Introduction to OPERATIONS RESEARCH

Ninth Edition

Frederick S. Hillier
Gerald J. Lieberman

INTRODUCTION TO OPERATIONS RESEARCH

Ninth Edition

FREDERICK S. HILLIER

Stanford University

GERALD J. LIEBERMAN

Late of Stanford University

McGraw Hill **Higher Education**

Boston Burr Ridge, IL Dubuque, IA New York San Francisco St. Louis
Bangkok Bogotá Caracas Kuala Lumpur Lisbon London Madrid Mexico City
Milan Montreal New Delhi Santiago Seoul Singapore Sydney Taipei Toronto

The McGraw-Hill Companies

Higher Education

INTRODUCTION TO OPERATIONS RESEARCH, NINTH EDITION

Published by McGraw-Hill, a business unit of The McGraw-Hill Companies, Inc., 1221 Avenue of the Americas, New York, NY 10020.

This book is printed on acid-free paper

1 2 3 4 5 6 7 8 9 0 CCW/CCW 0 9

ISBN 978-0-07-337629-5
MHID 0-07-337629-9

Global Publisher: *Raghothaman Srinivasan*
Sponsoring Editor: *Debra B. Hash*
Director of Development: *Kristine Tibbetts*
Developmental Editor: *Lora Neyens*
Senior Marketing Manager: *Curt Reynolds*
Project Manager: *Melissa M. Leick*
Senior Production Supervisor: *Laura Fuller*
Senior Media Project Manager: *Sandra M. Schnee*
Associate Design Coordinator: *Brenda A. Rolwes*
Cover Designer: *Studio Montage, St. Louis, Missouri*
Compositor: *Laserwords Private Limited*
Typeface: *10/12 Times Roman*
Printer: *Courier Westford, Inc.*

Library of Congress Cataloging-in-Publication Data

Hillier, Frederick S.
Introduction to operations research / Frederick S. Hillier, Gerald J. Lieberman.—9th ed.
p. cm.
Includes index.
ISBN 978-0-07-337629-5 — ISBN 0-07-337629-9 (hbk. : alk. paper) 1. Operations research. I. Lieberman, Gerald J. II. Title.
T57.6.H53 2010
658.4'032—dc22

2008039045

www.mhhe.com

ABOUT THE AUTHORS

Frederick S. Hillier was born and raised in Aberdeen, Washington, where he was an award winner in statewide high school contests in essay writing, mathematics, debate, and music. As an undergraduate at Stanford University he ranked first in his engineering class of over 300 students. He also won the McKinsey Prize for technical writing, won the Outstanding Sophomore Debater award, played in the Stanford Woodwind Quintet, and won the Hamilton Award for combining excellence in engineering with notable achievements in the humanities and social sciences. Upon his graduation with a B.S. degree in Industrial Engineering, he was awarded three national fellowships (National Science Foundation, Tau Beta Pi, and Danforth) for graduate study at Stanford with specialization in operations research. After receiving his PhD degree, he joined the faculty of Stanford University, where he earned tenure at the age of 28 and the rank of full professor at 32. He also received visiting appointments at Cornell University, Carnegie-Mellon University, the Technical University of Denmark, the University of Canterbury (New Zealand), and the University of Cambridge (England). After 35 years on the Stanford faculty, he took early retirement from his faculty responsibilities in 1996 in order to focus full time on textbook writing, and now is Professor Emeritus of Operations Research at Stanford.

Dr. Hillier's research has extended into a variety of areas, including integer programming, queueing theory and its application, statistical quality control, and the application of operations research to the design of production systems and to capital budgeting. He has published widely, and his seminal papers have been selected for republication in books of selected readings at least 10 times. He was the first-prize winner of a research contest on "Capital Budgeting of Interrelated Projects" sponsored by The Institute of Management Sciences (TIMS) and the U.S. Office of Naval Research. He and Dr. Lieberman also received the honorable mention award for the 1995 Lanchester Prize (best English-language publication of any kind in the field of operations research), which was awarded by the Institute of Operations Research and the Management Sciences (INFORMS) for the 6th edition of this book. In addition, he was the recipient of the prestigious 2004 INFORMS Expository Writing Award for the 8th edition of this book.

Dr. Hillier has held many leadership positions with the professional societies in his field. For example, he has served as Treasurer of the Operations Research Society of America (ORSA), Vice President for Meetings of TIMS, Co-General Chairman of the 1989 TIMS International Meeting in Osaka, Japan, Chair of the TIMS Publications Committee, Chair of the ORSA Search Committee for Editor of *Operations Research,* Chair of the ORSA Resources Planning Committee, Chair of the ORSA/TIMS Combined Meetings Committee, and Chair of the John von Neumann Theory Prize Selection Committee for INFORMS. He continues to serve as the Series Editor for Springer's International Series in Operations Research and Management Science, a particularly prominent book series that he founded in 1993.

In addition to *Introduction to Operations Research* and two companion volumes, *Introduction to Mathematical Programming* (2nd ed., 1995) and *Introduction to Stochastic Models in Operations Research* (1990), his books are *The Evaluation of Risky Interrelated Investments* (North-Holland, 1969), *Queueing Tables and Graphs* (Elsevier North-Holland, 1981, co-authored by O. S. Yu, with D. M. Avis, L. D. Fossett, F. D. Lo, and M. I. Reiman), and *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets* (3rd ed., McGraw-Hill/Irwin, 2008, co-authored by M. S. Hillier).

The late **Gerald J. Lieberman** sadly passed away in 1999. He had been Professor Emeritus of Operations Research and Statistics at Stanford University, where he was the founding chair of the Department of Operations Research. He was both an engineer (having received an undergraduate degree in mechanical engineering from Cooper Union) and an operations research statistician (with an AM from Columbia University in mathematical statistics, and a PhD from Stanford University in statistics).

Dr. Lieberman was one of Stanford's most eminent leaders in recent decades. After chairing the Department of Operations Research, he served as Associate Dean of the School of Humanities and Sciences, Vice Provost and Dean of Research, Vice Provost and Dean of Graduate Studies, Chair of the Faculty Senate, member of the University Advisory Board, and Chair of the Centennial Celebration Committee. He also served as Provost or Acting Provost under three different Stanford presidents.

Throughout these years of university leadership, he also remained active professionally. His research was in the stochastic areas of operations research, often at the interface of applied probability and statistics. He published extensively in the areas of reliability and quality control, and in the modeling of complex systems, including their optimal design, when resources are limited.

Highly respected as a senior statesman of the field of operations research, Dr. Lieberman served in numerous leadership roles, including as the elected president of The Institute of Management Sciences. His professional honors included being elected to the National Academy of Engineering, receiving the Shewhart Medal of the American Society for Quality Control, receiving the Cuthbertson Award for exceptional service to Stanford University, and serving as a fellow at the Center for Advanced Study in the Behavioral Sciences. In addition, the Institute of Operations Research and the Management Sciences (INFORMS) awarded him and Dr. Hillier the honorable mention award for the 1995 Lanchester Prize for the 6th edition of this book. In 1996, INFORMS also awarded him the prestigious Kimball Medal for his exceptional contributions to the field of operations research and management science.

In addition to *Introduction to Operations Research* and two companion volumes, *Introduction to Mathematical Programming* (2nd ed., 1995) and *Introduction to Stochastic Models in Operations Research* (1990), his books are *Handbook of Industrial Statistics* (Prentice-Hall, 1955, co-authored by A. H. Bowker), *Tables of the Non-Central t-Distribution* (Stanford University Press, 1957, co-authored by G. J. Resnikoff), *Tables of the Hypergeometric Probability Distribution* (Stanford University Press, 1961, co-authored by D. Owen), *Engineering Statistics*, Second Edition (Prentice-Hall, 1972, co-authored by A. H. Bowker), and *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets* (McGraw-Hill/Irwin, 2000, co-authored by F. S. Hillier and M. S. Hillier).

ABOUT THE CASE WRITERS

Karl Schmedders is an associate professor in the Department of Managerial Economics and Decision Sciences at the Kellogg Graduate School of Management (Northwestern University), where he teaches quantitative methods for managerial decision making. His research interests include applications of operations research in economic theory, general equilibrium theory with incomplete markets, asset pricing, and computational economics. Dr. Schmedders received his doctorate in operations research from Stanford University, where he taught both undergraduate and graduate classes in operations research. Among the classes taught was a case studies course in operations research, and he subsequently was invited to speak at a conference sponsored by the Institute of Operations Research and the Management Sciences (INFORMS) about his successful experience with this course. He received several teaching awards at Stanford, including the university's prestigious Walter J. Gores Teaching Award. He also has received several teaching awards, including the L. G. Lavengood Professor of the Year at the Kellogg School of Management. While serving as a visiting professor at WHU Koblenz (a leading German business school), he won teaching awards there as well.

Molly Stephens is an associate in the Los Angeles office of Quinn, Emanuel, Urquhart, Oliver & Hedges, LLP. She graduated from Stanford University with a B.S. degree in Industrial Engineering and an M.S. degree in Operations Research. Ms. Stephens taught public speaking in Stanford's School of Engineering and served as a teaching assistant for a case studies course in operations research. As a teaching assistant, she analyzed operations research problems encountered in the real world and the transformation of these problems into classroom case studies. Her research was rewarded when she won an undergraduate research grant from Stanford to continue her work and was invited to speak at an INFORMS conference to present her conclusions regarding successful classroom case studies. Following graduation, Ms. Stephens worked at Andersen Consulting as a systems integrator, experiencing real cases from the inside, before resuming her graduate studies to earn a JD degree (with honors) from the University of Texas Law School at Austin.

DEDICATION

To the memory of our parents

and

To the memory of my beloved mentor,
Gerald J. Lieberman, who was one of the true
giants of our field

TABLE OF CONTENTS

SUPPLEMENTS AVAILABLE ON THE TEXT WEBSITE www.mhhe.com/hillier

PREFACE

When Jerry Lieberman and I started working on the first edition of this book 45 years ago, our goal was to develop a pathbreaking textbook that would help establish the future direction of education in what was then the emerging field of operations research. Following publication, it was unclear how well this particular goal was met, but what did become clear was that the demand for the book was far larger than either of us had anticipated. Neither of us could have imagined that this extensive worldwide demand would continue at such a high level for such an extended period of time.

The enthusiastic response to our first eight editions has been most gratifying. It was a particular pleasure to have the field's leading professional society, the international Institute for Operations Research and the Management Sciences (INFORMS), award the 6th edition honorable mention for the 1995 INFORMS Lanchester Prize (the prize awarded for the year's most outstanding English-language publication of any kind in the field of operations research).

Then, just after the publication of the eighth edition, it was especially gratifying to be the recipient of the prestigious 2004 INFORMS Expository Writing Award for this book, including receiving the following citation:

> Over 37 years, successive editions of this book have introduced more than one-half million students to the field and have attracted many people to enter the field for academic activity and professional practice. Many leaders in the field and many current instructors first learned about the field via an edition of this book. The extensive use of international student editions and translations into 15 other languages has contributed to spreading the field around the world. The book remains preeminent even after 37 years. Although the eighth edition just appeared, the seventh edition had 46 percent of the market for books of its kind, and it ranked second in international sales among all McGraw-Hill publications in engineering.
>
> Two features account for this success. First, the editions have been outstanding from students' points of view due to excellent motivation, clear and intuitive explanations, good examples of professional practice, excellent organization of material, very useful supporting software, and appropriate but not excessive mathematics. Second, the editions have been attractive from instructors' points of view because they repeatedly infuse state-of-the-art material with remarkable lucidity and plain language. For example, a wonderful chapter on metaheuristics was created for the eighth edition.

When we began work on the book 45 years ago, Jerry already was a prominent member of the field, a successful textbook writer, and the chairman of a renowned operations research program at Stanford University. I was a very young assistant professor just starting my career. It was a wonderful opportunity for me to work with and to learn from the master. I will be forever indebted to Jerry for giving me this opportunity.

Now, sadly, Jerry is no longer with us. During the progressive illness that led to his death nine years ago, I resolved that I would pick up the torch and devote myself to subsequent editions of this book, maintaining a standard that would fully honor Jerry. Therefore, I took early retirement from my faculty responsibilities at Stanford in order to work full time on textbook writing for the foreseeable future. This has enabled me to spend far more than the usual amount of time in preparing each new edition. It also has enabled me to closely monitor new trends and developments in the field in order to bring this edition completely up to date. This monitoring has led to the choice of the major revisions outlined below.

THE MAJOR REVISIONS

- **A Greatly Increased Emphasis on Real Applications.** Unbeknownst to the general public, the field of operations research is continuing to have an increasingly dramatic impact on the success of numerous companies and organizations around the world. Therefore, a special goal of this edition has been to tell this story much more forcefully, thereby exciting students about the great relevance of the material they are studying. We have pursued this goal in four ways. One is the *addition of 29 application vignettes* separated from the regular textual material that describe in a few paragraphs how an actual application of operations research had a powerful impact on a company or organization by using techniques like those being studied in that portion of the book. A second is the *addition of 71 selected references of award winning OR applications* given at the end of various chapters. A third is the *addition of a link to the journal articles that fully describe these 100 applications*, through a special arrangement with INFORMS. The final way is the *addition of many problems that require reading one or more of these articles*. Thus, the instructor now can motivate his or her lectures by having the students delve into real applications that dramatically demonstrate the relevance of the material being covered in the lectures.

  We are particularly excited about our new partnership with INFORMS, our field's preeminent professional society, to provide a link to these 100 articles describing dramatic OR applications. The Institute for Operations Research and the Management Sciences (INFORMS®) is a learned professional society for students, academics, and practitioners in quantitative and analytical fields. Information about INFORMS® journals, meetings, job bank, scholarships, awards, and teaching materials is at www.informs.org.
- **Approximately 200 New or Revised Problems.** The new problems include the ones involving real applications mentioned above. Other new problems also have been added, including a considerable number that support the new or revised topics mentioned later. Two new cases have been added for the chapter on decision analysis that are less complex than the two that already were there. In addition, many of the problems from the eighth edition have been revised. Therefore, an instructor who does not wish to assign problems that were assigned in previous classes has a substantial number from which to choose.
- **An Updating of the Software Accompanying the Book.** The next section will outline the wealth of software options that are provided with this new edition. The main difference from the eighth edition is that new, improved versions of several of the software packages now are available. For example, *Excel 2007* represents by far the most major revision of Excel and its user interface in many, many years, so this new version of Excel and its Solver has been fully integrated into the book (while pointing out differences for those still using old versions). Another important example is that, for the first time in 10 years, new versions of *TreePlan* and *SensIt* have just now become available and have been fully integrated into the decision analysis chapter. The latest versions of all the other software packages also are being provided with this new edition.
- **A New Section on Revenue Management.** A hallmark of new editions of this book has been the addition of substantial coverage of dramatic, recent developments that are beginning to revolutionize how certain areas of operations research are being practiced. For example, the eighth edition added a new chapter on metaheuristics, a new section on the incorporation of constraint programming, and a new section on multiechelon inventory models for supply chain management. This edition is adding another key new

topic with the *addition of a complete section on revenue management in the chapter on inventory theory*. This is a timely addition because of the dramatic impact that revenue management has been having in the airline industry and now is beginning to have in several other industries.

- **A Reorganization of the Chapter on the Theory of the Simplex Method.** Some instructors do not wish to take the time to cover the revised simplex method but may still want to introduce the matrix form of the simplex method and may still want to cover what we call the "fundamental insight" regarding the simplex method. Therefore, rather than covering the revised simplex method in Section 5.2 before turning to the fundamental insight in Section 5.3—as in the eighth edition—we now simply introduce the matrix form of the simplex method in Section 5.2, which flows directly into the fundamental insight in Section 5.3, after which we focus on the revised simplex method as an optional topic in Section 5.4.
- **A Simplified Method for Determining Utilities.** Among the various other smaller revisions throughout the book, perhaps the most noteworthy is a simplified presentation in Section 15.6 of how to determine utilities. This is done through outlining a simple "equivalent lottery method."
- **A Reorganization to Reduce the Size of the Book.** An unfortunate trend with early editions of this book was that each new edition was significantly larger than the previous one. This continued until the seventh edition had become considerably larger than is desirable for an introductory survey textbook. Therefore, I worked hard to substantially reduce the size of the eighth edition and adopted the goal of avoiding any growth in subsequent editions. The goal has been achieved for the current edition. This was accomplished through a variety of means. One was being careful not to add too much new material. Another was deleting two sections on real applications that had been in the eighth edition but no longer were needed because of the addition of application vignettes. Another was moving both the long Appendix 3.1 on the LINGO modeling language and the section on optimizing with OptQuest to the supplements on the book's website. (This decision regarding OptQuest was made easy by the fact that a new version is due out momentarily, but not in time for this edition, so it will be added later as a supplement.) Finally, a considerable number of sections were shortened. Otherwise, I have stuck closely to what I hope has become the familiar organization of the eighth edition after having made major changes for that edition.
- **Updating to Reflect the Current State of the Art.** A special effort has been made to keep the book completely up to date. This has included carefully updating both the selected references at the end of each chapter and the various footnotes referencing the latest research on the topics being covered.

