growth constants.

Data Interface Testing is conducted to assess the accuracy of data inputted into the model or outputted from the model during execution. All data interfaces are examined to substantiate that all aspects of data input/output are correct.

Model Interface Testing is used to detect (sub) model-to-(sub)model interface errors or invalid assumptions about the interfaces. This form of testing deals with how well the (sub)models are integrated with each other and is particularly useful for object-oriented and distributed models.

User Interface Testing is used to detect user-model interface errors or invalid assumptions about the interfaces. This form of testing is particularly important for testing human-in-the-loop, interactive and training models.

Object-Flow Testing is used to assess model accuracy by way of exploring the life cycle of an object during model execution.

Partition Testing is used for testing the model with the test data generated by analyzing the model's functional representatives (partitions). It is accomplished by: (1) decomposing both model specification and implementation into functional representatives (partitions), (2) comparing the elements and prescribed functionality of each partition specification with the elements and actual functionality of corresponding partition implementation, (3) deriving test data to extensively test the functional behavior of each partition, and (4) testing the model by using the generated test data.

Predictive Validation requires past input and output data of the system being modeled. The model is driven by past system input data and its forecasts are compared with the corresponding past system output data to test the predictive ability of the model.

Product Testing is conducted by the model developer after all submodels are successfully integrated and before the acceptance testing is performed by the model sponsor.

Regression Testing is used to substantiate that correcting errors and/or making changes in the model do not create other errors and adverse side-effects. It is usually accomplished by retesting the modified model with the previous test data sets used.

Sensitivity Analysis is performed by systematically changing the values of model input variables and parameters over some range of interest and observing the effect upon model behavior. Unexpected effects may reveal invalidity.

Boundary Value Testing is employed to test model accuracy by using test cases on the boundaries of the model input domain.

Equivalence Partitioning Testing partitions the model input domain into equivalence classes in such a manner that a test of a representative value from a class is assumed to be a test of all values in that class.

Extreme Input Testing is conducted by running/exercising the model by using only minimum values, only maximum values, or arbitrary mixture of minimum and maximum values for the model input variables.

Invalid Input Testing is performed by running/exercising the model under incorrect input data and cases to determine whether the model behaves as expected. Unexplained behavior may reveal model representation errors.

Real-Time Input Testing is particularly important for assessing the accuracy of models built to represent embedded real-time systems. Real-time input data collected from a real system is used for testing the model's timing relationships and correlations between input data points.

Self-Driven Input Testing is conducted by running/exercising the model under input data randomly sampled from probabilistic models representing random phenomena in a real or futuristic system.

Stress Testing is intended to test the model validity under extreme workload conditions. This is usually accomplished by increasing the congestion in the model.

Trace-Driven Input Testing is conducted by running/exercising the model under input trace data collected from a real system.

Statistical Techniques can be used to conduct model validation by comparing model and system output data obtained by running both model and system under the same input data. Some example statistical techniques for model validation include Confidence Intervals/Regions, Hotelling's T^2 Tests, Multivariate Analysis of Variance, Nonparametric Goodness-of-fit Tests, Nonparametric Tests of Means, and Time Series Analysis.

Branch Testing is conducted by executing the model under test data so as to execute as many branch alternatives as possible, as many times as possible and to substantiate their accurate operations.

Condition Testing is conducted by executing the model under test data so as to execute as many (compound) logical conditions as possible, as many times as possible and to substantiate their accurate operations.

Data Flow Testing uses the control flow graph to explore sequences of events related to the status of data structures and to examine data-flow anomalies. For example, sufficient paths can be forced to execute under test data to assure that every data element and structure is initialized prior to use or every declared data structure is used at least once in an executed path.

Loop Testing is conducted by executing the model under test data so as to execute as many loop structures as possible, as many times as possible and to substantiate their accurate operations.

Path Testing is conducted by executing the model under test data so as to execute as many control flow paths as possible, as many times as possible and to substantiate their accurate operations.

Statement Testing is conducted by executing the model under test data so as to execute as many statements as possible, as many times as possible and to substantiate their accurate operations.

Submodel/Module Testing requires a top-down model decomposition in terms of submodels/modules. The executable model is instrumented to collect data on all input and output variables of a sub-model. The system is similarly instrumented (if possible) to collect similar data. Then, each submodel behavior is compared with corresponding sub-system behavior to judge submodel validity.

Symbolic Debugging assists in model VV&T by employing a debugging tool that allows the modeler to manipulate model execution while viewing the model at the source code level.

Top-Down Testing is used in conjunction with top-down model development strategy under which model construction starts with the submodels at the highest level and culminates with the submodels at the leaf nodes

Visualization/Animation of a model greatly assists in model VV&T. Displaying graphical images of internal and external dynamic behavior of a model during execution enables one to discover errors by watching.

Formal Techniques

These techniques are based on formal mathematical proof of correctness. If attainable, formal techniques provide the most effective means of model assessment. Induction, Inference, and Logical Deduction are acts of justifying conclusions on the basis of premises given. Lamda Calculus is a system of transforming the model representation into formal expressions for which mathematical proof techniques can be applied. Predicate Calculus provides rules for manipulating predicates (combinations of simple relations), which are derived from the model representation. Predicate Transformation is used to define the model semantics with a mapping that transforms model output states to all possible model input states. This definition provides the basis for proving whether or not the model is sufficiently correct. Proof of Correctness is employed to express the model in a precise notation and then mathematically proving that: (a) the executed model terminates and (b) it satisfies the requirements of its specification.

Concluding Remarks

In modeling studies, it is well to remember the dictum that "Nobody solves *the* problem. Rather, everybody solves the model that he [or she] has constructed of the problem" (Elmaghraby 1968). This dictum clearly identifies the crucial importance of model credibility. If the model does not represent the problem with sufficient accuracy, the modeling study becomes useless. The model VV&T principles and techniques presented here indicate that assessment of model credibility is an onerous task requiring multifaceted and interdisciplinary knowledge and experience. The applicability of the techniques should be judged with respect to the model type (e.g., mathematical programming model, stochastic optimization model, simulation model). Applying VV&T techniques increases confidence in model credibility. The amount of credibility required or when to stop testing is determined with respect to the study objectives.