A WEALTH OF SOFTWARE OPTIONS

A wealth of software options is being provided on the book's website www.mhhe.com/hillier as outlined below.

- Excel spreadsheets: state-of-the-art spreadsheet formulations are displayed in Excel files for all relevant examples throughout the book.
- Several Excel add-ins, including Premium Solver for Education (an enhancement of the basic Excel Solver), TreePlan (for decision analysis), SensIt (for probabilistic sensitivity analysis), RiskSim (for simulation), and Solver Table (for sensitivity analysis).
- A number of Excel templates for solving basic models.
- Student versions of LINDO (a traditional optimizer) and LINGO (a popular algebraic modeling language), along with formulations and solutions for all relevant examples throughout the book.

- Student versions of MPL (a leading algebraic modeling language) and its prime solver CPLEX (the most widely used state-of-the-art optimizer), along with an MPL Tutorial and MPL/CPLEX formulations and solutions for all relevant examples throughout the book.
- Student versions of several additional MPL solvers, including CONOPT (for convex programming), LGO (for global optimization), LINDO (for mathematical programming), CoinMP (for linear and integer programming), and BendX (for some stochastic models).
- Queueing Simulator (for the simulation of queueing systems).
- OR Tutor for illustrating various algorithms in action.
- Interactive Operations Research (IOR) Tutorial for efficiently learning and executing algorithms interactively, implemented in Java 2 in order to be platform independent.

Numerous students have found OR Tutor and IOR Tutorial very helpful for learning algorithms of operations research. When moving to the next stage of solving OR models automatically, surveys have found instructors almost equally split in preferring one of the following options for their students' use: (1) Excel spreadsheets, including the Excel Solver and other add-ins, (2) convenient traditional software (LINDO and LINGO), and (3) state-of-the-art OR software (MPL and CPLEX). For this edition, therefore, I have retained the philosophy of the last couple of editions of providing enough introduction in the book to enable the basic use of any of the three options without distracting those using another, while also providing ample supporting material for each option on the book's website.

We have elected to no longer include the Crystal Ball software package that was bundled with the eighth edition. Fortunately, many universities now have a site license for Crystal Ball and the package currently can also be downloaded for a free 30-day trial period, so it still is feasible to have students use this software, at least for a limited time. Therefore, this edition continues to use Crystal Ball in Section 20.6 and certain supplements to illustrate the exciting functionality that is now available for analyzing simulation models.

Additional Online Resources

- Several examples for nearly every book chapter are included in a *Worked Examples section* of the book's website to provide additional help to occasional students who need it without disrupting the flow of the text and adding unneeded pages for others. (The book uses boldface to highlight whenever an additional example on the current topic is available.)
- A *glossary* for every book chapter.
- *Data files* for various cases are included to enable students to focus on analysis rather than inputting large data sets.
- An abundance of supplementary textual material (including eight complete chapters).
- A *test bank* featuring moderately difficult questions that require students to show their work is being provided to instructors. Most of the questions in this test bank have previously been used successfully as test questions by the authors.
- Also available to instructors are a solutions manual and image files.

Electronic Textbook Option

This text is offered through CourseSmart for both instructors and students. CourseSmart is an online resource where students can purchase access to this and other McGraw-Hill textbooks in a digital format. Through their browser, students can access the complete text online at almost half the cost of a traditional text. Purchasing the eTextbook also allows students to take advantage of CourseSmart's web tools for learning, which include full text search, notes and highlighting, and e-mail tools for sharing notes between classmates.

To learn more about CourseSmart options, contact your sales representative or visit www.CourseSmart.com.

THE USE OF THE BOOK

The overall thrust of all the revision efforts has been to build upon the strengths of previous editions to more fully meet the needs of today's students. These revisions make the book even more suitable for use in a modern course that reflects contemporary practice in the field. The use of software is integral to the practice of operations research, so the wealth of software options accompanying the book provides great flexibility to the instructor in choosing the preferred types of software for student use. All the educational resources accompanying the book further enhance the learning experience. Therefore, the book and its website should fit a course where the instructor wants the students to have a single self-contained textbook that complements and supports what happens in the classroom.

The McGraw-Hill editorial team and I think that the net effect of the revision has been to make this edition even more of a "student's book"—clear, interesting, and well-organized with lots of helpful examples and illustrations, good motivation and perspective, easy-to-find important material, and enjoyable homework, without too much notation, terminology, and dense mathematics. We believe and trust that the numerous instructors who have used previous editions will agree that this is the best edition yet.

The prerequisites for a course using this book can be relatively modest. As with previous editions, the mathematics has been kept at a relatively elementary level. Most of Chaps. 1 to 14 (introduction, linear programming, and mathematical programming) require no mathematics beyond high school algebra. Calculus is used only in Chaps. 12 (Nonlinear Programming) and in one example in Chap. 10 (Dynamic Programming). Matrix notation is used in Chap. 5 (The Theory of the Simplex Method), Chap. 6 (Duality Theory and Sensitivity Analysis), Sec. 7.4 (An Interior-Point Algorithm), and Chap. 12, but the only background needed for this is presented in Appendix 4. For Chaps. 15 to 20 (probabilistic models), a previous introduction to probability theory is assumed, and calculus is used in a few places. In general terms, the mathematical maturity that a student achieves through taking an elementary calculus course is useful throughout Chaps. 15 to 20 and for the more advanced material in the preceding chapters.

The content of the book is aimed largely at the upper-division undergraduate level (including well-prepared sophomores) and at first-year (master's level) graduate students. Because of the book's great flexibility, there are many ways to package the material into a course. Chapters 1 and 2 give an introduction to the subject of operations research. Chapters 3 to 14 (on linear programming and on mathematical programming) may essentially be covered independently of Chaps. 15 to 20 (on probabilistic models), and vice-versa. Furthermore, the individual chapters among Chaps. 3 to 14 are almost independent, except that they all use basic material presented in Chap. 3 and perhaps in Chap. 4. Chapter 6 and Sec. 7.2 also draw upon Chap. 5. Sections 7.1 and 7.2 use parts of Chap. 6. Section 9.6 assumes an acquaintance with the problem formulations in Secs. 8.1 and 8.3, while prior exposure to Secs. 7.3 and 8.2 is helpful (but not essential) in Sec. 9.7. Within Chaps. 15 to 20, there is considerable flexibility of coverage, although some integration of the material is available.

An elementary survey course covering linear programming, mathematical programming, and some probabilistic models can be presented in a quarter (40 hours) or semester by selectively drawing from material throughout the book. For example, a good survey of the field can be obtained from Chaps. 1, 2, 3, 4, 15, 17, 18, and 20, along with parts of

Chaps. 9 to 13. A more extensive elementary survey course can be completed in two quarters (60 to 80 hours) by excluding just a few chapters, for example, Chaps. 7, 14, and 19. Chapters 1 to 8 (and perhaps part of Chap. 9) form an excellent basis for a (one-quarter) course in linear programming. The material in Chaps. 9 to 14 covers topics for another (one-quarter) course in other deterministic models. Finally, the material in Chaps. 15 to 20 covers the probabilistic (stochastic) models of operations research suitable for presentation in a (one-quarter) course. In fact, these latter three courses (the material in the entire text) can be viewed as a basic one-year sequence in the techniques of operations research, forming the core of a master's degree program. Each course outlined has been presented at either the undergraduate or graduate level at Stanford University, and this text has been used in the manner suggested.

The book's website will provide updates about the book, including an errata. To access this site, visit www.mhhe.com/hillier.

ACKNOWLEDGMENTS

I am indebted to an excellent group of reviewers who provided sage advice for the revision process. This group included

Chun-Hung Chen, George Mason University
Mary Court, University of Oklahoma
Todd Easton, Kansas State University
Samuel H. Huang, University of Cincinnati
Ronald Giachetti, Florida International University
Mary E. Kurz, Clemson University
Wooseung Jang, University of Missouri-Columbia
Shafiu Jibrin, Northern Arizona University
Roger Johnson, South Dakota School of Mines & Technology
Emanuel Melachrinoudis, Northeastern University
Clark A. Mount-Campbell, The Ohio State University
Jose A. Ventura, Pennsylvania State University
John Wu, Kansas State University

I also am grateful to Garrett Van Ryzin for his expert advice regarding the new section on revenue management, to Charles McCallum, Jr., for providing lists of typos in the 8th edition three times, and to Bjarni Kristjansson for providing up-to-date information on the sizes of problems being solved successfully by the latest optimization software. In addition, thanks go to those instructors and students who sent email messages to provide their feedback on the 8th edition.

This edition was very much of a team effort. Our case writers, Karl Schmedders and Molly Stephens (both graduates of our department), wrote 24 elaborate cases for the 7th edition, and all of these cases continue to accompany this new edition. One of our department's current PhD students, Pelin Canbolat, did an excellent job in preparing the solutions manual. She went above and beyond the call of duty by typing nearly all of the solutions that had been handwritten for preceding editions, as well as providing helpful input for this edition. One of our former PhD students, Michael O'Sullivan, developed OR Tutor for the 7th edition (and continued here), based on part of the software that my son Mark Hillier had developed for the 5th and 6th editions. Mark (who was born the same year as the first edition, earned his PhD at Stanford, and now is a tenured Associate Professor of Quantitative Methods at the University of Washington) provided both the spreadsheets and the Excel files (including many Excel templates) for this edition, as well as the Solver Table and Queueing Simulator. He also gave helpful advice on both the textual material and software for this edition, and contributed greatly to Chapters 21 and 28 on the book's website. Another Stanford PhD graduate, William Sun (CEO of the software company Accelet Corporation), and his team did a brilliant job of starting with much of Mark's earlier software and implementing it anew in Java 2 as IOR Tutorial for the 7th edition. They again did a masterful job of further enhancing IOR Tutorial for the 8th and subsequent editions. Linus Schrage of the University of Chicago and LINDO Systems (and who took an introductory operations research course from me 45 years ago) provided LINGO and LINDO for the book's website. He also supervised the further development of LINGO/LINDO files for the various chapters as well as providing tutorial material for the book's website. Another long-time friend, Bjarni Kristjansson (who heads Maximal Software), did the same thing for the MPL/CPLEX files and MPL tutorial material, as well as arranging to provide student versions of MPL, CPLEX, and various other solvers for the book's website. My wife, Ann Hillier, devoted numerous long days and nights to sitting with a Macintosh, doing word processing and constructing many figures and tables. They all were vital members of the team.

In addition to Accelet Corporation, LINDO Systems, and Maximal Software, we are deeply indebted to several other companies for providing software to accompany this edition. These include Frontline Systems (for providing Premium Solver for Education), ILOG (for providing the CPLEX solver used with the MPL Student Edition), ARKI Corporation (for providing the CONOPT convex programming solver used with the MPL Student Edition), and PCS Inc. (for providing the LGO global optimization solver used with the MPL Student Edition). We also are grateful to Professor Michael Middleton for providing newly improved versions of TreePlan and SensIt, as well as RiskSim. Finally, we appreciate the cooperation of INFORMS in providing a link to the articles in *Interfaces* that describe the applications of OR that are summarized in the application vignettes and other selected references of award winning OR applications provided in the book.

It was a real pleasure working with McGraw-Hill's thoroughly professional editorial and production staff, including Debra Hash (Sponsoring Editor) and Lora Kalb-Neyens (Developmental Editor).

Just as so many individuals made important contributions to this edition, I would like to invite each of you to start contributing to the next edition by using my email address below to send me your comments, suggestions, and errata to help me improve the book in the future. In giving my email address, let me also assure instructors that I will continue to follow the policy of not providing solutions to problems and cases in the book to anybody (including your students) who contacts me.

Enjoy the book.

Frederick S. Hillier
Stanford University (fhillier@stanford.edu)

May 2008

CHAPTER 1

Introduction

1.1 THE ORIGINS OF OPERATIONS RESEARCH

Since the advent of the industrial revolution, the world has seen a remarkable growth in the size and complexity of organizations. The artisans' small shops of an earlier era have evolved into the billion-dollar corporations of today. An integral part of this revolutionary change has been a tremendous increase in the division of labor and segmentation of management responsibilities in these organizations. The results have been spectacular. However, along with its blessings, this increasing specialization has created new problems, problems that are still occurring in many organizations. One problem is a tendency for the many components of an organization to grow into relatively autonomous empires with their own goals and value systems, thereby losing sight of how their activities and objectives mesh with those of the overall organization. What is best for one component frequently is detrimental to another, so the components may end up working at cross purposes. A related problem is that as the complexity and specialization in an organization increase, it becomes more and more difficult to allocate the available resources to the various activities in a way that is most effective for the organization as a whole. These kinds of problems and the need to find a better way to solve them provided the environment for the emergence of **operations research** (commonly referred to as **OR**).

The roots of OR can be traced back many decades,[1] when early attempts were made to use a scientific approach in the management of organizations. However, the beginning of the activity called *operations research* has generally been attributed to the military services early in World War II. Because of the war effort, there was an urgent need to allocate scarce resources to the various military operations and to the activities within each operation in an effective manner. Therefore, the British and then the U.S. military management called upon a large number of scientists to apply a scientific approach to dealing with this and other strategic and tactical problems. In effect, they were asked to do *research on* (military) *operations*. These teams of scientists were the first OR teams. By developing effective methods of using the new tool of radar, these teams were instrumental in winning the Air Battle of Britain. Through their research on how to better manage convoy and antisubmarine

[1]Selected Reference 2 provides an entertaining history of operations research that traces its roots as far back as 1564 by describing a considerable number of scientific contributions from 1564 to 1935 that influenced the subsequent development of OR.

operations, they also played a major role in winning the Battle of the North Atlantic. Similar efforts assisted the Island Campaign in the Pacific.

When the war ended, the success of OR in the war effort spurred interest in applying OR outside the military as well. As the industrial boom following the war was running its course, the problems caused by the increasing complexity and specialization in organizations were again coming to the forefront. It was becoming apparent to a growing number of people, including business consultants who had served on or with the OR teams during the war, that these were basically the same problems that had been faced by the military but in a different context. By the early 1950s, these individuals had introduced the use of OR to a variety of organizations in business, industry, and government. The rapid spread of OR soon followed.

At least two other factors that played a key role in the rapid growth of OR during this period can be identified. One was the substantial progress that was made early in improving the techniques of OR. After the war, many of the scientists who had participated on OR teams or who had heard about this work were motivated to pursue research relevant to the field; important advancements in the state of the art resulted. A prime example is the *simplex method* for solving linear programming problems, developed by George Dantzig in 1947. Many of the standard tools of OR, such as linear programming, dynamic programming, queueing theory, and inventory theory, were relatively well developed before the end of the 1950s.

A second factor that gave great impetus to the growth of the field was the onslaught of the *computer revolution.* A large amount of computation is usually required to deal most effectively with the complex problems typically considered by OR. Doing this by hand would often be out of the question. Therefore, the development of electronic digital computers, with their ability to perform arithmetic calculations millions of times faster than a human being can, was a tremendous boon to OR. A further boost came in the 1980s with the development of increasingly powerful personal computers accompanied by good software packages for doing OR. This brought the use of OR within the easy reach of much larger numbers of people, and this progress further accelerated in the 1990s and into the 21st century. Today, literally millions of individuals have ready access to OR software. Consequently, a whole range of computers from mainframes to laptops now are being routinely used to solve OR problems, including some of enormous size.

1.2 THE NATURE OF OPERATIONS RESEARCH

As its name implies, operations research involves "research on operations." Thus, operations research is applied to problems that concern how to conduct and coordinate the *operations* (i.e., the *activities*) within an organization. The nature of the organization is essentially immaterial, and, in fact, OR has been applied extensively in such diverse areas as manufacturing, transportation, construction, telecommunications, financial planning, health care, the military, and public services, to name just a few. Therefore, the breadth of application is unusually wide.