See

► Battle Modeling
► Model Accreditation

- ▶ Model Evaluation
- ▶ Model Management
- ▶ Practice of Operations Research and Management Science
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Structured Modeling
- ▶ Validation
- ▶ Verification

References

Balci, O. (1998a). Verification, validation and accreditation. *Proceedings of the 1998 Winter Simulation Conference, IEEE, Piscataway, New Jersey*, 41–48.

Balci, O. (1998b). Verification, validation, and testing. In J. Banks (Ed.), *The handbook of simulation* (pp. 335–393). New York: John Wiley.

Balci, O. (2010). Golden rules of verification, validation, testing, and certification of modeling and simulation applications. SCS M&S Magazine, Oct. 2010 Issue 4, The Society for Modeling and Simulation International (SCS), Vista, CA.

Balci, O., & Sargent, R. G. (1981). A methodology for cost-risk analysis in the statistical validation of simulation models. *Communications of the ACM, 24*, 190–197.

Elmaghraby, S. E. (1968). The role of modeling in I.E. Design. *Industrial Engineering, 19*, 292–305.

VERT

Venture evaluation and review technique. A network simulation technique design for systematic assessment of the risks involved in undertaking a new venture and in resource planning, control monitoring and overall evaluation of ongoing projects, programs and systems.

See

- ▶ Network Planning
- ▶ Project Management
- ▶ Research and Development

Vertex

- ▶ Extreme Point
- ▶ Node

Virtual Reality

An extension of the simulator concept in which the computer-simulated external physical world is dynamic rather than static. Actions by the user(s) may change the simulated external world in the same way those actions would affect the real world through the use of the real equipment. The primary sensory environment of virtual reality systems is visual, but sound is prevalent in many systems, with touch/feel also found in some systems. Such systems are routinely used for test pilots and military combat training. Another major application is computer gaming.

See

- ▶ Battle Modeling

References

Sherman, W. R., & Craig, A. B. (2002). *Understanding virtual reality: Interface, application, and design*. San Francisco: Morgan Kaufmann.

Visualization

Peter C. Bell
University of Western Ontario, London, Ontario, Canada

Introduction

Operations research/management science (OR/MS) is frequently involved in the development and use of visualization techniques at various stages of the problem-solving cycle. Jones (1994, 1996) provides many examples, including the use of natural language and informal diagrams at the problem conceptualization stage; spreadsheets, and block structured languages at the problem formulation stage; spreadsheets and relational databases during data collection; interactive optimization, and network flow graphics during problem solution; objective plots and matrix images at the solution analysis stage; and

animation, hypertext, hypermedia, and presentation graphics for results presentation.

The use of visualization techniques such as these is not new: many of the earliest examples of OR/MS problem solving made use of visual concepts. Graphs were routinely used to summarize the results of modeling studies, flowcharts were used to sketch out the flow of an algorithm, and graphical techniques have long been a mainstay of teaching about the simplex method. However, visualization has moved from a position at the periphery of OR/MS to being an important driver of new developments in the field.

The emergence of visualization into the centre stage of OR/MS parallels developments in computing, which have seen the industry's early emphases on number-crunching speed and storage capacity superseded by those of user-friendliness and marketability. These developments have led to spreadsheet software with a ubiquitous visual presentation based on rows and columns, the WIMP (Windows/Icons/Mouse/Pull-down-menu) user interface, and a large variety of user-friendly software for the production of colorful dynamic computer-generated pictures. Just as these developments have proved marketable for the computer industry, so have they proved marketable within OR/MS, with the consequence that OR/MS software that produces vivid visualizations is now widely available.

Sophisticated computer-generated graphics represent the most elaborate extreme of the spectrum of visualization possibilities. Many other methods are employed that use many different spatial techniques to add information content to data. At the opposite end of this spectrum are some very simple tools where the picture elements are characters spatially arranged to have limited visual characteristics. For example, text, the arrangement of data in a table, a set of corporate accounts, and the block structure in a computer code all use a simple visual layout to improve understanding of the numbers and characters. Between these two extremes are a host of tools that have traditionally been characterized in two main groups: presentation graphics and iconic graphics.

Presentation Graphics

Presentation graphics are pictures (bar charts, line graphs, or pie charts) that are used to illustrate or summarize data. The use of these types of tools predates the computer era. From the earliest days of OR/MS, presentation graphics have been used to summarize data, to illustrate the results of OR/MS work, and to aid in communicating data or results to decision makers or management. Considerable research has been done that addresses issues such as when a graph is more useful than numbers, what type of graph is most useful in which situation, and when the use of color adds value to a presentation graphic (Desanctis 1984). The results of this body of research suggest that the nature of the task is very important in determining the appropriateness of numerical display or various presentation graphic forms (Vessey 1991).

Iconic Graphics

Iconic graphics include picture elements that map to elements of the real world. A road map is an iconic graphic consisting of lines that are icons, representing roads, and blocks that represent urban areas. Other common iconic graphics include floor plans, PERT charts, and network flow diagrams. Again, iconic graphics have a long history within OR/MS, but research on the value of iconic formats is lacking. For many people, the value has been obvious: try driving from New York to San Francisco using only numeric data on road and town locations! Often, however, there are alternative iconic representations for a problem, but research has been slow to provide answers to resolve these choices. As a consequence, the market has been the determining factor in deciding which iconic formats survive and which die, with the result that the survivors are often high on color and razzle-dazzle but perhaps not the most useful.

Iconic graphics can be categorized as static or dynamic. An important application area for static iconic graphics has been transportation systems routing and planning. Models that link mathematical programming models to computer-generated road or street maps have been used to solve truck routing and scheduling problems, mass transit system route planning and scheduling problems, and school bus routing problems (Florian et al. 1987; Bodin and Levy 1994).

Dynamic iconic graphics, or animations, were first applied to the study of operations problems by Hurrion (1980) and have proved to be a huge market success.

A major application area for animation is simulation modeling, where animation is now routinely used to illustrate the progress of a simulation code. The use of animation seems to aid code debugging, model verification and model validation (Sargent 2011), and the presentation of the results of simulation studies to decision makers. Visual interactive simulation couples animation with interactive access to the running simulation model to produce decision support systems with visual user interfaces that provide useful tools to aid problem formulation and interactive problem solution (Bell 1991). The use of animation and interaction with simulation models is now so pervasive that every major simulation modeling software package includes these capabilities.