The *research* part of the name means that operations research uses an approach that resembles the way research is conducted in established scientific fields. To a considerable extent, the *scientific method* is used to investigate the problem of concern. (In fact, the term *management science* sometimes is used as a synonym for operations research.) In particular, the process begins by carefully observing and formulating the problem, including gathering all relevant data. The next step is to construct a scientific (typically mathematical) model that attempts to abstract the essence of the real problem. It is then hypothesized that this model is a sufficiently precise representation of the essential features of the situation that the conclusions (solutions) obtained from the model are also valid for the real problem. Next, suitable experiments are conducted to test this hypothesis, modify it as needed, and eventually verify some form of the hypothesis. (This step is frequently referred to as *model validation.*) Thus, in a certain sense, operations research involves creative scientific research into the fundamental properties of operations. However, there is more to it than this. Specifically, OR is also concerned with the practical management of the organization. Therefore, to be successful, OR must also provide positive, understandable conclusions to the decision maker(s) when they are needed.

Still another characteristic of OR is its broad viewpoint. As implied in the preceding section, OR adopts an organizational point of view. Thus, it attempts to resolve the conflicts of interest among the components of the organization in a way that is best for the organization as a whole. This does not imply that the study of each problem must give explicit consideration to all aspects of the organization; rather, the objectives being sought must be consistent with those of the overall organization.

An additional characteristic is that OR frequently attempts to search for a *best* solution (referred to as an *optimal* solution) for the model that represents the problem under consideration. (We say *a* best instead of *the* best solution because there may be multiple solutions tied as best.) Rather than simply improving the status quo, the goal is to identify a best possible course of action. Although it must be interpreted carefully in terms of the practical needs of management, this "search for optimality" is an important theme in OR.

All these characteristics lead quite naturally to still another one. It is evident that no single individual should be expected to be an expert on all the many aspects of OR work or the problems typically considered; this would require a group of individuals having diverse backgrounds and skills. Therefore, when a full-fledged OR study of a new problem is undertaken, it is usually necessary to use a *team approach.* Such an OR team typically needs to include individuals who collectively are highly trained in mathematics, statistics and probability theory, economics, business administration, computer science, engineering and the physical sciences, the behavioral sciences, and the special techniques of OR. The team also needs to have the necessary experience and variety of skills to give appropriate consideration to the many ramifications of the problem throughout the organization.

1.3 THE IMPACT OF OPERATIONS RESEARCH

Operations research has had an impressive impact on improving the efficiency of numerous organizations around the world. In the process, OR has made a significant contribution to increasing the productivity of the economies of various countries. There now are a few dozen member countries in the International Federation of Operational Research Societies (IFORS), with each country having a national OR society. Both Europe and Asia have federations of OR societies to coordinate holding international conferences and publishing international journals in those continents. In addition, the Institute for Operations Research and the Management Sciences (INFORMS) is an international OR society. Among its various journals is one called *Interfaces* that regularly publishes articles describing major OR studies and the impact they had on their organizations.

To give you a better notion of the wide applicability of OR, we list some actual applications in Table 1.1. Note the diversity of organizations and applications in the first two columns. The third column identifies the section where an "application vignette" devotes several paragraphs to describing the application and also references an article that provides full details. (You can see the first of these application vignettes in this section.) The last column indicates that these applications typically resulted in annual savings in the many millions of dollars. Furthermore, additional benefits not recorded in the table

TABLE 1.1 Applications of operations research to be described in application vignettes

Organization	Area of Application	Section	Annual Savings
Federal Express	Logistical planning of shipments	1.3	Not estimated
Continental Airlines	Reassign crews to flights when schedule disruptions occur	2.2	$40 million
Swift & Company	Improve sales and manufacturing performance	3.1	$12 million
Memorial Sloan-Kettering Cancer Center	Design of radiation therapy	3.4	$459 million
United Airlines	Plan employee work schedules at airports and reservations offices	3.4	$6 million
Welch's	Optimize use and movement of raw materials	3.6	$150,000
Samsung Electronics	Reduce manufacturing times and inventory levels	4.3	$200 million more revenue
Pacific Lumber Company	Long-term forest ecosystem management	6.7	$398 million NPV
Procter & Gamble	Redesign the production and distribution system	8.1	$200 million
Canadian Pacific Railway	Plan routing of rail freight	9.3	$100 million
United Airlines	Reassign airplanes to flights when disruptions occur	9.6	Not estimated
U.S. Military	Logistical planning of Operations Desert Storm	10.3	Not estimated
Air New Zealand	Airline crew scheduling	11.2	$6.7 million
Taco Bell	Plan employee work schedules at restaurants	11.5	$13 million
Waste Management	Develop a route-management system for trash collection and disposal	11.7	$100 million
Bank Hapoalim Group	Develop a decision-support system for investment advisors	12.1	$31 million more revenue
Sears	Vehicle routing and scheduling for home services and deliveries	13.2	$42 million
Conoco-Phillips	Evaluate petroleum exploration projects	15.2	Not estimated
Workers' Compensation Board	Manage high-risk disability claims and rehabilitation	15.3	$4 million
Westinghouse	Evaluate research-and-development projects	15.4	Not estimated
Merrill Lynch	Manage liquidity risk for revolving credit lines	16.2	$4 billion more liquidity
PSA Peugeot Citroën	Guide the design process for efficient car assembly plants	16.8	$130 million more profit
KeyCorp	Improve efficiency of bank teller service	17.6	$20 million
General Motors	Improve efficiency of production lines	17.9	$90 million
Deere & Company	Management of inventories throughout a supply chain	18.5	$1 billion less inventory
Time Inc.	Management of distribution channels for magazines	18.7	$3.5 million more profit
Bank One Corporation	Management of credit lines and interest rates for credit cards	19.2	$75 million more profit
Merrill Lynch	Pricing analysis for providing financial services	20.2	$50 million more revenue
AT&T	Design and operation of call centers	20.5	$750 million more profit

(e.g., improved service to customers and better managerial control) sometimes were considered to be even more important than these financial benefits. (You will have an opportunity to investigate these less tangible benefits further in Probs. 1.3-1, 1.3-2, and 1.3-3.) A link to the articles that describe these applications in detail is included on our website, www.mhhe.com/hillier.

Although most routine OR studies provide considerably more modest benefits than the applications summarized in Table 1.1, the figures in the rightmost column of this table do accurately reflect the dramatic impact that large, well-designed OR studies occasionally can have.

An Application Vignette

Federal Express (FedEx) is the world's largest express transportation company. Every working day, it delivers more than 6.5 million documents, packages, and other items throughout the United States and more than 220 countries and territories around the world. In some cases, these shipments can be guaranteed overnight delivery by 10:30 A.M. the next morning.

The logistical challenges involved in providing this service are staggering. These millions of daily shipments must be individually sorted and routed to the correct general location (usually by aircraft) and then delivered to the exact destination (usually by motorized vehicle) in an amazingly short period of time. How is all this possible?

Operations research (OR) is the technological engine that drives this company. Ever since its founding in 1973, OR has helped make its major business decisions, including equipment investment, route structure, scheduling, finances, and location of facilities. After OR was credited with literally saving the company during its early years, it became the custom to have OR represented at the weekly senior management meetings and, indeed, several of the senior corporate vice presidents have come up from the outstanding FedEx OR group.

FedEx has come to be acknowledged as a world-class company. It routinely ranks among the top companies on *Fortune Magazine*'s annual listing of the "World's Most Admired Companies." It also was the first winner (in 1991) of the prestigious prize now known as the INFORMS Prize, which is awarded annually for the effective and repeated integration of OR into organizational decision making in pioneering, varied, novel, and lasting ways.

Source: R. O. Mason, J. L. McKenney, W. Carlson, and D. Copeland, "Absolutely, Positively Operations Research: The Federal Express Story," *Interfaces,* **27**(2): 17–36, March-April 1997. (A link to this article is provided on our website, www.mhhe.com/hillier.)

1.4 ALGORITHMS AND OR COURSEWARE

An important part of this book is the presentation of the major **algorithms** (systematic solution procedures) of OR for solving certain types of problems. Some of these algorithms are amazingly efficient and are routinely used on problems involving hundreds or thousands of variables. You will be introduced to how these algorithms work and what makes them so efficient. You then will use these algorithms to solve a variety of problems on a computer. The **OR Courseware** contained on the book's website (www.mhhe.com/hillier) will be a key tool for doing all this.

One special feature in your OR Courseware is a program called **OR Tutor.** This program is intended to be your personal tutor to help you learn the algorithms. It consists of many *demonstration examples* that display and explain the algorithms in action. These "demos" supplement the examples in the book.

In addition, your OR Courseware includes a special software package called **Interactive Operations Research Tutorial,** or **IOR Tutorial** for short. Implemented in Java, this innovative package is designed specifically to enhance the learning experience of students using this book. IOR Tutorial includes many *interactive procedures* for executing the algorithms interactively in a convenient format. The computer does all the routine calculations while you focus on learning and executing the logic of the algorithm. You should find these interactive procedures a very efficient and enlightening way of doing many of your homework problems. IOR Tutorial also includes a number of other helpful procedures, including some *automatic procedures* for executing algorithms automatically and several procedures that provide graphical displays of how the solution provided by an algorithm varies with the data of the problem.

In practice, the algorithms normally are executed by commercial software packages. We feel that it is important to acquaint students with the nature of these packages that they will be using after graduation. Therefore, your OR Courseware includes a wealth of material to introduce you to three particularly popular software packages described next.

Together, these packages will enable you to solve nearly all the OR models encountered in this book very efficiently. We have added our own *automatic procedures* to IOR Tutorial in a few cases where these packages are not applicable.

A very popular approach now is to use today's premier spreadsheet package, *Microsoft Excel,* to formulate small OR models in a spreadsheet format. The **Excel Solver** (or an enhanced version of this add-in, such as **Premium Solver for Education** included in your OR Courseware) then is used to solve the models. Your OR Courseware includes separate Excel files, based on the relatively new Excel 2007, for nearly every chapter in this book. Each time a chapter presents an example that can be solved using Excel, the complete spreadsheet formulation and solution is given in that chapter's Excel files. For many of the models in the book, an *Excel template* also is provided that already includes all the equations necessary to solve the model. Some *Excel add-ins* also are included on the book's website.

After many years, **LINDO** (and its companion modeling language **LINGO**) continues to be a popular OR software package. Student versions of LINDO and LINGO now can be downloaded free from the Web. This student version also is provided in your OR Courseware. As for Excel, each time an example can be solved with this package, all the details are given in a LINGO/LINDO file for that chapter in your OR Courseware.

CPLEX is an elite state-of-the-art software package that is widely used for solving large and challenging OR problems. When dealing with such problems, it is common to also use a *modeling system* to efficiently formulate the mathematical model and enter it into the computer. **MPL** is a user-friendly modeling system that uses CPLEX as its main solver, but also has several other solvers, including LINDO, CoinMP (introduced in Sec. 4.8), CONOPT (introduced in Sec. 12.9), LGO (introduced in Sec. 12.10), and BendX (useful for solving some stochastic models). A student version of MPL, along with the latest student version of CPLEX and its other solvers, is available free by downloading it from the Web. For your convenience, we also have included this student version (including all the solvers just mentioned) in your OR Courseware. Once again, all the examples that can be solved with this package are detailed in MPL/CPLEX files for the corresponding chapters in your OR Courseware.

We will further describe these three software packages and how to use them later (especially near the end of Chaps. 3 and 4). Appendix 1 also provides documentation for the OR Courseware, including OR Tutor and IOR Tutorial.

To alert you to relevant material in OR Courseware, the end of each chapter from Chap. 3 onward has a list entitled *Learning Aids for This Chapter on our Website.* As explained at the beginning of the problem section for each of these chapters, symbols also are placed to the left of each problem number or part where any of this material (including demonstration examples and interactive procedures) can be helpful.

Another learning aid provided on our website is a set of **Worked Examples** for each chapter (from Chap. 3 onward). These complete examples supplement the examples in the book for your use as needed, but without interrupting the flow of the material on those many occasions when you don't need to see an additional example. You also might find these supplementary examples helpful when preparing for an examination. We always will mention whenever a supplementary example on the current topic is included in the Worked Examples section of the book's website. To make sure you don't overlook this mention, we will boldface the words **additional example** (or something similar) each time.

The website also includes a glossary for each chapter.

SELECTED REFERENCES

1. Bell, P. C., C. K. Anderson, and S. P. Kaiser: "Strategic Operations Research and the Edelman Prize Finalist Applications 1989–1998," *Operations Research,* **51**(1): 17–31, January–February 2003.
2. Gass, S. I., and A. A. Assad: *An Annotated Timeline of Operations Research: An Informal History,* Kluwer Academic Publishers (now Springer), Boston, 2005.
3. Gass, S. I., and C. M. Harris (eds.): *Encyclopedia of Operations Research and Management Science,* 2d ed., Kluwer Academic Publishers (now Springer), Boston, 2001.
4. Horner, P.: "History in the Making," *OR/MS Today,* **29**(5): 30–39, October 2002.
5. Horner, P. (ed.): "Special Issue: Executive's Guide to Operations Research," *OR/MS Today,* Institute for Operations Research and the Management Sciences, **27**(3), June 2000.
6. Kirby, M. W.: "Operations Research Trajectories: The Anglo-American Experience from the 1940s to the 1990s," *Operations Research,* **48**(5): 661–670, September–October 2000.
7. Miser, H. J.: "The Easy Chair: What OR/MS Workers Should Know About the Early Formative Years of Their Profession," *Interfaces,* **30**(2): 99–111, March–April 2000.
8. Wein, L. M. (ed.): "50th Anniversary Issue," *Operations Research* (a special issue featuring personalized accounts of some of the key early theoretical and practical developments in the field), **50**(1), January–February 2002.

PROBLEMS

1.3-1. Select one of the applications of operations research listed in Table 1.1. Read the article that is referenced in the application vignette presented in the section shown in the third column. (A link to all these articles is provided on our website, www.mhhe.com/hillier.) Write a two-page summary of the application and the benefits (including nonfinancial benefits) it provided.

1.3-2. Select three of the applications of operations research listed in Table 1.1. For each one, read the article that is referenced in the application vignette presented in the section shown in the third column. (A link to all these articles is provided on our website, www.mhhe.com/hillier.) For each one, write a one-page summary of the application and the benefits (including nonfinancial benefits) it provided.

1.3-3. Read the referenced article that fully describes the OR study summarized in the application vignette presented in Sec. 1.3. List the various financial and nonfinancial benefits that resulted from this study.

CHAPTER 2

Overview of the Operations Research Modeling Approach

The bulk of this book is devoted to the mathematical methods of operations research (OR). This is quite appropriate because these quantitative techniques form the main part of what is known about OR. However, it does not imply that practical OR studies are primarily mathematical exercises. As a matter of fact, the mathematical analysis often represents only a relatively small part of the total effort required. The purpose of this chapter is to place things into better perspective by describing all the major phases of a typical OR study.

One way of summarizing the usual (overlapping) phases of an OR study is the following:

1. Define the problem of interest and gather relevant data.
2. Formulate a mathematical model to represent the problem.
3. Develop a computer-based procedure for deriving solutions to the problem from the model.
4. Test the model and refine it as needed.
5. Prepare for the ongoing application of the model as prescribed by management.
6. Implement.

Each of these phases will be discussed in turn in the following sections.

The selected references at the end of the chapter include some award-winning OR studies that provide excellent examples of how to execute these phases well. We will intersperse snippets from some of these examples throughout the chapter. If you decide that you would like to learn more about these award-winning applications of operations research, a link to the articles that describe these OR studies in detail is included on the book's website, www.mhhe.com/hillier.

2.1 DEFINING THE PROBLEM AND GATHERING DATA

In contrast to textbook examples, most practical problems encountered by OR teams are initially described to them in a vague, imprecise way. Therefore, the first order of business is to study the relevant system and develop a well-defined statement of the problem to be considered. This includes determining such things as the appropriate objectives, constraints on what can be done, interrelationships between the area to be studied and other areas of the organization, possible alternative courses of action, time limits for making a decision, and so on. This process of problem definition is a crucial one because it greatly affects how relevant the conclusions of the study will be. It is difficult to extract a "right" answer from the "wrong" problem!