Animated sensitivity analysis uses dynamic graphics to illustrate the sensitivity of an optimal solution to changes in a parameter (Jones 1992). As the parameter is changed, a visual screen display is updated 30 times/second to illustrate the response of the optimal solution to the change.

Impact of New Technologies

The traditional view of visualization has been considerably expanded by new technologies. Text is a graphic format (the location of the characters has meaning), as is hypertext. These tools provide a host of visual formats, including choice of font, size, and layout. Both text and hypertext are used as a front-end for OR/MS models. Again, there exists a broad spectrum of possibilities from simple examples, such as the use of textual data on punched cards as input to mathematical programming software, to hypertext systems that provide the ability to navigate through a complex optimization problem (Kimbrough et al. 1990).

The emergence of multimedia and virtual reality development tools at reasonable cost has driven new developments within OR/MS. As these technologies have become more commonplace, there have appeared many new kinds of OR/MS models that take advantage of the new delivery systems available for OR/MS work (Lembersky and Chi 1984).

While the emergence of visualization as an important field within OR/MS appears to have been market driven, a body of research evidence has appeared which supports a view that visualization helps decision makers solve problems. Surveys of model builders (Kirkpatrick and Bell 1989) and of decision makers who have used visual and interactive models (Bell et al. 1995) strongly support a view that model developers and decision makers believe that these types of tools lead to improved decision making, and explain the market success of software that provides animation capability for simulation models. Task-based behavioral research comparing dynamic iconic graphic tools with non-visual tools has demonstrated the superiority of the graphic tools for some specific tasks (Bell and O'Keefe 1995; Chau and Bell 1995).

Finally, there is a growing body of evidence that suggests that the use of visualization and interaction in conjunction with OR/MS models and new information technology tools will have a revolutionary effect on OR/MS. These tools facilitate, or may even require, the use of innovative problem-solving methodologies (Bell and O'Keefe 1994), and the development of areas of new theory and new algorithms to support these methodologies (Bell 1994). Jones (1994, 1996) are recommended for further reading.

See

- ▶ Computational Geometry
- ▶ Computer Science and Operations Research Interfaces
- ▶ Scheduling and Sequencing
- ▶ Simulation of Stochastic Discrete-Event Systems
- ▶ Vehicle Routing

References

Bell, P. C. (1991). Visual interactive modelling: The past, the present, and the prospects. *European Journal of Operational Research, 54*, 274–286.

Bell, P. C. (1994). Visualization and optimization: The future lies together. *ORSA Journal on Computing, 6*, 258–260.

Bell, P. C., Elder, M., & Staples, S. (1995). Decision makers' perceptions of the value and impact of visual interactive models, (Technical Report). Ivey School of Business, University of Western Ontario, London, Ontario.

Bell, P. C. & O'Keefe, R. (1994). Visual interactive simulation: A methodological perspective. *Annals of Operations Research*, 53, Volume on Simulation and Modeling, O. Balci, (Ed.)., 321–342.

Bell, P. C., & O'Keefe, R. (1995). An experimental investigation into the efficacy of visual interactive simulation. *Management Science, 41*, 1018–1038.

Bodin, L., & Levy, L. (1994). Visualization in vehicle routing and scheduling problems. *ORSA Journal on Computing, 6*, 261–269.
Chau, P., & Bell, P. C. (1995). Designing effective simulation-based decision support systems: An empirical assessment of three types of decision support system. *Journal of the Operational Research Society, 46*, 315–331.
Desanctis, G. (1984). Computer graphics as decision aids: Directions for research. *Decision Sciences, 15*, 463–487.
Florian, M., Crainic, T., & Guelat, J. (1987). FRET — An interactive graphic method for strategic planning of freight flows, presented at the IFORS '87 Conference, Buenos Aires.
Hurrion, R. D. (1980). An interactive visual simulation system for industrial management. *European Journal of Operational Research, 5*, 86–93.
Jones, C. V. (1992). Animated sensitivity analysis. In O. Balci, R. Sharda, & S. A. Zenios (Eds.), *Computer science and operations research: New developments in their interface* (pp. 177–196). Oxford, UK: Pergamon Press.
Jones, C. V. (1994). Visualization and optimization. *ORSA Journal on Computing, 6*, 221–257.
Jones, C. V. (1996). *Visualization and optimization*. New York: Springer.
Kimbrough, S. O., Pritchett, C. W., Bieber, M. P., & Bhargava, H. K. (1990). The coast guard's KSS project. *Interfaces, 20*, 5–16.
Kirkpatrick, P., & Bell, P. C. (1989). Visual interactive modelling in industry: Results from a survey of visual interactive model builders. *Interfaces, 19*(5), 71–79.
Lembersky, M. R., & Chi, U. H. (1984). Decision simulators speed implementation and improve operations. *Interfaces, 14*(4), 1–15.
Sargent, R. G. (2011). Verification and validation of simulation models. In S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, & M. Fu, (Eds.), *Proceedings of the 2011 winter simulation conference* (pp. 183–198).
Vessey, I. (1991). Cognitive fit: A theory-based analysis of the graphics versus table literature. *Decision Sciences, 22*, 219–241.

Vogel's Approximation Method (VAM)

A method for finding a first feasible solution to a transportation problem. The procedure begins by finding the two lowest cost cells for each row and column in the transportation problem array. Subtracting the smaller of these costs from the other produces a Vogel number for each row and column. Select the largest Vogel number and make the first assignment to the corresponding lowest cost cell, where the assignment is the maximum amount that can be sent from the corresponding origin to the corresponding destination. After each assignment, the Vogel numbers are recomputed based on the remaining rows and columns in the array. The procedure is repeated until all assignments (shipments) are made. Although VAM tends to find a good (low cost) first feasible solution, the extra computational work required has proven to be a detriment to its use in computer-based software for solving transportation problems.

See

- ▶ Northwest-Corner Solution
- ▶ Transportation Simplex (Primal-Dual) Method

References

Reinfeld, N., & Vogel, W. (1958). *Mathematical Programming*. New Jersey: Prentice-Hall, Englewood Cliffs.