The first thing to recognize is that an OR team normally works in an *advisory capacity.* The team members are not just given a problem and told to solve it however they see fit. Instead, they advise management (often one key decision maker). The team performs a detailed technical analysis of the problem and then presents recommendations to management. Frequently, the report to management will identify a number of alternatives that are particularly attractive under different assumptions or over a different range of values of some policy parameter that can be evaluated only by management (e.g., the trade-off between *cost* and *benefits*). Management evaluates the study and its recommendations, takes into account a variety of intangible factors, and makes the final decision based on its best judgment. Consequently, it is vital for the OR team to get on the same wavelength as management, including identifying the "right" problem from management's viewpoint, and to build the support of management for the course that the study is taking.

Ascertaining the *appropriate objectives* is a very important aspect of problem definition. To do this, it is necessary first to identify the member (or members) of management who actually will be making the decisions concerning the system under study and then to probe into this individual's thinking regarding the pertinent objectives. (Involving the decision maker from the outset also is essential to build her or his support for the implementation of the study.)

By its nature, OR is concerned with the welfare of the *entire organization* rather than that of only certain of its components. An OR study seeks solutions that are optimal for the overall organization rather than suboptimal solutions that are best for only one component. Therefore, the objectives that are formulated ideally should be those of the entire organization. However, this is not always convenient. Many problems primarily concern only a portion of the organization, so the analysis would become unwieldy if the stated objectives were too general and if explicit consideration were given to all side effects on the rest of the organization. Instead, the objectives used in the study should be as specific as they can be while still encompassing the main goals of the decision maker and maintaining a reasonable degree of consistency with the higher-level objectives of the organization.

For profit-making organizations, one possible approach to circumventing the problem of suboptimization is to use *long-run profit maximization* (considering the time value of money) as the sole objective. The adjective *long-run* indicates that this objective provides the flexibility to consider activities that do not translate into profits *immediately* (e.g., research and development projects) but need to do so *eventually* in order to be worthwhile. This approach has considerable merit. This objective is specific enough to be used conveniently, and yet it seems to be broad enough to encompass the basic goal of profit-making organizations. In fact, some people believe that all other legitimate objectives can be translated into this one.

However, in actual practice, many profit-making organizations do not use this approach. A number of studies of U.S. corporations have found that management tends to adopt the goal of *satisfactory profits,* combined with *other objectives,* instead of focusing on long-run profit maximization. Typically, some of these *other* objectives might be to maintain stable profits, increase (or maintain) one's share of the market, provide for product diversification, maintain stable prices, improve worker morale, maintain family control of the business, and increase company prestige. Fulfilling these objectives might achieve long-run profit maximization, but the relationship may be sufficiently obscure that it may not be convenient to incorporate them all into this one objective.

Furthermore, there are additional considerations involving social responsibilities that are distinct from the profit motive. The five parties generally affected by a business firm located in a single country are (1) the *owners* (stockholders, etc.), who desire profits (dividends, stock appreciation, and so on); (2) the *employees,* who desire steady employment at reasonable wages; (3) the *customers,* who desire a reliable product at a reasonable price; (4) the *suppliers,* who desire integrity and a reasonable selling price for their goods; and (5) the *government* and hence the *nation,* which desire payment of fair taxes and consideration of the national interest. All five parties make essential contributions to the firm, and the firm should not be viewed as the exclusive servant of any one party for the exploitation of others. By the same token, international corporations acquire additional obligations to follow socially responsible practices. Therefore, while granting that management's prime responsibility is to make profits (which ultimately benefits all five parties), we note that its broader social responsibilities also must be recognized.

OR teams typically spend a surprisingly large amount of time *gathering relevant data* about the problem. Much data usually are needed both to gain an accurate understanding of the problem and to provide the needed input for the mathematical model being formulated in the next phase of study. Frequently, much of the needed data will not be available when the study begins, either because the information never has been kept or because what was kept is outdated or in the wrong form. Therefore, it often is necessary to install a new computer-based *management information system* to collect the necessary data on an ongoing basis and in the needed form. The OR team normally needs to enlist the assistance of various other key individuals in the organization, including *information technology* (IT) specialists, to track down all the vital data. Even with this effort, much of the data may be quite "soft," i.e., rough estimates based only on educated guesses. Typically, an OR team will spend considerable time trying to improve the precision of the data and then will make do with the best that can be obtained.

With the widespread use of databases and the explosive growth in their sizes in recent years, OR teams now frequently find that their biggest data problem is not that too little is available but that there is too much data. There may be thousands of sources of data, and the total amount of data may be measured in gigabytes or even terabytes. In this environment, locating the particularly relevant data and identifying the interesting patterns in these data can become an overwhelming task. One of the newer tools of OR teams is a technique called **data mining** that addresses this problem. Data mining methods search large databases for interesting patterns that may lead to useful decisions. (Selected Reference 2 at the end of the chapter provides further background about data mining.)

Example. In the late 1990s, full-service financial services firms came under assault from electronic brokerage firms offering extremely low trading costs. **Merrill Lynch** responded by conducting a major OR study that led to a complete overhaul in how it charged for its services, ranging from a full-service asset-based option (charge a fixed percentage of the value of the assets held rather than for individual trades) to a low-cost option for clients wishing to invest online directly. *Data collection and processing* played a key role in the study. To analyze the impact of individual client behavior in response to different options, the team needed to assemble a comprehensive 200 gigabyte client database involving 5 million clients, 10 million accounts, 100 million trade records, and 250 million ledger records. This required merging, reconciling, filtering, and cleaning data from numerous production databases. The adoption of the recommendations of the study led to a one-year increase of nearly \$50 billion in client assets held and nearly \$80 million more revenue. (Selected Reference A2 describes this study in detail.)

2.2 FORMULATING A MATHEMATICAL MODEL

After the decision maker's problem is defined, the next phase is to reformulate this problem in a form that is convenient for analysis. The conventional OR approach for doing this is to construct a mathematical model that represents the essence of the problem. Before discussing how to formulate such a model, we first explore the nature of models in general and of mathematical models in particular.

Models, or idealized representations, are an integral part of everyday life. Common examples include model airplanes, portraits, globes, and so on. Similarly, models play an important role in science and business, as illustrated by models of the atom, models of genetic structure, mathematical equations describing physical laws of motion or chemical reactions, graphs, organizational charts, and industrial accounting systems. Such models are invaluable for abstracting the essence of the subject of inquiry, showing interrelationships, and facilitating analysis.

Mathematical models are also idealized representations, but they are expressed in terms of mathematical symbols and expressions. Such laws of physics as $F = ma$ and $E = mc^2$ are familiar examples. Similarly, the mathematical model of a business problem is the system of equations and related mathematical expressions that describe the essence of the problem. Thus, if there are n related quantifiable decisions to be made, they are represented as **decision variables** (say, $x_1, x_2, \ldots, x_n$) whose respective values are to be determined. The appropriate measure of performance (e.g., profit) is then expressed as a mathematical function of these decision variables (for example, $P = 3x_1 + 2x_2 + \ldots + 5x_n$). This function is called the **objective function.** Any restrictions on the values that can be assigned to these decision variables are also expressed mathematically, typically by means of inequalities or equations (for example, $x_1 + 3x_1x_2 + 2x_2 \leq 10$). Such mathematical expressions for the restrictions often are called **constraints.** The constants (namely, the coefficients and right-hand sides) in the constraints and the objective function are called the **parameters** of the model. The mathematical model might then say that the problem is to choose the values of the decision variables so as to maximize the objective function, subject to the specified constraints. Such a model, and minor variations of it, typifies the models used in OR.

Determining the appropriate values to assign to the parameters of the model (one value per parameter) is both a critical and a challenging part of the model-building process. In contrast to textbook problems where the numbers are given to you, determining parameter values for real problems requires *gathering relevant data.* As discussed in the preceding section, gathering accurate data frequently is difficult. Therefore, the value assigned to a parameter often is, of necessity, only a rough estimate. Because of the uncertainty about the true value of the parameter, it is important to analyze how the solution derived from the model would change (if at all) if the value assigned to the parameter were changed to other plausible values. This process is referred to as **sensitivity analysis,** as discussed further in the next section (and much of Chap. 6).

Although we refer to "the" mathematical model of a business problem, real problems normally don't have just a single "right" model. Section 2.4 will describe how the process of testing a model typically leads to a succession of models that provide better and better representations of the problem. It is even possible that two or more completely different types of models may be developed to help analyze the same problem.

You will see numerous examples of mathematical models throughout the remainder of this book. One particularly important type that is studied in the next several chapters is the **linear programming model,** where the mathematical functions appearing in both the objective function and the constraints are all linear functions. In Chap. 3, specific linear programming models are constructed to fit such diverse problems as determining (1) the mix of products that maximizes profit, (2) the design of radiation therapy that effectively attacks a tumor while minimizing the damage to nearby healthy tissue, (3) the allocation of acreage to crops that maximizes total net return, and (4) the combination of pollution abatement methods that achieves air quality standards at minimum cost.

Mathematical models have many advantages over a verbal description of the problem. One advantage is that a mathematical model describes a problem much more concisely. This tends to make the overall structure of the problem more comprehensible, and it helps to reveal important cause-and-effect relationships. In this way, it indicates more clearly what additional data are relevant to the analysis. It also facilitates dealing with the problem in its entirety and considering all its interrelationships simultaneously. Finally, a mathematical model forms a bridge to the use of high-powered mathematical techniques and computers to analyze the problem. Indeed, packaged software for both personal computers and mainframe computers has become widely available for solving many mathematical models.

However, there are pitfalls to be avoided when you use mathematical models. Such a model is necessarily an abstract idealization of the problem, so approximations and simplifying assumptions generally are required if the model is to be *tractable* (capable of being solved). Therefore, care must be taken to ensure that the model remains a valid representation of the problem. The proper criterion for judging the validity of a model is whether the model predicts the relative effects of the alternative courses of action with sufficient accuracy to permit a sound decision. Consequently, it is not necessary to include unimportant details or factors that have approximately the same effect for all the alternative courses of action considered. It is not even necessary that the absolute magnitude of the measure of performance be approximately correct for the various alternatives, provided that their relative values (i.e., the differences between their values) are sufficiently precise. Thus, all that is required is that there be a high *correlation* between the prediction by the model and what would actually happen in the real world. To ascertain whether this requirement is satisfied, it is important to do considerable *testing* and consequent modifying of the model, which will be the subject of Sec. 2.4. Although this testing phase is placed later in the chapter, much of this *model validation* work actually is conducted during the model-building phase of the study to help guide the construction of the mathematical model.

In developing the model, a good approach is to begin with a very simple version and then move in evolutionary fashion toward more elaborate models that more nearly reflect the complexity of the real problem. This process of *model enrichment* continues only as long as the model remains tractable. The basic trade-off under constant consideration is between the *precision* and the *tractability* of the model. (See Selected Reference 8 for a detailed description of this process.)

A crucial step in formulating an OR model is the construction of the objective function. This requires developing a quantitative measure of performance relative to each of the decision maker's ultimate objectives that were identified while the problem was being defined. If there are multiple objectives, their respective measures commonly are then transformed and combined into a composite measure, called the **overall measure of performance.** This overall measure might be something tangible (e.g., profit) corresponding to a higher goal of the organization, or it might be abstract (e.g., utility). In the latter case, the task of developing this measure tends to be a complex one requiring a careful comparison of the objectives and their relative importance. After the overall measure of performance is developed, the objective function is then obtained by expressing this measure as a mathematical function of the decision variables. Alternatively, there also are methods for explicitly considering multiple objectives simultaneously, and one of these (goal programming) is discussed in the supplement to Chap. 7.

An Application Vignette

Continental Airlines is a major U.S. air carrier that transports passengers, cargo, and mail. It operates more than 2,000 daily departures to well over 100 domestic destinations and nearly 100 foreign destinations.

Airlines like Continental face schedule disruptions daily because of unexpected events, including inclement weather, aircraft mechanical problems, and crew unavailability. These disruptions can cause flight delays and cancellations. As a result, crews may not be in position to service their remaining scheduled flights. Airlines must reassign crews quickly to cover open flights and to return them to their original schedules in a cost-effective manner while honoring all government regulations, contractual obligations, and quality-of-life requirements.

To address such problems, an OR team at Continental Airlines developed a detailed *mathematical model* for reassigning crews to flights as soon as such emergencies arise. Because the airline has thousands of crews and daily flights, the model needed to be huge to consider all possible pairings of crews with flights. Therefore, the model has *millions of decision variables* and *many thousands of constraints.* In its first year of use (mainly in 2001), the model was applied four times to recover from major schedule disruptions (two snowstorms, a flood, and the September 11 terrorist attacks). This led to *savings of approximately* **$40 million.** Subsequent applications extended to many daily minor disruptions as well.

Although other airlines subsequently scrambled to apply operations research in a similar way, this initial advantage over other airlines in being able to recover more quickly from schedule disruptions with fewer delays and cancelled flights left Continental Airlines in a relatively strong position as the airline industry struggled through a difficult period during the initial years of the 21st century. This initiative led to Continental winning the prestigious First Prize in the 2002 international competition for the Franz Edelman Award for Achievement in Operations Research and the Management Sciences.

Source: G. Yu, M. Argüello, C. Song, S. M. McGowan, and A. White, "A New Era for Crew Recovery at Continental Airlines," *Interfaces,* **33**(1): 5–22, Jan.–Feb. 2003. (A link to this article is provided on our website, www.mhhe.com/hillier.)

Example. The Netherlands government agency responsible for water control and public works, the **Rijkswaterstaat,** commissioned a major OR study to guide the development of a new national water management policy. The new policy saved hundreds of millions of dollars in investment expenditures and reduced agricultural damage by about $15 million per year, while decreasing thermal and algae pollution. Rather than formulating *one* mathematical model, this OR study developed a comprehensive, integrated system of 50 models! Furthermore, for some of the models, both simple and complex versions were developed. The simple version was used to gain basic insights, including trade-off analyses. The complex version then was used in the final rounds of the analysis or whenever greater accuracy or more detailed outputs were desired. The overall OR study directly involved over 125 person-years of effort (more than one-third in data gathering), created several dozen computer programs, and structured an enormous amount of data. (Selected Reference A7 describes this study in detail.)

2.3 DERIVING SOLUTIONS FROM THE MODEL

After a mathematical model is formulated for the problem under consideration, the next phase in an OR study is to develop a procedure (usually a computer-based procedure) for deriving solutions to the problem from this model. You might think that this must be the major part of the study, but actually it is not in most cases. Sometimes, in fact, it is a relatively simple step, in which one of the standard **algorithms** (systematic solution procedures) of OR is applied on a computer by using one of a number of readily available software packages. For experienced OR practitioners, finding a solution is the fun part, whereas the real work comes in the preceding and following steps, including the *postoptimality analysis* discussed later in this section.

Since much of this book is devoted to the subject of how to obtain solutions for various important types of mathematical models, little needs to be said about it here. However, we do need to discuss the nature of such solutions.

A common theme in OR is the search for an **optimal,** or best, **solution.** Indeed, many procedures have been developed, and are presented in this book, for finding such solutions for certain kinds of problems. However, it needs to be recognized that these solutions are optimal only with respect to the model being used. Since the model necessarily is an idealized rather than an exact representation of the real problem, there cannot be any utopian guarantee that the optimal solution for the model will prove to be the best possible solution that could have been implemented for the real problem. There just are too many imponderables and uncertainties associated with real problems. However, if the model is well formulated and tested, the resulting solution should tend to be a good approximation to an ideal course of action for the real problem. Therefore, rather than be deluded into demanding the impossible, you should make the test of the practical success of an OR study hinge on whether it provides a better guide for action than can be obtained by other means.

Eminent management scientist and Nobel Laureate in economics Herbert Simon points out that **satisficing** is much more prevalent than optimizing in actual practice. In coining the term *satisficing* as a combination of the words *satisfactory* and *optimizing,* Simon is describing the tendency of managers to seek a solution that is "good enough" for the problem at hand. Rather than trying to develop an overall measure of performance to optimally reconcile conflicts between various desirable objectives (including well-established criteria for judging the performance of different segments of the organization), a more pragmatic approach may be used. Goals may be set to establish minimum satisfactory levels of performance in various areas, based perhaps on past levels of performance or on what the competition is achieving. If a solution is found that enables all these goals to be met, it is likely to be adopted without further ado. Such is the nature of satisficing.

The distinction between optimizing and satisficing reflects the difference between theory and the realities frequently faced in trying to implement that theory in practice. In the words of one of England's pioneering OR leaders, Samuel Eilon, "Optimizing is the science of the ultimate; satisficing is the art of the feasible."[1]

OR teams attempt to bring as much of the "science of the ultimate" as possible to the decision-making process. However, the successful team does so in full recognition of the overriding need of the decision maker to obtain a satisfactory guide for action in a reasonable period of time. Therefore, the goal of an OR study should be to conduct the study in an optimal manner, regardless of whether this involves finding an optimal solution for the model. Thus, in addition to pursuing the science of the ultimate, the team should also consider the cost of the study and the disadvantages of delaying its completion, and then attempt to maximize the net benefits resulting from the study. In recognition of this concept, OR teams occasionally use only **heuristic procedures** (i.e., intuitively designed procedures that do not guarantee an optimal solution) to find a good **suboptimal solution.** This is most often the case when the time or cost required to find an optimal solution for an adequate model of the problem would be very large. In recent years, great progress has been made in developing efficient and effective **metaheuristics** that provide both a general structure and strategy guidelines for designing a specific heuristic procedure to fit a particular kind of problem. The use of metaheuristics (the subject of Chap. 13) is continuing to grow.

[1]S. Eilon, "Goals and Constraints in Decision-making," *Operational Research Quarterly,* **23:** 3–15, 1972. Address given at the 1971 annual conference of the Canadian Operational Research Society.

The discussion thus far has implied that an OR study seeks to find only one solution, which may or may not be required to be optimal. In fact, this usually is not the case. An optimal solution for the original model may be far from ideal for the real problem, so additional analysis is needed. Therefore, **postoptimality analysis** (analysis done after finding an optimal solution) is a very important part of most OR studies. This analysis also is sometimes referred to as **what-if analysis** because it involves addressing some questions about *what* would happen to the optimal solution *if* different assumptions are made about future conditions. These questions often are raised by the managers who will be making the ultimate decisions rather than by the OR team.

The advent of powerful spreadsheet software now has frequently given spreadsheets a central role in conducting postoptimality analysis. One of the great strengths of a spreadsheet is the ease with which it can be used interactively by anyone, including managers, to see what happens to the optimal solution when changes are made to the model. This process of experimenting with changes in the model also can be very helpful in providing understanding of the behavior of the model and increasing confidence in its validity.

In part, postoptimality analysis involves conducting **sensitivity analysis** to determine which parameters of the model are most critical (the "sensitive parameters") in determining the solution. A common definition of *sensitive parameter* (used throughout this book) is the following.

> For a mathematical model with specified values for all its parameters, the model's **sensitive parameters** are the parameters whose value cannot be changed without changing the optimal solution.

Identifying the sensitive parameters is important, because this identifies the parameters whose value must be assigned with special care to avoid distorting the output of the model.

The value assigned to a parameter commonly is just an *estimate* of some quantity (e.g., unit profit) whose exact value will become known only after the solution has been implemented. Therefore, after the sensitive parameters are identified, special attention is given to estimating each one more closely, or at least its range of likely values. One then seeks a solution that remains a particularly good one for all the various combinations of likely values of the sensitive parameters.

If the solution is implemented on an ongoing basis, any later change in the value of a sensitive parameter immediately signals a need to change the solution.

In some cases, certain parameters of the model represent policy decisions (e.g., resource allocations). If so, there frequently is some flexibility in the values assigned to these parameters. Perhaps some can be increased by decreasing others. Postoptimality analysis includes the investigation of such trade-offs.

In conjunction with the study phase discussed in Sec. 2.4 (testing the model), postoptimality analysis also involves obtaining a sequence of solutions that comprises a series of improving approximations to the ideal course of action. Thus, the apparent weaknesses in the initial solution are used to suggest improvements in the model, its input data, and perhaps the solution procedure. A new solution is then obtained, and the cycle is repeated. This process continues until the improvements in the succeeding solutions become too small to warrant continuation. Even then, a number of alternative solutions (perhaps solutions that are optimal for one of several plausible versions of the model and its input data) may be presented to management for the final selection. As suggested in Sec. 2.1, this presentation of alternative solutions would normally be done whenever the final choice among these alternatives should be based on considerations that are best left to the judgment of management.

Example. Consider again the **Rijkswaterstaat** OR study of national water management policy for the Netherlands, introduced at the end of Sec. 2.2. This study did not conclude by recommending just a single solution. Instead, a number of attractive alternatives were identified, analyzed, and compared. The final choice was left to the Dutch political process, culminating with approval by Parliament. *Sensitivity analysis* played a major role in this study. For example, certain parameters of the models represented environmental standards. Sensitivity analysis included assessing the impact on water management problems if the values of these parameters were changed from the current environmental standards to other reasonable values. Sensitivity analysis also was used to assess the impact of changing the assumptions of the models, e.g., the assumption on the effect of future international treaties on the amount of pollution entering the Netherlands. A variety of *scenarios* (e.g., an extremely dry year and an extremely wet year) also were analyzed, with appropriate probabilities assigned.

2.4 TESTING THE MODEL

Developing a large mathematical model is analogous in some ways to developing a large computer program. When the first version of the computer program is completed, it inevitably contains many bugs. The program must be thoroughly tested to try to find and correct as many bugs as possible. Eventually, after a long succession of improved programs, the programmer (or programming team) concludes that the current program now is generally giving reasonably valid results. Although some minor bugs undoubtedly remain hidden in the program (and may never be detected), the major bugs have been sufficiently eliminated that the program now can be reliably used.

Similarly, the first version of a large mathematical model inevitably contains many flaws. Some relevant factors or interrelationships undoubtedly have not been incorporated into the model, and some parameters undoubtedly have not been estimated correctly. This is inevitable, given the difficulty of communicating and understanding all the aspects and subtleties of a complex operational problem as well as the difficulty of collecting reliable data. Therefore, before you use the model, it must be thoroughly tested to try to identify and correct as many flaws as possible. Eventually, after a long succession of improved models, the OR team concludes that the current model now is giving reasonably valid results. Although some minor flaws undoubtedly remain hidden in the model (and may never be detected), the major flaws have been sufficiently eliminated so that the model now can be reliably used.

This process of testing and improving a model to increase its validity is commonly referred to as **model validation.**

It is difficult to describe how model validation is done, because the process depends greatly on the nature of the problem being considered and the model being used. However, we make a few general comments, and then we give an example. (See Selected Reference 3 for a detailed discussion.)

Since the OR team may spend months developing all the detailed pieces of the model, it is easy to "lose the forest for the trees." Therefore, after the details ("the trees") of the initial version of the model are completed, a good way to begin model validation is to take a fresh look at the overall model ("the forest") to check for obvious errors or oversights. The group doing this review preferably should include at least one individual who did not participate in the formulation of the model. Reexamining the definition of the problem and comparing it with the model may help to reveal mistakes. It is also useful to make sure that all the mathematical expressions are *dimensionally consistent* in the units used. Additional insight into the validity of the model can sometimes be obtained by varying the values of the parameters and/or the decision variables and checking to see whether the output from the model behaves in a plausible manner. This is often especially revealing when the parameters or variables are assigned extreme values near their maxima or minima.

A more systematic approach to testing the model is to use a **retrospective test.** When it is applicable, this test involves using historical data to reconstruct the past and then determining how well the model and the resulting solution would have performed if they had been used. Comparing the effectiveness of this hypothetical performance with what actually happened then indicates whether using this model tends to yield a significant improvement over current practice. It may also indicate areas where the model has shortcomings and requires modifications. Furthermore, by using alternative solutions from the model and estimating their hypothetical historical performances, considerable evidence can be gathered regarding how well the model predicts the relative effects of alternative courses of actions.

On the other hand, a disadvantage of retrospective testing is that it uses the same data that guided the formulation of the model. The crucial question is whether the past is truly representative of the future. If it is not, then the model might perform quite differently in the future than it would have in the past.

To circumvent this disadvantage of retrospective testing, it is sometimes useful to continue the status quo temporarily. This provides new data that were not available when the model was constructed. These data are then used in the same ways as those described here to evaluate the model.

Documenting the process used for model validation is important. This helps to increase confidence in the model for subsequent users. Furthermore, if concerns arise in the future about the model, this documentation will be helpful in diagnosing where problems may lie.

Example. Consider an OR study done for **IBM** to integrate its national network of spare-parts inventories to improve service support for IBM's customers. This study resulted in a new inventory system that improved customer service while reducing the value of IBM's inventories by over $250 million and saving an additional $20 million per year through improved operational efficiency. A particularly interesting aspect of the model validation phase of this study was the way that *future users* of the inventory system were incorporated into the testing process. Because these future users (IBM managers in functional areas responsible for implementation of the inventory system) were skeptical about the system being developed, representatives were appointed to a *user team* to serve as advisers to the OR team. After a preliminary version of the new system had been developed (based on a multiechelon inventory model), a *preimplementation test* of the system was conducted. Extensive feedback from the user team led to major improvements in the proposed system. (Selected Reference A5 describes this study in detail.)

2.5 PREPARING TO APPLY THE MODEL

What happens after the testing phase has been completed and an acceptable model has been developed? If the model is to be used repeatedly, the next step is to install a well-documented *system* for applying the model as prescribed by management. This system will include the model, solution procedure (including postoptimality analysis), and operating procedures for implementation. Then, even as personnel changes, the system can be called on at regular intervals to provide a specific numerical solution.

This system usually is *computer-based.* In fact, a considerable number of computer programs often need to be used and integrated. *Databases* and *management information systems* may provide up-to-date input for the model each time it is used, in which case interface programs are needed. After a solution procedure (another program) is applied to the model, additional computer programs may trigger the implementation of the results automatically. In other cases, an *interactive* computer-based system called a **decision support system** is installed to help managers use data and models to support (rather than replace) their decision making as needed. Another program may generate *managerial reports* (in the language of management) that interpret the output of the model and its implications for application.

In major OR studies, several months (or longer) may be required to develop, test, and install this computer system. Part of this effort involves developing and implementing a process for maintaining the system throughout its future use. As conditions change over time, this process should modify the computer system (including the model) accordingly.

Example. The application vignette in Sec. 2.2 described an OR study done for **Continental Airlines** that led to the formulation of a huge mathematical model for reassigning crews to flights when schedule disruptions occur. Because the model needs to be applied immediately when a disruption occurs, a *decision support system* called *CrewSolver* was developed to incorporate both the model and a huge in-memory data store representing current operations. CrewSolver enables a crew coordinator to input data about the schedule disruption and then to use a graphical user interface to request an immediate solution for how to reassign crews to flights.

2.6 IMPLEMENTATION

After a system is developed for applying the model, the last phase of an OR study is to implement this system as prescribed by management. This phase is a critical one because it is here, and only here, that the benefits of the study are reaped. Therefore, it is important for the OR team to participate in launching this phase, both to make sure that model solutions are accurately translated to an operating procedure and to rectify any flaws in the solutions that are then uncovered.

The success of the implementation phase depends a great deal upon the support of both top management and operating management. The OR team is much more likely to gain this support if it has kept management well informed and encouraged management's active guidance throughout the course of the study. Good communications help to ensure that the study accomplishes what management wanted, and also give management a greater sense of ownership of the study, which encourages their support for implementation.

The implementation phase involves several steps. First, the OR team gives operating management a careful explanation of the new system to be adopted and how it relates to operating realities. Next, these two parties share the responsibility for developing the procedures required to put this system into operation. Operating management then sees that a detailed indoctrination is given to the personnel involved, and the new course of action is initiated. If successful, the new system may be used for years to come. With this in mind, the OR team monitors the initial experience with the course of action taken and seeks to identify any modifications that should be made in the future.

Throughout the entire period during which the new system is being used, it is important to continue to obtain feedback on how well the system is working and whether the assumptions of the model continue to be satisfied. When significant deviations from the original assumptions occur, the model should be revisited to determine if any modifications should be made in the system. The postoptimality analysis done earlier (as described in Sec. 2.3) can be helpful in guiding this review process.

Upon culmination of a study, it is appropriate for the OR team to *document* its methodology clearly and accurately enough so that the work is *reproducible. Replicability* should be part of the professional ethical code of the operations researcher. This condition is especially crucial when controversial public policy issues are being studied.

Example. This example illustrates how a successful implementation phase might need to involve thousands of employees before undertaking the new procedures. **Samsung Electronics Corp.** initiated a major OR study in March 1996 to develop new methodologies and scheduling applications that would streamline the entire semiconductor manufacturing process and reduce work-in-progress inventories. The study continued for over five years, culminating in June 2001, largely because of the extensive effort required for the implementation phase. The OR team needed to gain the support of numerous managers, manufacturing staff, and engineering staff by training them in the principles and logic of the new manufacturing procedures. Ultimately, more than 3,000 people attended training sessions. The new procedures then were phased in gradually to build confidence. However, this patient implementation process paid huge dividends. The new procedures transformed the company from being the least efficient manufacturer in the semiconductor industry to becoming the most efficient. This resulted in increased revenues of over $1 billion by the time the implementation of the OR study was completed. (Selected Reference A11 describes this study in detail.)

2.7 CONCLUSIONS

Although the remainder of this book focuses primarily on *constructing* and *solving* mathematical models, in this chapter we have tried to emphasize that this constitutes only a portion of the overall process involved in conducting a typical OR study. The other phases described here also are very important to the success of the study. Try to keep in perspective the role of the model and the solution procedure in the overall process as you move through the subsequent chapters. Then, after gaining a deeper understanding of mathematical models, we suggest that you plan to return to review this chapter again in order to further sharpen this perspective.

OR is closely intertwined with the use of computers. In the early years, these generally were mainframe computers, but now personal computers and workstations are being widely used to solve OR models.

In concluding this discussion of the major phases of an OR study, it should be emphasized that there are many exceptions to the "rules" prescribed in this chapter. By its very nature, OR requires considerable ingenuity and innovation, so it is impossible to write down any standard procedure that should always be followed by OR teams. Rather, the preceding description may be viewed as a model that roughly represents how successful OR studies are conducted.

SELECTED REFERENCES

1. Board, J., C. Sutcliffe, and W. T. Ziemba: "Applying Operations Research Techniques to Financial Markets," *Interfaces,* **33**(2): 12–24, March–April 2003.
2. Bradley, P. S., U. M. Fayyad, and O. L. Mangasarian: "Mathematical Programming for Data Mining: Formulations and Challenges," *INFORMS Journal on Computing,* **11**(3): 217–238, Summer 1999.
3. Gass, S. I.: "Decision-Aiding Models: Validation, Assessment, and Related Issues for Policy Analysis," *Operations Research,* **31:** 603–631, 1983.
4. Gass, S. I.: "Model World: Danger, Beware the User as Modeler," *Interfaces,* **20**(3): 60–64, May–June 1990.
5. Hall, R. W.: "What's So Scientific about MS/OR?" *Interfaces,* **15**(2): 40–45, March–April 1985.
6. Howard, R. A.: "The Ethical OR/MS Professional," *Interfaces,* **31**(6): 69–82, November–December 2001.
7. Miser, H. J.: "The Easy Chair: Observation and Experimentation," *Interfaces,* **19**(5): 23–30, September–October 1989.
8. Morris, W. T.: "On the Art of Modeling," *Management Science,* **13:** B707–717, 1967.

9. Murphy, F. H.: "The Occasional Observer: Some Simple Precepts for Project Success," *Interfaces,* **28**(5): 25–28, September–October 1998.
10. Murphy, F. H.: "ASP, The Art and Science of Practice: Elements of the Practice of Operations Research: A Framework," *Interfaces,* **35**(2): 154–163, March–April 2005.
11. Pidd, M.: "Just Modeling Through: A Rough Guide to Modeling," *Interfaces,* **29**(2): 118–132, March–April 1999.
12. Williams, H. P.: *Model Building in Mathematical Programming,* 4th ed., Wiley, New York, 1999.
13. Wright, P. D., M. J. Liberatore, and R. L. Nydick: "A Survey of Operations Research Models and Applications in Homeland Security," *Interfaces,* **36**(6): 514–529, November–December 2006.

Some Award-Winning Applications of the OR Modeling Approach:

(A link to all these articles is provided on our website, www.mhhe.com/hillier.)

A1. Alden, J. M., L. D. Burns, T. Costy, R. D. Hutton, C. A. Jackson, D. S. Kim, K. A. Kohls, J. H. Owen, M. A. Turnquist, and D. J. V. Veen: "General Motors Increases Its Production Throughput," *Interfaces,* **36**(1): 6–25, January–February 2006.

A2. Altschuler, S., D. Batavia, J. Bennett, R. Labe, B. Liao, R. Nigam, and J. Oh: "Pricing Analysis for Merrill Lynch Integrated Choice," *Interfaces,* **32**(1): 5–19, January–February 2002.

A3. Bixby, A., B. Downs, and M. Self: "A Scheduling and Capable-to-Promise Application for Swift & Company," *Interfaces,* **36**(1): 69–86, January–February 2006.