Von Neumann-Morgenstern (Expected) Utility Theory

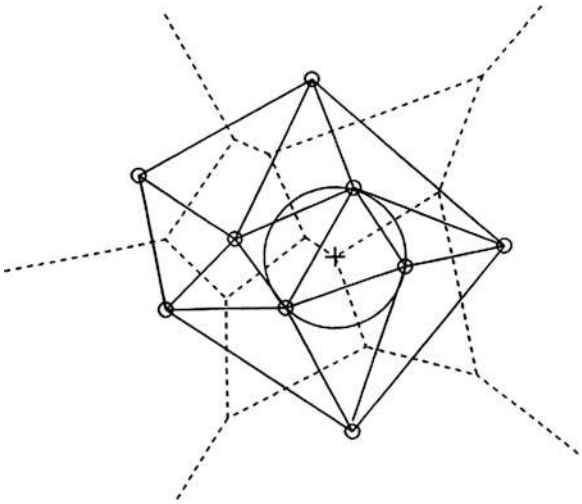
- ▶ Decision Analysis
- ▶ Preference Theory
- ▶ Utility Theory

Voronoi Constructs

Isabel M. Beichl[1], Javier Bernal[1], Christoph Witzgall[1] and Francis Sullivan[2]
[1]National Institute of Standards & Technology, Gaithersburg, MD, USA
[2]Supercomputing Research Center, Bowie, MD, USA

Introduction

Given a finite set S of "sites" p_i located in Euclidean space $\mathfrak{R}^d$, the Voronoi polyhedron $V(p_j)$ of site p_j is the set of all points $p \in \mathfrak{R}^d$ which are at least as close to site p_j as to any other site p_i. Such a Voronoi polyhedron (also called Thiessen polygon or Wigner-Seitz cell) is convex, its facets determined by perpendicular bisectors — (hyper)planes or lines of



Voronoi Constructs, Fig. 1 Planar Voronoi diagram (dashed lines) and Delaunay triangulation of nine sites. The circle around one of the Delaunay triangles illustrates the empty circle criterion

equal Euclidean distance from two distinct sites. The Voronoi polyhedra $V(p_i)$, $p_i \in S$ cover the space $\mathfrak{R}^d$ and define a polyhedral cell-complex known as a Voronoi diagram (Voronoi 1908) or Dirichlet tesselation (Dirichlet 1850). For a survey, consult Aurenhammer (1991); also see the texts by Okabe, Boots, and Sugihara (1992); Goodman and O'Rourke (2004).

The cells of the dual complex are convex and, in general, simplicial. By partitioning nonsimplicial cells of the dual complex into simplices, the Delaunay triangulation results (Fig. 1). It provides a canonical scheme for triangulating the convex hull of an arbitrary set $S \subset \mathfrak{R}^d$ of sites, with these sites as vertices. Under the assumption that sites are realizations of a homogeneous Poisson process, statistics for geometrical parameters of Voronoi diagrams and Delaunay triangulations have been derived (Miles 1970; Stoyan et al. 1987).

Delaunay Triangulation

For each site $p_i \in S$, the Delaunay triangulation contains an edge from p_i to each of its nearest Euclidean neighbors $q \in S$. In particular, edges in that triangulation connect all pairs of points of minimum distance in S. The 1-skeleton of the Delaunay triangulation contains the relative neighborhood graph, which in turn contains a Euclidean minimum spanning tree. The Delaunay triangulation thus provides a convenient tool for solving various proximity problems (Shamos and Hoey 1975). Delaunay triangulations avoid narrow triangles (see below) as much as possible, are essentially unique, and are readily determined. They are often the triangulations of choice for constructing piecewise-linear surfaces and for applications of finite-element techniques in engineering.

Delaunay triangulations are characterized by the empty sphere criterion: the circumsphere of a simplex in a Delaunay triangulation does not contain any of the triangulation vertices in its interior (Delaunay 1934). This criterion determines a triangulation uniquely in the absence of degeneracy, i.e., the occurrence of several simplices sharing a circumsphere.

In two dimensions, the empty circle criterion is equivalent to the requirement that the ascending sequence of angles, formed by selecting a smallest interior angle from each triangle in the triangulation, lexicographically maximizes the corresponding sequences for all triangulations of the same vertex set (equiangularity). The requirement that the sequence of *all* interior angles be lexicographically maximum is, in the presence of degeneracy, stronger, and can therefore serve in some instances as a tie-breaker in the presence of degeneracy.

The Delaunay triangulation of a set $S \subset \mathfrak{R}^d$ of n sites can be obtained as a projection of the face lattice of the convex hull of n suitable points in $\mathfrak{R}^{d+1}$. Those points can be chosen on a sphere — stereographic projection — or on a rotational paraboloid whose axis is perpendicular to the space of the triangulation. This implies that the Voronoi/Delaunay problem in d dimensions is computationally subsumed under the strong formulation of the convex hull problem in $d + 1$ dimensions.

To check whether a given triangulation satisfies the empty sphere criterion, it is not necessary to verify that criterion for each simplex by scanning all sites which are not vertices of the simplex: only pairs of facet-adjacent simplices whose union is convex need to be examined as to whether anyone of the two vertices not in the common facet might lie in the interior of its opposite circumsphere. This corresponds to establishing convexity of a (hyper) surface by examining the angles at which adjacent facets are joined. In two dimensions, the above

criterion reduces to checking each strictly convex quadrangle formed by edge-adjacent triangles as to whether the correct diagonal of the quadrangle belongs to the triangulation (Lawson 1977). Based on this observation, several simple and efficient methods such as the insertion method swap diagonals in quadrangles. Alternatively, divide-and-conquer as well as plane sweep techniques yield $O(n \log n)$ algorithms for planar Delaunay triangulation of n sites. Determination of Voronoi diagrams in linear expected time is discussed in Bentley, Weide, and Yao (1980), and Dwyer (1991).

In many applications, it is desirable to construct a planar triangulation with some prescribed edges while preserving the advantages — avoiding unnecessarily narrow triangles, essential uniqueness — of the Delaunay approach. In that case, the empty circle criterion can be generalized by testing for potential inclusion only those sites whose "visibility" from any point of the triangle is not blocked by a prescribed edge. This generalized empty circle criterion defines a constrained Delaunay triangulation, which is unique except for sites on the peripheries of empty circles (De Floriani and Puppo 1988).