A4. Braklow, J. W., W. W. Graham, S. M. Hassler, K. E. Peck, and W. B. Powell: "Interactive Optimization Improves Service and Performance for Yellow Freight System," *Interfaces,* **22**(1): 147–172, January–February 1992.

A5. Cohen, M., P. V. Kamesam, P. Kleindorfer, H. Lee, and A. Tekerian: "Optimizer: IBM's Multi-Echelon Inventory System for Managing Service Logistics," *Interfaces,* **20**(1): 65–82, January–February 1990.

A6. DeWitt, C. W., L. S. Lasdon, A. D. Waren, D. A. Brenner, and S. A. Melhem: "OMEGA: An Improved Gasoline Blending System for Texaco," *Interfaces,* **19**(1): 85–101, January–February 1990.

A7. Goeller, B. F., and the PAWN team: "Planning the Netherlands' Water Resources," *Interfaces,* **15**(1): 3–33, January–February 1985.

A8. Hicks, R., R. Madrid, C. Milligan, R. Pruneau, M. Kanaley, Y. Dumas, B. Lacroix, J. Desrosiers, and F. Soumis: "Bombardier Flexjet Significantly Improves Its Fractional Aircraft Ownership Operations," *Interfaces,* **35**(1): 49–60, January–February 2005.

A9. Kaplan, E. H., and E. O'Keefe: "Let the Needles Do the Talking! Evaluating the New Haven Needle Exchange," *Interfaces,* **23**(1): 7–26, January–February 1993.

A10. Kok, T. de, F. Janssen, J. van Doremalen, E. van Wachem, M. Clerkx, and W. Peeters: "Philips Electronics Synchronizes Its Supply Chain to End the Bullwhip Effect," *Interfaces,* **35**(1): 37–48, January–February 2005.

A11. Leachman, R. C., J. Kang, and V. Lin: "SLIM: Short Cycle Time and Low Inventory in Manufacturing at Samsung Electronics," *Interfaces,* **32**(1): 61–77, January–February 2002.

A12. Taylor, P. E., and S. J. Huxley: "A Break from Tradition for the San Francisco Police: Patrol Officer Scheduling Using an Optimization-Based Decision Support System," *Interfaces,* **19**(1): 4–24, January–February 1989.

PROBLEMS

2.1-1. The example in Sec. 2.1 summarizes an award-winning OR study done for Merrill Lynch. Read Selected Reference A2 that describes this study in detail.

(a) Summarize the background that led to undertaking this study.
(b) Quote the one-sentence statement of the general mission of the OR group (called the management science group) that conducted this study.
(c) Identify the type of data that the management science group obtained for each client.
(d) Identify the new pricing options that were provided to the company's clients as a result of this study.
(e) What was the resulting impact on Merrill Lynch's competitive position?

2.1-2. Read Selected Reference A1 that describes an award-winning OR study done for General Motors.

(a) Summarize the background that led to undertaking this study.
(b) What was the goal of this study?

(c) Describe how software was used to automate the collection of the needed data.
(d) The improved production throughput that resulted from this study yielded how much in documented savings and increased revenue?

2.1-3. Read Selected Reference A12 that describes an OR study done for the San Francisco Police Department.
(a) Summarize the background that led to undertaking this study.
(b) Define part of the problem being addressed by identifying the six directives for the scheduling system to be developed.
(c) Describe how the needed data were gathered.
(d) List the various tangible and intangible benefits that resulted from the study.

2.1-4. Read Selected Reference A9 that describes an OR study done for the Health Department of New Haven, Connecticut.
(a) Summarize the background that led to undertaking this study.
(b) Outline the system developed to track and test each needle and syringe in order to gather the needed data.
(c) Summarize the initial results from this tracking and testing system.
(d) Describe the impact and potential impact of this study on public policy.

2.2-1. Read the referenced article that fully describes the OR study summarized in the application vignette presented in Sec. 2.2. List the various financial and nonfinancial benefits that resulted from this study.

2.2-2. Read Selected Reference A3 that describes an OR study done for Swift & Company.
(a) Summarize the background that led to undertaking this study.
(b) Describe the purpose of each of the three general types of models formulated during this study.
(c) How many specific models does the company now use as a result of this study?
(d) List the various financial and nonfinancial benefits that resulted from this study.

2.2-3. Read Selected Reference A7 that describes an OR study done for the Rijkswaterstaat of the Netherlands. (Focus especially on pp. 3–20 and 30–32.)
(a) Summarize the background that led to undertaking this study.
(b) Summarize the purpose of each of the five mathematical models described on pp. 10–18.
(c) Summarize the "impact measures" (measures of performance) for comparing policies that are described on pp. 6–7 of this article.
(d) List the various tangible and intangible benefits that resulted from the study.

2.2-4. Read Selected Reference 5.
(a) Identify the author's example of a model in the natural sciences and of a model in OR.
(b) Describe the author's viewpoint about how basic precepts of using models to do research in the natural sciences can also be used to guide *research on operations* (OR).

2.3-1. Read Selected Reference A10 that describes an OR study done for Philips Electronics.
(a) Summarize the background that led to undertaking this study.
(b) What was the purpose of this study?
(c) What were the benefits of developing software to support problem solving speedily?
(d) List the four steps in the collaborative-planning process that resulted from this study.
(e) List the various financial and nonfinancial benefits that resulted from this study.

2.3-2. Refer to Selected Reference 5.
(a) Describe the author's viewpoint about whether the sole goal in using a model should be to find its optimal solution.
(b) Summarize the author's viewpoint about the complementary roles of modeling, evaluating information from the model, and then applying the decision maker's judgment when deciding on a course of action.

2.4-1. Refer to pp. 18–20 of Selected Reference A7 that describes an OR study done for the Rijkswaterstaat of the Netherlands. Describe an important lesson that was gained from model validation in this study.

2.4-2. Read Selected Reference 7. Summarize the author's viewpoint about the roles of observation and experimentation in the model validation process.

2.4-3. Read pp. 603–617 of Selected Reference 3.
(a) What does the author say about whether a model can be completely validated?
(b) Summarize the distinctions made between *model validity, data validity, logical/mathematical validity, predictive validity, operational validity,* and *dynamic validity.*
(c) Describe the role of *sensitivity analysis* in testing the *operational validity* of a model.
(d) What does the author say about whether there is a validation methodology that is appropriate for all models?
(e) Cite the page in the article that lists basic validation steps.

2.5-1. Read Selected Reference A6 that describes an OR study done for Texaco.
(a) Summarize the background that led to undertaking this study.
(b) Briefly describe the user interface with the decision support system OMEGA that was developed as a result of this study.
(c) OMEGA is constantly being updated and extended to reflect changes in the operating environment. Briefly describe the various kinds of changes involved.
(d) Summarize how OMEGA is used.
(e) List the various tangible and intangible benefits that resulted from the study.

2.5-2. Refer to Selected Reference A4 that describes an OR study done for Yellow Freight System, Inc.
(a) Referring to pp. 147–149 of this article, summarize the background that led to undertaking this study.
(b) Referring to p. 150, briefly describe the computer system SYSNET that was developed as a result of this study. Also summarize the applications of SYSNET.

(c) Referring to pp. 162–163, describe why the *interactive* aspects of SYSNET proved important.
(d) Referring to p. 163, summarize the outputs from SYSNET.
(e) Referring to pp. 168–172, summarize the various benefits that have resulted from using SYSNET.

2.6-1. Refer to pp. 163–167 of Selected Reference A4 that describes an OR study done for Yellow Freight System, Inc., and the resulting computer system SYSNET.
(a) Briefly describe how the OR team gained the support of upper management for implementing SYSNET.
(b) Briefly describe the implementation strategy that was developed.
(c) Briefly describe the field implementation.
(d) Briefly describe how management incentives and enforcement were used in implementing SYSNET.

2.6-2. Read Selected Reference A5 that describes an OR study done for IBM and the resulting computer system Optimizer.
(a) Summarize the background that led to undertaking this study.
(b) List the complicating factors that the OR team members faced when they started developing a model and a solution algorithm.
(c) Briefly describe the preimplementation test of Optimizer.
(d) Briefly describe the field implementation test.
(e) Briefly describe national implementation.
(f) List the various tangible and intangible benefits that resulted from the study.

2.7-1. From the bottom part of the selected references given at the end of the chapter, select one of these award-winning applications of the OR modeling approach (excluding any that have been assigned for other problems). Read this article and then write a two-page summary of the application and the benefits (including nonfinancial benefits) it provided.

2.7-2. From the bottom part of the selected references given at the end of the chapter, select three of these award-winning applications of the OR modeling approach (excluding any that have been assigned for other problems). For each one, read this article and write a one-page summary of the application and the benefits (including nonfinancial benefits) it provided.

2.7-3. Read Selected Reference 4. The author describes 13 detailed phases of any OR study that develops and applies a computer-based model, whereas this chapter describes six broader phases. For each of these broader phases, list the detailed phases that fall partially or primarily within the broader phase.

CHAPTER 3

Introduction to Linear Programming

The development of linear programming has been ranked among the most important scientific advances of the mid-20th century, and we must agree with this assessment. Its impact since just 1950 has been extraordinary. Today it is a standard tool that has saved many thousands or millions of dollars for many companies or businesses of even moderate size in the various industrialized countries of the world, and its use in other sectors of society has been spreading rapidly. A major proportion of all scientific computation on computers is devoted to the use of linear programming. Dozens of textbooks have been written about linear programming, and *published* articles describing important applications now number in the hundreds.

What is the nature of this remarkable tool, and what kinds of problems does it address? You will gain insight into this topic as you work through subsequent examples. However, a verbal summary may help provide perspective. Briefly, the most common type of application involves the general problem of allocating *limited resources* among *competing activities* in a best possible (i.e., *optimal*) way. More precisely, this problem involves selecting the level of certain activities that compete for scarce resources that are necessary to perform those activities. The choice of activity levels then dictates how much of each resource will be consumed by each activity. The variety of situations to which this description applies is diverse, indeed, ranging from the allocation of production facilities to products to the allocation of national resources to domestic needs, from portfolio selection to the selection of shipping patterns, from agricultural planning to the design of radiation therapy, and so on. However, the one common ingredient in each of these situations is the necessity for allocating resources to activities by choosing the levels of those activities.

Linear programming uses a mathematical model to describe the problem of concern. The adjective *linear* means that all the mathematical functions in this model are required to be *linear functions.* The word *programming* does not refer here to computer programming; rather, it is essentially a synonym for *planning.* Thus, linear programming involves the *planning of activities* to obtain an optimal result, i.e., a result that reaches the specified goal best (according to the mathematical model) among all feasible alternatives.

Although allocating resources to activities is the most common type of application, linear programming has numerous other important applications as well. In fact, *any* problem whose mathematical model fits the very general format for the linear programming model is a linear programming problem. (For this reason, a linear programming problem and its model often are referred to interchangeably as simply a *linear program,* or even as just an *LP.*) Furthermore, a remarkably efficient solution procedure, called the **simplex method,** is available for solving linear programming problems of even enormous size. These are some of the reasons for the tremendous impact of linear programming in recent decades.

Because of its great importance, we devote this and the next six chapters specifically to linear programming. After this chapter introduces the general features of linear programming, Chaps. 4 and 5 focus on the simplex method. Chapter 6 discusses the further analysis of linear programming problems *after* the simplex method has been initially applied. Chapter 7 presents several widely used extensions of the simplex method and introduces an *interior-point algorithm* that sometimes can be used to solve even larger linear programming problems than the simplex method can handle. Chapters 8 and 9 consider some special types of linear programming problems whose importance warrants individual study.

You also can look forward to seeing applications of linear programming to other areas of operations research (OR) in several later chapters.

We begin this chapter by developing a miniature prototype example of a linear programming problem. This example is small enough to be solved graphically in a straightforward way. Sections 3.2 and 3.3 present the general *linear programming model* and its basic assumptions. Section 3.4 gives some additional examples of linear programming applications. Section 3.5 describes how linear programming models of modest size can be conveniently displayed and solved on a spreadsheet. However, some linear programming problems encountered in practice require truly *massive* models. Section 3.6 illustrates how a massive model can arise and how it can still be formulated successfully with the help of a special modeling language such as MPL (its formulation is described in this section) or LINGO (its formulation of this model is presented in Supplement 2 to this chapter on the book's website).

3.1 PROTOTYPE EXAMPLE

The WYNDOR GLASS CO. produces high-quality glass products, including windows and glass doors. It has three plants. Aluminum frames and hardware are made in Plant 1, wood frames are made in Plant 2, and Plant 3 produces the glass and assembles the products.

Because of declining earnings, top management has decided to revamp the company's product line. Unprofitable products are being discontinued, releasing production capacity to launch two new products having large sales potential:

Product 1: An 8-foot glass door with aluminum framing
Product 2: A 4 × 6 foot double-hung wood-framed window

Product 1 requires some of the production capacity in Plants 1 and 3, but none in Plant 2. Product 2 needs only Plants 2 and 3. The marketing division has concluded that the company could sell as much of either product as could be produced by these plants. However, because both products would be competing for the same production capacity in Plant 3, it is not clear which *mix* of the two products would be *most profitable.* Therefore, an OR team has been formed to study this question.

The OR team began by having discussions with upper management to identify management's objectives for the study. These discussions led to developing the following definition of the problem:

> Determine what the *production rates* should be for the two products in order to *maximize their total profit,* subject to the restrictions imposed by the limited production capacities available in the three plants. (Each product will be produced in batches of 20, so the

An Application Vignette

Swift & Company is a diversified protein-producing business based in Greeley, Colorado. With annual sales of over $8 billion, beef and related products are by far the largest portion of the company's business.

To improve the company's sales and manufacturing performance, upper management concluded that it needed to achieve three major objectives. One was to enable the company's customer service representatives to talk to their more than 8,000 customers with accurate information about the availability of current and future inventory while considering requested delivery dates and maximum product age upon delivery. A second was to produce an efficient shift-level schedule for each plant over a 28-day horizon. A third was to accurately determine whether a plant can ship a requested order-line-item quantity on the requested date and time given the availability of cattle and constraints on the plant's capacity.

To meet these three challenges, an OR team developed an *integrated system of 45 linear programming models* based on three model formulations to dynamically schedule its beef-fabrication operations at five plants in real time as it receives orders. *The total audited benefits realized in the first year* of operation of this system *were* **$12.74 million,** including $12 million due to *optimizing the product mix.* Other benefits include a reduction in orders lost, a reduction in price discounting, and better on-time delivery.

Source: A. Bixby, B. Downs, and M. Self, "A Scheduling and Capable-to-Promise Application for Swift & Company," *Interfaces,* **36**(1): 39–50, Jan.–Feb. 2006. (A link to this article is provided on our website, www.mhhe.com/hillier.)

production rate is defined as the number of batches produced per week.) *Any* combination of production rates that satisfies these restrictions is permitted, including producing none of one product and as much as possible of the other.

The OR team also identified the data that needed to be gathered:

1. Number of hours of production time available per week in each plant for these new products. (Most of the time in these plants already is committed to current products, so the available capacity for the new products is quite limited.)
2. Number of hours of production time used in each plant for each batch produced of each new product.
3. Profit per batch produced of each new product. (*Profit per batch produced* was chosen as an appropriate measure after the team concluded that the incremental profit from each additional batch produced would be roughly *constant* regardless of the total number of batches produced. Because no substantial costs will be incurred to initiate the production and marketing of these new products, the total profit from each one is approximately this *profit per batch produced* times *the number of batches produced.*)

Obtaining reasonable estimates of these quantities required enlisting the help of key personnel in various units of the company. Staff in the manufacturing division provided the data in the first category above. Developing estimates for the second category of data required some analysis by the manufacturing engineers involved in designing the production processes for the new products. By analyzing cost data from these same engineers and the marketing division, along with a pricing decision from the marketing division, the accounting department developed estimates for the third category.

Table 3.1 summarizes the data gathered.

The OR team immediately recognized that this was a linear programming problem of the classic **product mix** type, and the team next undertook the formulation of the corresponding mathematical model.

TABLE 3.1 Data for the Wyndor Glass Co. problem

	Production Time per Batch, Hours		
	Product		
Plant	**1**	**2**	**Production Time Available per Week, Hours**
1	1	0	4
2	0	2	12
3	3	2	18
Profit per batch	\$3,000	\$5,000	

Formulation as a Linear Programming Problem

The definition of the problem given above indicates that the decisions to be made are the number of batches of the respective products to be produced per week so as to maximize their total profit. Therefore, to formulate the mathematical (linear programming) model for this problem, let

x_1 = number of batches of product 1 produced per week

x_2 = number of batches of product 2 produced per week

Z = total profit per week (in thousands of dollars) from producing these two products

Thus, x_1 and x_2 are the *decision variables* for the model. Using the bottom row of Table 3.1, we obtain

$$Z = 3x_1 + 5x_2.$$

The objective is to choose the values of x_1 and x_2 so as to *maximize* $Z = 3x_1 + 5x_2$, subject to the restrictions imposed on their values by the limited production capacities available in the three plants. Table 3.1 indicates that each batch of product 1 produced per week uses 1 hour of production time per week in Plant 1, whereas only 4 hours per week are available. This restriction is expressed mathematically by the inequality $x_1 \leq 4$. Similarly, Plant 2 imposes the restriction that $2x_2 \leq 12$. The number of hours of production time used per week in Plant 3 by choosing x_1 and x_2 as the new products' production rates would be $3x_1 + 2x_2$. Therefore, the mathematical statement of the Plant 3 restriction is $3x_1 + 2x_2 \leq 18$. Finally, since production rates cannot be negative, it is necessary to restrict the decision variables to be nonnegative: $x_1 \geq 0$ and $x_2 \geq 0$.

To summarize, in the mathematical language of linear programming, the problem is to choose values of x_1 and x_2 so as to

$$\text{Maximize} \quad Z = 3x_1 + 5x_2,$$

subject to the restrictions

$$\begin{aligned} x_1 \qquad &\leq 4 \\ 2x_2 &\leq 12 \\ 3x_1 + 2x_2 &\leq 18 \end{aligned}$$

and

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(Notice how the layout of the coefficients of x_1 and x_2 in this linear programming model essentially duplicates the information summarized in Table 3.1.)

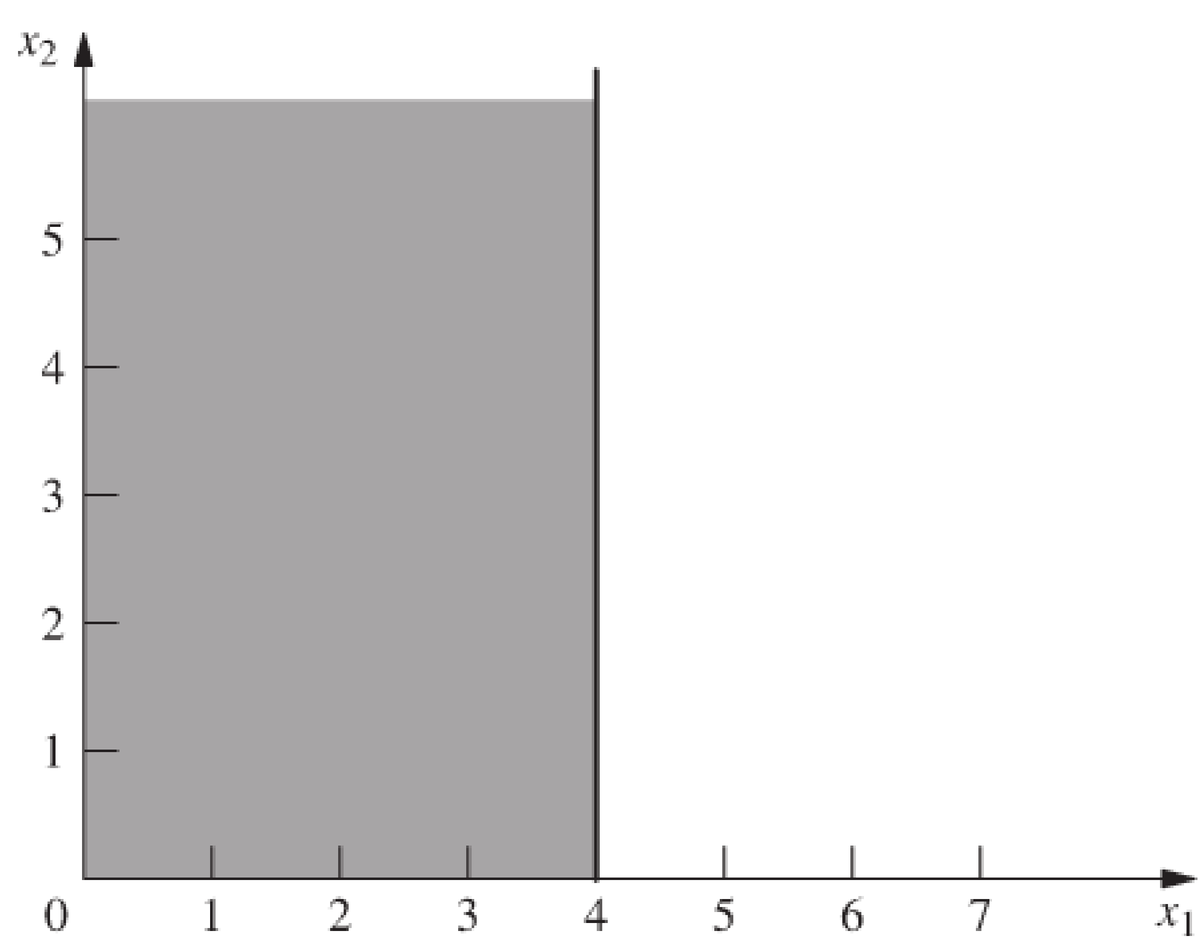
Graphical Solution

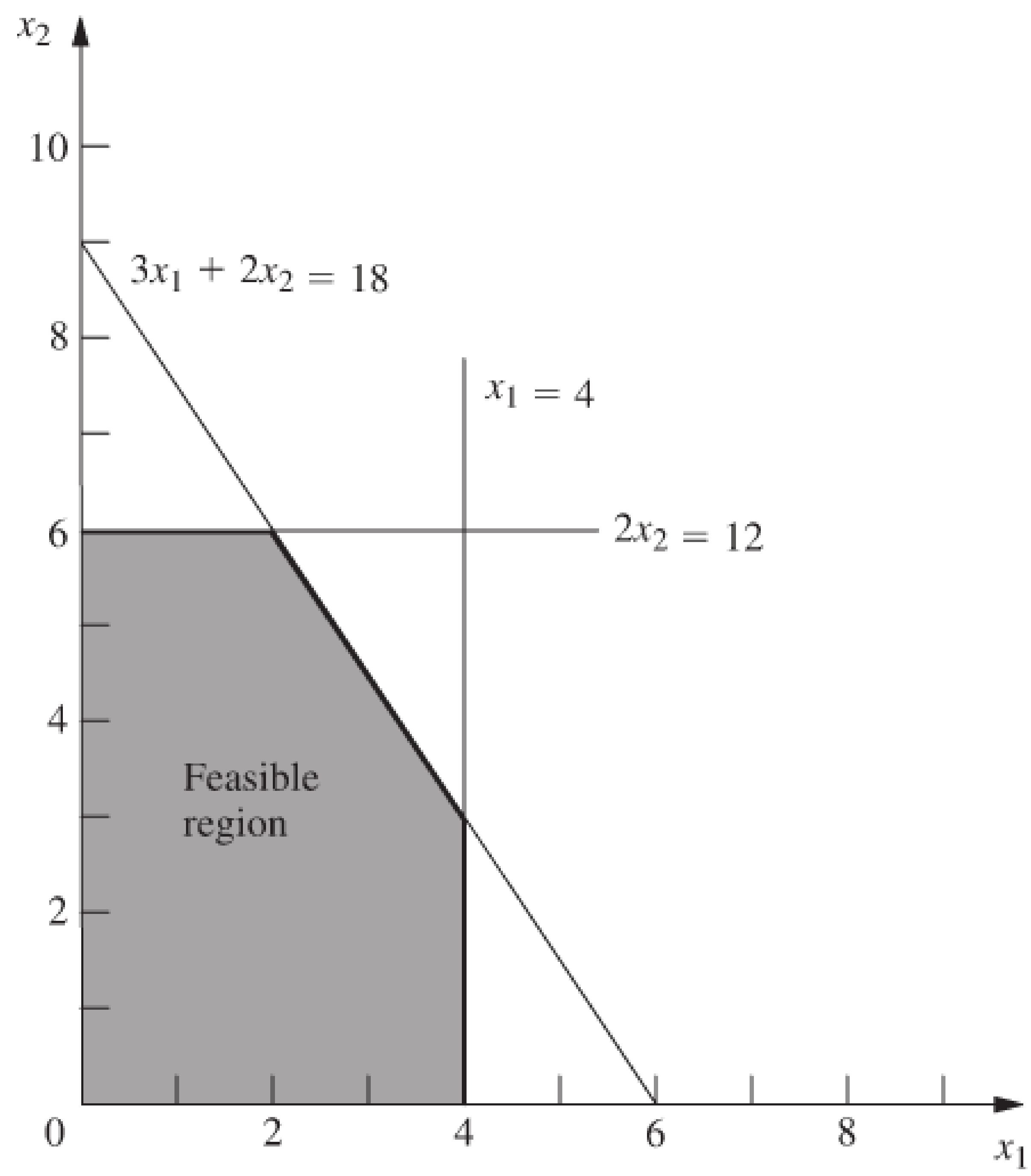
This very small problem has only two decision variables and therefore only two dimensions, so a graphical procedure can be used to solve it. This procedure involves constructing a two-dimensional graph with x_1 and x_2 as the axes. The first step is to identify the values of (x_1, x_2) that are permitted by the restrictions. This is done by drawing each line that borders the range of permissible values for one restriction. To begin, note that the nonnegativity restrictions $x_1 \geq 0$ and $x_2 \geq 0$ require (x_1, x_2) to lie on the *positive* side of the axes (including actually *on* either axis), i.e., in the first quadrant. Next, observe that the restriction $x_1 \leq 4$ means that (x_1, x_2) cannot lie to the right of the line $x_1 = 4$. These results are shown in Fig. 3.1, where the shaded area contains the only values of (x_1, x_2) that are still allowed.

In a similar fashion, the restriction $2x_2 \leq 12$ (or, equivalently, $x_2 \leq 6$) implies that the line $2x_2 = 12$ should be added to the boundary of the permissible region. The final restriction, $3x_1 + 2x_2 \leq 18$, requires plotting the points (x_1, x_2) such that $3x_1 + 2x_2 = 18$ (another line) to complete the boundary. (Note that the points such that $3x_1 + 2x_2 \leq 18$ are those that lie either underneath or on the line $3x_1 + 2x_2 = 18$, so this is the limiting line above which points do not satisfy the inequality.) The resulting region of permissible values of (x_1, x_2), called the **feasible region,** is shown in Fig. 3.2. (The demo called *Graphical Method* in your OR Tutor provides a more detailed example of constructing a feasible region.)

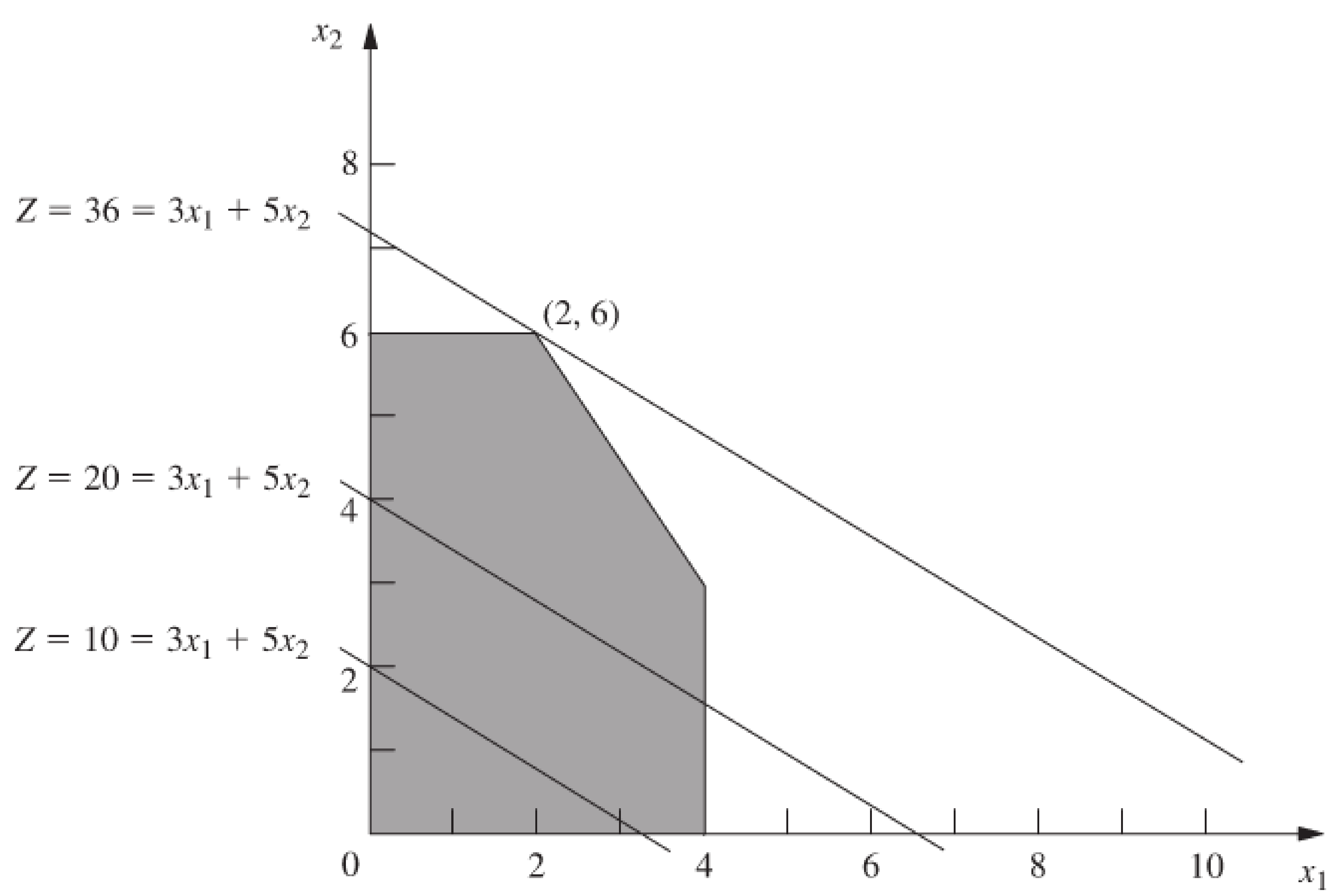
The final step is to pick out the point in this feasible region that maximizes the value of $Z = 3x_1 + 5x_2$. To discover how to perform this step efficiently, begin by trial and error. Try, for example, $Z = 10 = 3x_1 + 5x_2$ to see if there are in the permissible region any values of (x_1, x_2) that yield a value of Z as large as 10. By drawing the line $3x_1 + 5x_2 = 10$ (see Fig. 3.3), you can see that there are many points on this line that lie within the region. Having gained perspective by trying this arbitrarily chosen value of $Z = 10$, you should next try a larger arbitrary value of Z, say, $Z = 20 = 3x_1 + 5x_2$. Again, Fig. 3.3 reveals that a segment of the line $3x_1 + 5x_2 = 20$ lies within the region, so that the maximum permissible value of Z must be at least 20.

FIGURE 3.1
Shaded area shows values of (x_1, x_2) allowed by $x_1 \geq 0$, $x_2 \geq 0$, $x_1 \leq 4$.





■ **FIGURE 3.2**
Shaded area shows the set of permissible values of (x_1, x_2), called the feasible region.



■ **FIGURE 3.3**
The value of (x_1, x_2) that maximizes $3x_1 + 5x_2$ is (2, 6).

Now notice in Fig. 3.3 that the two lines just constructed are parallel. This is no coincidence, since *any* line constructed in this way has the form $Z = 3x_1 + 5x_2$ for the chosen value of Z, which implies that $5x_2 = -3x_1 + Z$ or, equivalently,

$$x_2 = -\frac{3}{5}x_1 + \frac{1}{5}Z$$

This last equation, called the **slope-intercept form** of the objective function, demonstrates that the *slope* of the line is $-\frac{3}{5}$ (since each unit increase in x_1 changes x_2 by $-\frac{3}{5}$), whereas the *intercept* of the line with the x_2 axis is $\frac{1}{5}Z$ (since $x_2 = \frac{1}{5}Z$ when $x_1 = 0$). The fact that the slope is fixed at $-\frac{3}{5}$ means that *all* lines constructed in this way are parallel.