A second important generalization of the Voronoi diagram is the power diagram (see Aurenhammer 1987) or radical Voronoi diagram (Gellatly and Finney 1982). Here sites may be enlarged to spheres of positive radius. The intersection, real or imaginary, of two spheres lies on and defines the "radical" (hyper) plane of that pair. These (hyper)planes then play the same role as the perpendicular bisectors in the classical Voronoi diagram. The radical Voronoi diagram of site spheres of radius $r_i \geq 0$ centered at locations $p_i \in \Re^d$ respectively, can be obtained by intersecting the classical Voronoi diagram for the sites (p_i, r_i) in $d + 1$ dimensions with the original d-dimensional space. Radical Voronoi diagrams are used in crystallography in order to account for differences in atomic radii.

There are numerous other generalizations of the Voronoi/Delaunay construct. Alternatives to the Euclidean norm, as well as general sets instead of single point sites, are considered. There are order-k, furthest site, weighted, discrete, and abstract Voronoi diagrams. Voronoi constructs based on the Euclidean metric are instances of cell-complexes derived from arrangements of hyperplanes. Data structures, algorithms and combinatorial results concerning such cell-complexes in general are presented by Edelsbrunner, O'Rourke, and Seidel (1986) and in the text by Agarwal (1991).

See

- ▶ Computational Geometry
- ▶ Graph Theory
- ▶ Minimum Spanning Tree Problem

References

Agarwal, P. K. (1991). *Intersection and decomposition algorithms for planar arrangements*. New York: Cambridge University Press.

Aurenhammer, F. (1987). Power diagrams: Properties, algorithms, and applications. *SIAM Journal on Computing, 16*, 78–96.

Aurenhammer, F. (1991). Voronoi diagrams — A survey of a fundamental geometric data structure. *ACM Computing Surveys, 23*, 345–405.

Bentley, J. L., Weide, B. W., & Yao, A. C. (1980). Optimal expected-time algorithms for closest point problems. *ACM Transactions on Mathematical Software, 6*, 563–580.

De Floriani, L., & Puppo, E. (1988). *Constrained delaunay triangulation for multiresolution surface description*. 9th International conference on pattern recognition, Rome, Italy, 1, 566–569.

Delaunay, B. (1934). Sur la sphère vide. *Bulletin of Academic Sciences USSR* VII: Class. Sci. Mat. Nat., 793–800.

Dirichlet, G. L. (1850). Uber die Reduction der positiven quadratischen Formen mit drei unbestimmten ganzen Zahlen. *Journal Reine Angew Mathematics, 40*, 209–227.

Dwyer, R. A. (1991). Higher-dimensional Voronoi diagrams in linear expected time. *Discrete & Computational Geometry, 6*, 343–367.

Edelsbrunner, H., O'Rourke, J., & Seidel, R. (1986). Constructing arrangements of lines and hyperplanes with applications. *SIAM Journal on Computing, 15*, 341–363.

Gellatly, B. J., & Finney, J. L. (1982). Characterizations of models of multicomponent amorphous metals: The radical alternative to the Voronoi polyhedron. *Journal of Non-Crystalline Solids, 50*, 313–329.

Goodman, J. E., & O'Rourke, J. (2004). Handbook of discrete and computational geometry, 2nd edition, Boca Raton, FL: CRC Press.

Lawson, C. L. (1977). Software for C^1 surface interpolation. In J. R. Rice (Ed.), *Mathematical software III*. New York: Academic Press.

Miles, R. E. (1970). On the homogeneous planar Poisson process. *Mathematical Biosciences, 6*, 85–127.

Okabe, A., Boots, B., & Sugihara, K. (1992). *Spatial tesselations: Concepts and applications of Voronoi diagrams*. New York: Wiley.

Shamos, M. I., & Hoey, D. (1975). Closest-point problems. *Proceedings of 16th Annual IEEE Symposium on Foundation of Computer Science*, pp. 151–162.

Stoyan, D., Kendall, W. S., & Mecke, J. (1987). *Stochastic geometry and its applications*. New York: Wiley.

Toussaint, G. T. (1980). The relative neighborhood graph of a finite planar set. *Pattern Recognition, 12*, 261–268.
Voronoi, M. G. (1908). Nouvelles applications des paramètres continus à la théorie des formes quadratiques. *Journal Reine Angew Mathematics, 134*, 198–287.

Voronoi Diagram

- ► Computational Geometry
- ► Voronoi Constructs

VV&A

Verification, validation, and accreditation.

See

- ► Battle Modeling
- ► Model Accreditation
- ► Model Evaluation
- ► Model Management
- ► Validation
- ► Verification
- ► Verification, Validation, and Testing of Models

VV&T

- ► Verification, Validation, and Testing of Models

W

Waiting Time

In a single queue, the time from customer entrance into the queue until completion of service; in a queueing network, the total elapsed time between customer arrival to the network and final departure from the network. Sometimes, however, waiting time refers only to the time from arrival until the *beginning* of service. These two different quantities are often differentiated by referring to the former as the system or sojourn time and the latter as the queueing time or delay in queue.

See

► Queueing Theory

War Game

A model whose object is military combat or some aspect of combat. "War game" is used to emphasize the competitive nature of the model, either through human interaction on one or more sides of the combat or automated, game-theoretic competition or computer simulation.

See

► Battle Modeling

Warehouse Problem

A warehouse has a fixed capacity C and an initial stock s_0 of a certain product that is subject to known seasonal fluctuations in selling price and cost. The problem is to determine the optimal pattern of purchases, storage, and sales for the next n months. The problem can be formulated as a linear-programming problem. Its dual has an interesting form that enables the dual solution to be determined readily.

Water Resources

Roman Krzysztofowicz
University of Virginia, Charlottesville, VA, USA

Introduction

Methodologies and techniques of operations research and management science have been applied to a vast array of water resource problems since the early 1960s. Conversely, water resource problems stimulated several methodological developments, notably in statistics of extremes, dynamic programming algorithms, and multi-objective optimization methods. Four classes of problems are discussed herein, techniques that have been employed are noted, and exemplary models are sketched. Fundamental to building operational models is the science of water transport processes.

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Hydrology and Hydraulics

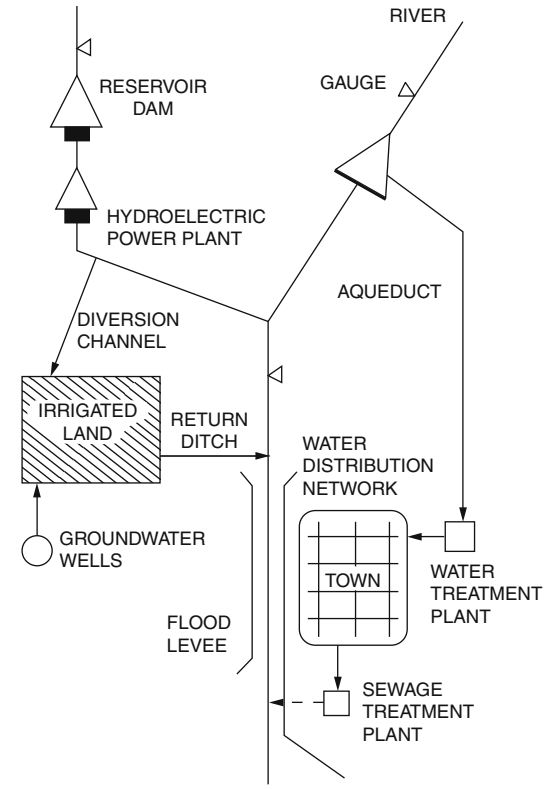
Variability of the quantity and quality of natural waters and their renewability in space and time are governed by the water cycle – a sequence of processes through which water is transported between the atmosphere, the land, and the ocean: precipitation, evaporation, transpiration, infiltration, groundwater flow, and river flow. Hydrology develops models of these natural processes; the models can be used to describe or predict the quantity and quality of water available in a given place and time (Musy and Higy 2010). This information, in turn, constitutes an input into models of water resource systems. Hydraulics develops models of flow in channels and lakes, and through constructed facilities such as spillways, sluices, fish ladders, turbines, pumps, pipelines, aqueducts, culverts, navigation locks, and floodways (Mays 2005). These models serve as building blocks of control and management models.

Planning Water Resource Development

The purpose of water resource development is to alter the natural water cycle so as to ensure the quantity and quality of water in places and times dictated by socioeconomic objectives of human activities. Specific purposes are: (i) flood control, (ii) hydroelectric power generation, (iii) water supply for domestic, municipal, industrial and agricultural uses, and (iv) low-flow augmentation for navigation, recreation, water quality control (by diluting wastewater and contaminated runoff) and aquatic life maintenance (by increasing volume and decreasing temperature of rivers during summer).

For comprehensive planning, the natural boundary of a water resource system is a river basin (source of surface water) and the underlying aquifer (source of groundwater). Figure 1 depicts an exemplary system. Planning involves tasks such as deciding the type, location and size of facilities, sequencing investments, and developing control policies; facilities may be operated individually or conjunctively (as two reservoirs in a cascade, or wells and reservoirs supplying irrigation water to the same district).

The planning process begins with identification of a time horizon (usually several decades) and objectives



Water Resources, Fig. 1 Water resources

(usually multiple ones). Next, the available water resources are characterized: groundwater supply and the rate of its recharge are estimated; river flows at gauge sites are modeled, for example, as time series (Hipel 1985); extreme events such as floods (Krzysztofowicz 2002) and droughts (SHH Special Issue 1991) are modeled stochastically. Predictive models of water demands for various purposes are developed. Alternative system plans are designed, and operations of individual projects or subsystems are described via simulation models or optimization models (in the form of integer, linear, nonlinear, chance-constrained, stochastic, or dynamic programming). Finally, all models are synthesized into a comprehensive river basin planning model, which provides a decision support for a multi-objective analysis (WRB Special Issue 1992). The purpose of the analysis is to screen a large number of alternative plans and to select a few which are Pareto-optimal. The choice of a plan for implementation is usually left to the political process (Loucks and van Beek 2005).

Operation of Hydrosystems

One of the most active and challenging research areas has been optimal control of reservoirs, aqueducts, irrigation systems, water distribution networks, urban drainage and sewage systems. Control policies are almost always discrete-time (with the time interval of an hour, day, week, month or year), but otherwise they may be discrete- or continuous-state, finite- or infinite-horizon, stationary or nonstationary (e.g., periodic as the annual regime of river flows).

In a generic single reservoir control problem, the state x_n denotes the storage at the beginning of time interval n, the input ω_n represents the inflow during interval n, the control u_n is the release decided at the beginning of interval n, and the output y_n represents the outflow during interval n. With any finite ($n = 1, \ldots, N$) trajectories $\boldsymbol{x} = \{x_n\}$ and $\boldsymbol{y} = \{y_n\}$, there is associated a performance measure $g(\boldsymbol{x}, \boldsymbol{y})$, whose form is dictated by reservoir purposes (e.g., generated hydropower, prevented flood damages). In a deterministic case, inflows $\{\omega_n\}$ are assumed to be known; hence $y_n = u_n$, and one wishes to find a policy $\boldsymbol{u}^* = \{u_n^*\}$ maximizing $g(\boldsymbol{x}, \boldsymbol{y})$ subject to the state dynamics, $x_{n+1} = x_n - u_n + \omega_n$, and constraints on storage and release. In a stochastic case, inflows follow a probabilistic law (usually of Markovian structure), and one wishes to find a strategy $\mu^* = \{u_n^*\}$, a sequence of control rules, $u_n = \mu_n^*(x_n)$, that maximizes the expectation $E[g(\boldsymbol{X}, \boldsymbol{Y})]$, subject to nonlinear state dynamics, output operators, and possibly probabilistic constraints.

The complexity of hydrosystems is reflected in the many control models described in the literature. They can be classified according to these features: (i) single reservoir vs. multi-reservoir, (ii) single purpose vs. multi-purpose, (iii) deterministic inflows vs. stochastic inflows, (iv) climatic statistics vs. hydrologic forecasts, (v) linear objective functions vs. non-linear objective functions, (vi) separable objective functions vs. non-separable objective functions, (vii) single objective control vs. multi-objective control, (viii) short-term control (hourly, daily, weekly) vs. long-term control (monthly, yearly), (ix) one-level control vs. hierarchical control, (x) terminal condition vs. infinite horizon.

Deterministic control problems are often formulated as dynamic programs (DP) solved via discrete DP, successive approximation algorithms such as state incremental DP and differential DP (Yakowitz 1982), or approximating linear-quadratic controllers (Protopapas and Georgakakos 1990). Among other techniques one finds linear programming (Yeh et al. 1980), and its chance-constrained variations, network flow algorithms, both linear and nonlinear (Rosenthal 1981), and multi-objective optimization. Stochastic control problems are almost exclusively formulated as dynamic programs solved via discrete DP, policy iteration methods, or approximating linear-quadratic controllers. Various quasi-stochastic approaches have also been tried, such as sampling DP, simulation methods, combined simulation-optimization methods, and heuristic control strategies (Faber and Stedinger 2001). Despite these advances, stochastic control of hydrosystems remains at the forefront of research–the challenges stemming from the dimensionality of the state space, spatial and temporal dependence of hydrologic inputs, nonlinear state dynamics, nonlinear and multiple objective functions.

Mitigation of Floods

Structural solutions, such as dams, diversion channels with retention basins, and levees, offer protection against floods up to a certain magnitude. Risk and benefit-cost analyses have guided decisions concerning the degree of protection and size of structures. Heuristic rules, simulation, and optimization methods have been employed to develop strategies for operation of reservoirs during floods.

Nonstructural solutions, such as floodplain zoning, flood insurance, and flood warning systems, aim at reducing the negative consequences of floods. Risk and decision analyses have been proposed for delineating land use zones, setting insurance rates, issuing flood warnings, and evaluating economic benefits of flood forecasts (Krzysztofowicz and Davis 1984).

A decision-theoretic model of a flood warning system provides an example (Krzysztofowicz 1993). Having received forecast (s, t) of (H, Λ), the uncertain flood crest H and time to crest Λ at a river gauge, a manager must decide whether to issue ($w = 1$) or not to issue ($w = 0$) a warning for a zone of the floodplain above elevation y. Thereafter the zone is

flooded ($\theta = 1$) or not ($\theta = 0$). Each decision-event vector (w, θ) leads to disutility

$$D_{w\theta}(s,t) = \int_y^\infty \int_0^\infty d_{w\theta}(h,\lambda)\,\phi\,(h,\lambda|s,t)d\lambda dh,$$

where ϕ ($\cdot$, $\cdot|s$, t) is the posterior density of (H, Λ), conditional on the forecast, and $d_{w\theta}$ (h, λ) is the disutility of all economic, social, and behavioral outcomes resulting from flood crest h occurring at time λ. The expected disutility associated with decision w, termed the risk function, is

$$R(s,t,w) = D_{w0}(s,t)Pr\{\theta = 0|s,t\} + D_{w1}(s,t)Pr\{\theta = 1|s,t\}.$$

For each (s, t), the optimal warning rule W^* prescribes decision $w = W^*(s,t)$ which minimizes the risk $R(s,t,w)$.

Management of Water Quality

Water pollution comes from either point sources, which can be directly monitored (e.g., industrial wastewater discharges), or nonpoint sources, from which loadings can only be estimated (e.g., contaminated runoff from agricultural fields and urban areas). The preference of downstream users for clean water and the preference of upstream entities (such as municipalities, industries, and agricultural producers) for free discharging of contaminants create a societal conflict whose resolution requires legislative, economic, and institutional means.

Management models are typically formulated in support of planning by a regional authority faced with decisions such as locating and sizing waste-water treatment plants and effluent disposal fields, setting charges for release of wastewater, locating and operating monitoring networks, and devising enforcement policies. These decision problems are multi-objective and hierarchical in nature (Loucks and van Beek 2005). At the upper level, the authority's objectives are (i) to minimize the total cost, (ii) to equitably allocate the cost to entities, and (iii) to improve the quality of waste-receiving waters. At the lower level, an entity's objectives are (i) to minimize its cost and (ii) to optimize its compliance with effluent standards and discharge regulations. Game-theoretic models are developed to predict the compliance behavior of entities, and thus the effectiveness of policies (WRB Special Issue 1992). Water quality models – simulating the physical, chemical, and biological processes taking place in water bodies – are employed to predict impacts of alternative management plans on concentration of constituents (e.g., biochemical oxygen demand, dissolved oxygen deficit, nitrogen, phosphorus, metals, organics, bacteria), which collectively define water quality (Young 1993).

See

- Dynamic Programming
- Environmental Systems Analysis
- Game Theory
- Global Models
- Linear Programming
- Multiobjective Programming
- Nonlinear Programming
- Stochastic Programming

References

Faber, B. A., & Stedinger, J. R. (2001). Reservoir optimization using sampling SDP with ensemble streamflow prediction (ESP) forecasts. *Journal of Hydrology, 249*, 113–133.

Hipel, K. W. (Ed.). (1985). *Time series analysis in water resources*. Bethesda, MD: American Water Resources Association.

Krzysztofowicz, R. (1993). A theory of flood warning systems. *Water Resources Research, 29*, 3981–3994.

Krzysztofowicz, R. (2002). Probabilistic flood forecast: Bounds and approximations. *Journal of Hydrology, 268*, 41–55.

Krzysztofowicz, R., & Davis, D. R. (1984). Toward improving flood forecast-response systems. *Interfaces, 14*(3), 1–14.

Loucks, D. P., & van Beek, E. (2005). *Water resources systems planning and management: An introduction to methods, models and applications*. Paris: United Nations Educational, Scientific and Cultural Organization.

Mays, L. W. (Ed.). (2005). *Water resources systems management tools*. New York: McGraw-Hill.

Musy, A., & Higy, C. (2010). *Hydrology: A science of nature*. Enfield, New Hampshire: Science Publishers.

Protopapas, A. L., & Georgakakos, A. P. (1990). An optimal control method for real-time irrigation scheduling. *Water Resources Research, 26*, 647–669.

Rosenthal, R. E. (1981). A nonlinear network flow algorithm for maximization of benefits in a hydroelectric power system. *Operations Research, 29*, 763–786.
SHH Special Issue. (1991). Drought analysis. *Stochastic Hydrology and Hydraulics, 5*, 253–322.
WRB Special Issue. (1992). Multiple objective decision making in water resources. *Water Resources Bulletin, 28*, 1–231.
Yakowitz, S. (1982). Dynamic programming applications in water resources. *Water Resources Research, 18*, 673–696.
Yeh, W. W.-G., Becker, L., Toy, D., & Graves, A. L. (1980). Central Arizona project: Operations model. *Journal of Water Resources Planning and Management, 106*, 521–540.
Young, P. C. (Ed.). (1993). *Concise encyclopedia of environmental systems*. New York: Pergamon Press.

Weak Derivatives

A method used in stochastic simulation for deriving unbiased gradient estimators of outputs with respect to input parameters, usually in probability distributions; also known as measure-valued differentiation.

See

- ▶ Perturbation Analysis
- ▶ Score Functions
- ▶ Simulation of Stochastic Discrete-Event Systems

References

Fu, M. C. (2008). What you should know about simulation and derivatives. *Naval Research Logistics, 55*, 723–736.
Pflug, G. C. (1989). Sampling derivatives of probabilities. *Computing, 42*, 315–328.

Weak Duality Theorem

- ▶ Strong Duality Theorem

Weakly-Coupled Systems

A linear-programming problem that has a few variables that connect (couple) the constraints or subsets of constraints. Such systems usually arise in time dimensioned large-scale problems that exhibit a block-angular structure. The dual of such systems are weakly-coupled in the sense of having a few constraints that tie the blocks together. Special adaptations of the simplex method exist that take advantage of such structures in their computations.

See

- ▶ Dualplex Method
- ▶ Large-Scale Systems
- ▶ Rosen's Partitioning Method

Weber Problem

- ▶ Location Analysis

Wicked Problems

As first described by Professor Horst Rittel, University of California Architecture Department, the term wicked problem refers "to that class of social system problems which are ill-formulated, where the information is confusing, where there are many clients and decision makers with conflicting values, and where the ramifications in the whole system are thoroughly confusing" (Churchman 1967, B141). Such problems, however, are not restricted just to social system problems; they are encountered especially in systemic problems in many areas of business, industry, and government.

In their paper, "Dilemmas in a General Theory of Planning," Rittel and Weber (1973, pp. 161–166) point out that there are ten distinguishing properties of wicked problems:

1. There is no definitive formulation of a wicked problem.
2. Wicked problems have no stopping rule.
3. Solutions to wicked problems are not true or false but good or bad.
4. There is no immediate and no ultimate test of a solution to a wicked problem.
5. Every solution to a wicked problem is a "one-shot operation," because there is no opportunity to

learn by trial and error, every attempt counts significantly.
6. Wicked problems do not have an enumerable (or an exhaustively describable) set of potential solutions, nor is there a well-described set of permissible operations that may be incorporated into the plan.
7. Every wicked problem is essentially unique.
8. Every wicked problem can be considered to be a symptom of another problem.
9. The existence of a discrepancy representing a wicked problem can be explained in numerous ways. The choice of explanation determines the nature of the problem's resolution.
10. The planner has no right to be wrong.

See

► Soft Systems Methodology

References

Churchman, C. W. (1967). Wicked problems. *Management Science, 14*(4), B141–B142.
Rittel, H., & Webber, M. (1973). Dilemmas in a general theory of planning. *Policy Science, 4*, 155–169.

Wilkinson Equivalent Random Technique

An approximation for the blocking probability that an overflow stream sees in an Erlang loss system. The method is primarily used to analyze congestion in telecommunication networks.

See

► Queueing Theory

References

Wilkinson, R. I. (1956). Theories for toll traffic engineering in the U.S.A. *Bell System Technical Journal, 35*, 421–514.

WIMP

Windows/Icons/Menus/Pointers or Windows/Icons/Mouse/Pull-down-menu. Style of graphical user interface (GUI) first popularized in the Apple Macintosh personal computers.

See

► GUI
► Visualization

References

Stadler, A. (2009). Graphical user interfaces. In B. W. Wah (Ed.), *Wiley Encyclopedia of Computer Science and Engineering* (pp. 1464–1476).

Winner's Curse

The selection bias that occurs in an auction or other situation in which bidders with independent estimates of the value of an item compete to buy it. Even though all of the competitors' estimates are unbiased, the winner will have tended to overestimate the value. Also used, less precisely, to denote an expected loss by a winning bidder.

See

► Bidding Models

Wolfe's Quadratic-Programming Problem Algorithm

An adaptation of the simplex method that solves quadratic-programming problems with positive definite or positive semidefinite quadratic forms. It is based on the simultaneous solution of the linear constraints of the problems and associated Karush-Kuhn-Tucker conditions. It uses a restricted basis entry for the solution of necessary complementarity conditions.

See

► Quadratic Programming

Work Schedule

A schedule of hours and days to be worked. This issue is of special importance to emergency services which are usually provided 24 hours-a-day, 7 days-a-week.

See

► Emergency Services

Worst-Case Analysis

For an algorithm and associated problem, the determination of an upper bound on the number of steps that the algorithm can take on any instance of the problem. For an optimization problem and an associated heuristic or suboptimal algorithm, worst-case analysis may include a statement regarding bounds on how far the objective function value for the solution returned by the algorithm can be from the true optimal value.

See

► Computational Complexity

X

$\bar{X}$ Chart

A quality control chart that plots a sample average of process output data over time, along with upper and lower control limits, to monitor variation in the process.

See

► Quality Control
► R Chart

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Y

Yield Management

Managing capacity/inventory of a fixed perishable product such as airline seats or hotel rooms to maximize profit or revenue. The term was coined in the airline industry, precursor to the modern fields of dynamic pricing and revenue management.

See

► Revenue Management

References

Ben-Yosef, E. (2005). *The evolution of the US airline industry: Theory, strategy and policy*. Dordrecht: Springer. Chapter 7: Yield management.

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,

Z

Zero-One Goal Programming

A goal programming methodology that generates solutions for decision variables where the variables must be equal to one or zero.

Zero-One Variables

► Binary Variable

Zero-Sum

A competitive or economic situation is termed zero-sum when the total amount of money or comparable measure that is gained by some participants is exactly equal to the total amount of the measure that is lost by the remaining participants. The term is specifically associated with a game in which the sum of the payoffs lost or gained by the players is fixed.

See

► Game Theory

Zero-Sum Game

A game in which one side's gain (or loss) is exactly offset by the total losses (or gains) of the remaining participant(s). In a two-person game, outcomes in which both sides win would not be possible.

See

► Game Theory

Zero-Sum Two-Person Game

► Game Theory

S.I. Gass, M.C. Fu (eds.), *Encyclopedia of Operations Research and Management Science*,
DOI 10.1007/978-1-4419-1153-7,