Again, comparing the $10 = 3x_1 + 5x_2$ and $20 = 3x_1 + 5x_2$ lines in Fig. 3.3, we note that the line giving a larger value of Z ($Z = 20$) is farther up and away from the origin than the other line ($Z = 10$). This fact also is implied by the slope-intercept form of the objective function, which indicates that the intercept with the x_1 axis $(\frac{1}{5}Z)$ increases when the value chosen for Z is increased.

These observations imply that our trial-and-error procedure for constructing lines in Fig. 3.3 involves nothing more than drawing a family of parallel lines containing at least one point in the feasible region and selecting the line that corresponds to the largest value of Z. Figure 3.3 shows that this line passes through the point (2, 6), indicating that the **optimal solution** is $x_1 = 2$ and $x_2 = 6$. The equation of this line is $3x_1 + 5x_2 = 3(2) + 5(6) = 36 = Z$, indicating that the optimal value of Z is $Z = 36$. The point (2, 6) lies at the intersection of the two lines $2x_2 = 12$ and $3x_1 + 2x_2 = 18$, shown in Fig. 3.2, so that this point can be calculated algebraically as the simultaneous solution of these two equations.

Having seen the trial-and-error procedure for finding the optimal point (2, 6), you now can streamline this approach for other problems. Rather than draw several parallel lines, it is sufficient to form a single line with a ruler to establish the slope. Then move the ruler with fixed slope through the feasible region in the direction of improving Z. (When the objective is to *minimize* Z, move the ruler in the direction that *decreases* Z.) Stop moving the ruler at the last instant that it still passes through a point in this region. This point is the desired *optimal solution.*

This procedure often is referred to as the **graphical method** for linear programming. It can be used to solve any linear programming problem with two decision variables. With considerable difficulty, it is possible to extend the method to three decision variables but not more than three. (The next chapter will focus on the *simplex method* for solving larger problems.)

Conclusions

The OR team used this approach to find that the optimal solution is $x_1 = 2$, $x_2 = 6$, with $Z = 36$. This solution indicates that the Wyndor Glass Co. should produce products 1 and 2 at the rate of 2 batches per week and 6 batches per week, respectively, with a resulting total profit of \$36,000 per week. No other mix of the two products would be so profitable—*according to the model.*

However, we emphasized in Chap. 2 that well-conducted OR studies do not simply find *one* solution for the *initial* model formulated and then stop. All six phases described in Chap. 2 are important, including thorough testing of the model (see Sec. 2.4) and postoptimality analysis (see Sec. 2.3).

In full recognition of these practical realities, the OR team now is ready to evaluate the validity of the model more critically (to be continued in Sec. 3.3) and to perform sensitivity analysis on the effect of the estimates in Table 3.1 being different because of inaccurate estimation, changes of circumstances, etc. (to be continued in Sec. 6.7).

Continuing the Learning Process with Your OR Courseware

This is the first of many points in the book where you may find it helpful to use your *OR Courseware* on the book's website. A key part of this courseware is a program called **OR Tutor.** This program includes a complete demonstration example of the *graphical method* introduced in this section. To provide you with **another example** of a model formulation as well, this demonstration begins by introducing a problem and formulating a linear programming model for the problem before then applying the graphical method step by step to solve the model. Like the many other demonstration examples accompanying other sections of the book, this computer demonstration highlights concepts that are difficult to convey on the printed page. You may refer to Appendix 1 for documentation of the software.

If you would like to see still **more examples,** you can go to the **Worked Examples** section of the book's website. This section includes a few examples with complete solutions for almost every chapter as a supplement to the examples in the book and in OR Tutor. The examples for the current chapter begin with a relatively straightforward problem that involves formulating a small linear programming model and applying the graphical method. The subsequent examples become progressively more challenging.

Another key part of your OR Courseware is a program called **IOR Tutorial.** This program features many interactive procedures for interactively executing various solution methods presented in the book, which enables you to focus on learning and executing the logic of the method efficiently while the computer does the number crunching. Included is an interactive procedure for applying the graphical method for linear programming. Once you get the hang of it, a second procedure enables you to quickly apply the graphical method for performing sensitivity analysis on the effect of revising the data of the problem. You then can print out your work and results for your homework. Like the other procedures in IOR Tutorial, these procedures are designed specifically to provide you with an efficient, enjoyable, and enlightening learning experience while you do your homework.

When you formulate a linear programming model with more than two decision variables (so the graphical method cannot be used), the *simplex method* described in Chap. 4 enables you to still find an optimal solution immediately. Doing so also is helpful for *model validation,* since finding a *nonsensical* optimal solution signals that you have made a mistake in formulating the model.

We mentioned in Sec. 1.4 that your OR Courseware introduces you to three particularly popular commercial software packages—the Excel Solver, LINGO/LINDO, and MPL/CPLEX—for solving a variety of OR models. All three packages include the simplex method for solving linear programming models. Section 3.5 describes how to use Excel to formulate and solve linear programming models in a spreadsheet format. Descriptions of the other packages are provided in Sec. 3.6 (MPL and LINGO), Supplements 1 and 2 to this chapter on the book's website (LINGO), Sec. 4.8 (CPLEX and LINDO), and Appendix 4.1 (LINGO and LINDO). MPL, LINGO, and LINDO tutorials also are provided on the book's website. In addition, your OR Courseware includes a file for each of the three packages showing how it can be used to solve each of the examples in this chapter.

3.2 THE LINEAR PROGRAMMING MODEL

The Wyndor Glass Co. problem is intended to illustrate a typical linear programming problem (miniature version). However, linear programming is too versatile to be completely characterized by a single example. In this section we discuss the general characteristics of linear programming problems, including the various legitimate forms of the mathematical model for linear programming.

Let us begin with some basic terminology and notation. The first column of Table 3.2 summarizes the components of the Wyndor Glass Co. problem. The second column then introduces more general terms for these same components that will fit many linear programming problems. The key terms are *resources* and *activities,* where m denotes the number of different kinds of resources that can be used and n denotes the number of activities being considered. Some typical resources are money and particular kinds of machines,

TABLE 3.2 Common terminology for linear programming

Prototype Example	General Problem
Production capacities of plants	Resources
3 plants	m resources
Production of products	Activities
2 products	n activities
Production rate of product j, x_j	Level of activity j, x_j
Profit Z	Overall measure of performance Z

equipment, vehicles, and personnel. Examples of activities include investing in particular projects, advertising in particular media, and shipping goods from a particular source to a particular destination. In any application of linear programming, all the activities may be of one general kind (such as any one of these three examples), and then the individual activities would be particular alternatives within this general category.

As described in the introduction to this chapter, the most common type of application of linear programming involves allocating resources to activities. The amount available of each resource is limited, so a careful allocation of resources to activities must be made. Determining this allocation involves choosing the *levels* of the activities that achieve the best possible value of the *overall measure of performance.*

Certain symbols are commonly used to denote the various components of a linear programming model. These symbols are listed below, along with their interpretation for the general problem of allocating resources to activities.

$Z =$ value of overall measure of performance.

$x_j =$ level of activity j (for $j = 1, 2, \dots, n$).

$c_j =$ increase in Z that would result from each unit increase in level of activity j.

$b_i =$ amount of resource i that is available for allocation to activities (for $i = 1, 2, \dots, m$).

$a_{ij} =$ amount of resource i consumed by each unit of activity j.

The model poses the problem in terms of making decisions about the levels of the activities, so $x_1, x_2, \dots, x_n$ are called the **decision variables.** As summarized in Table 3.3, the

TABLE 3.3 Data needed for a linear programming model involving the allocation of resources to activities

	Resource Usage per Unit of Activity				
	Activity				
Resource	**1**	**2**	**. . .**	***n***	**Amount of Resource Available**
1	a_{11}	a_{12}	. . .	a_{1n}	b_1
2	a_{21}	a_{22}	. . .	a_{2n}	b_2
.					.
.	. . .	. . .	. . .	. . .	.
.					.
m	a_{m1}	a_{m2}	. . .	a_{mn}	b_m
Contribution to Z per unit of activity	c_1	c_2	. . .	c_n	

values of c_j, b_i, and a_{ij} (for $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$) are the *input constants* for the model. The c_j, b_i, and a_{ij} are also referred to as the **parameters** of the model.

Notice the correspondence between Table 3.3 and Table 3.1.

A Standard Form of the Model

Proceeding as for the Wyndor Glass Co. problem, we can now formulate the mathematical model for this general problem of allocating resources to activities. In particular, this model is to select the values for $x_1, x_2, \ldots, x_n$ so as to

$$\text{Maximize} \quad Z = c_1x_1 + c_2x_2 + \cdots + c_nx_n,$$

subject to the restrictions

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &\le b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &\le b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &\le b_m, \end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad \ldots, \quad x_n \ge 0.$$

We call this *our standard form*[1] for the linear programming problem. Any situation whose mathematical formulation fits this model is a linear programming problem.

Notice that the model for the Wyndor Glass Co. problem fits our standard form, with $m = 3$ and $n = 2$.

Common terminology for the linear programming model can now be summarized. The function being maximized, $c_1x_1 + c_2x_2 + \cdots + c_nx_n$, is called the **objective function.** The restrictions normally are referred to as **constraints.** The first m constraints (those with a *function* of all the variables $a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n$ on the left-hand side) are sometimes called **functional constraints** (or *structural constraints*). Similarly, the $x_j \ge 0$ restrictions are called **nonnegativity constraints** (or *nonnegativity conditions*).

Other Forms

We now hasten to add that the preceding model does not actually fit the natural form of some linear programming problems. The other *legitimate forms* are the following:

1. Minimizing rather than maximizing the objective function:

$$\text{Minimize} \quad Z = c_1x_1 + c_2x_2 + \cdots + c_nx_n.$$

2. Some functional constraints with a greater-than-or-equal-to inequality:

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \ge b_i \qquad \text{for some values of } i.$$

3. Some functional constraints in equation form:

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n = b_i \qquad \text{for some values of } i.$$

4. Deleting the nonnegativity constraints for some decision variables:

$$x_j \text{ unrestricted in sign} \qquad \text{for some values of } j.$$

Any problem that mixes some of or all these forms with the remaining parts of the preceding model is still a linear programming problem. Our interpretation of the words *allocating*

[1]This is called *our* standard form rather than *the* standard form because some textbooks adopt other forms.

limited resources among competing activities may no longer apply very well, if at all; but regardless of the interpretation or context, all that is required is that the mathematical statement of the problem fit the allowable forms. Thus, the concise definition of a linear programming problem is that each component of its model fits either the standard form or one of the other legitimate forms listed above.

Terminology for Solutions of the Model

You may be used to having the term *solution* mean the final answer to a problem, but the convention in linear programming (and its extensions) is quite different. Here, *any* specification of values for the decision variables $(x_1, x_2, \ldots, x_n)$ is called a **solution,** regardless of whether it is a desirable or even an allowable choice. Different types of solutions are then identified by using an appropriate adjective.

A **feasible solution** is a solution for which *all* the constraints are *satisfied.*
An **infeasible solution** is a solution for which *at least one* constraint is *violated.*

In the example, the points (2, 3) and (4, 1) in Fig. 3.2 are *feasible solutions,* while the points $(-1, 3)$ and (4, 4) are *infeasible solutions.*

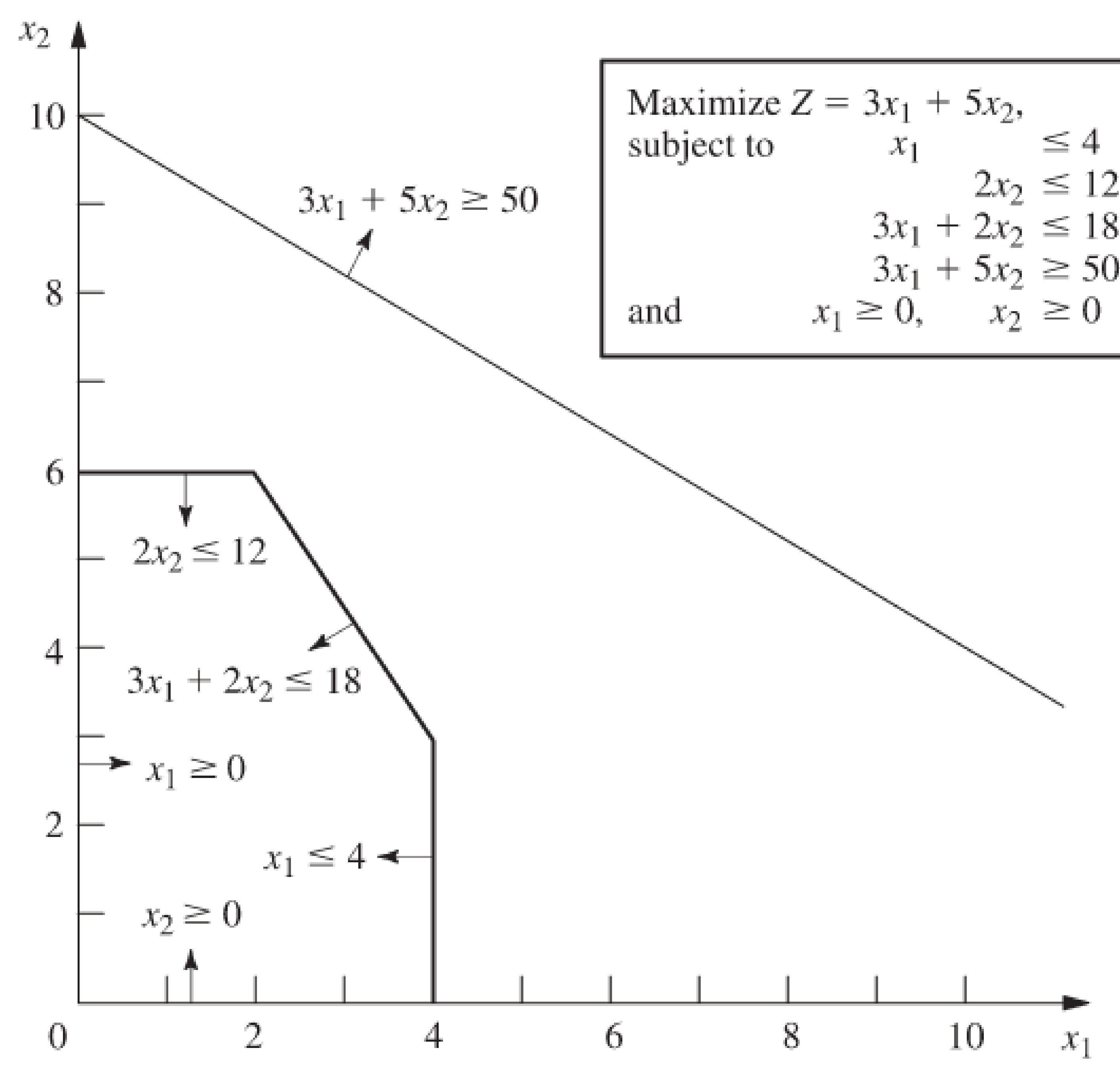
The **feasible region** is the collection of all feasible solutions.

The feasible region in the example is the entire shaded area in Fig. 3.2.

It is possible for a problem to have **no feasible solutions.** This would have happened in the example if the new products had been required to return a net profit of at least \$50,000 per week to justify discontinuing part of the current product line. The corresponding constraint, $3x_1 + 5x_2 \geq 50$, would eliminate the entire feasible region, so no mix of new products would be superior to the status quo. This case is illustrated in Fig. 3.4.

Given that there are feasible solutions, the goal of linear programming is to find a best feasible solution, as measured by the value of the objective function in the model.

■ **FIGURE 3.4**
The Wyndor Glass Co. problem would have no feasible solutions if the constraint $3x_1 + 5x_2 \geq 50$ were added to the problem.



An **optimal solution** is a feasible solution that has the *most favorable value* of the objective function.

The **most favorable value** is the *largest value* if the objective function is to be *maximized,* whereas it is the *smallest value* if the objective function is to be *minimized.*

Most problems will have just one optimal solution. However, it is possible to have more than one. This would occur in the example if the *profit per batch produced* of product 2 were changed to \$2,000. This changes the objective function to $Z = 3x_1 + 2x_2$, so that all the points on the line segment connecting (2, 6) and (4, 3) would be optimal. This case is illustrated in Fig. 3.5. As in this case, *any* problem having **multiple optimal solutions** will have an *infinite* number of them, each with the same optimal value of the objective function.

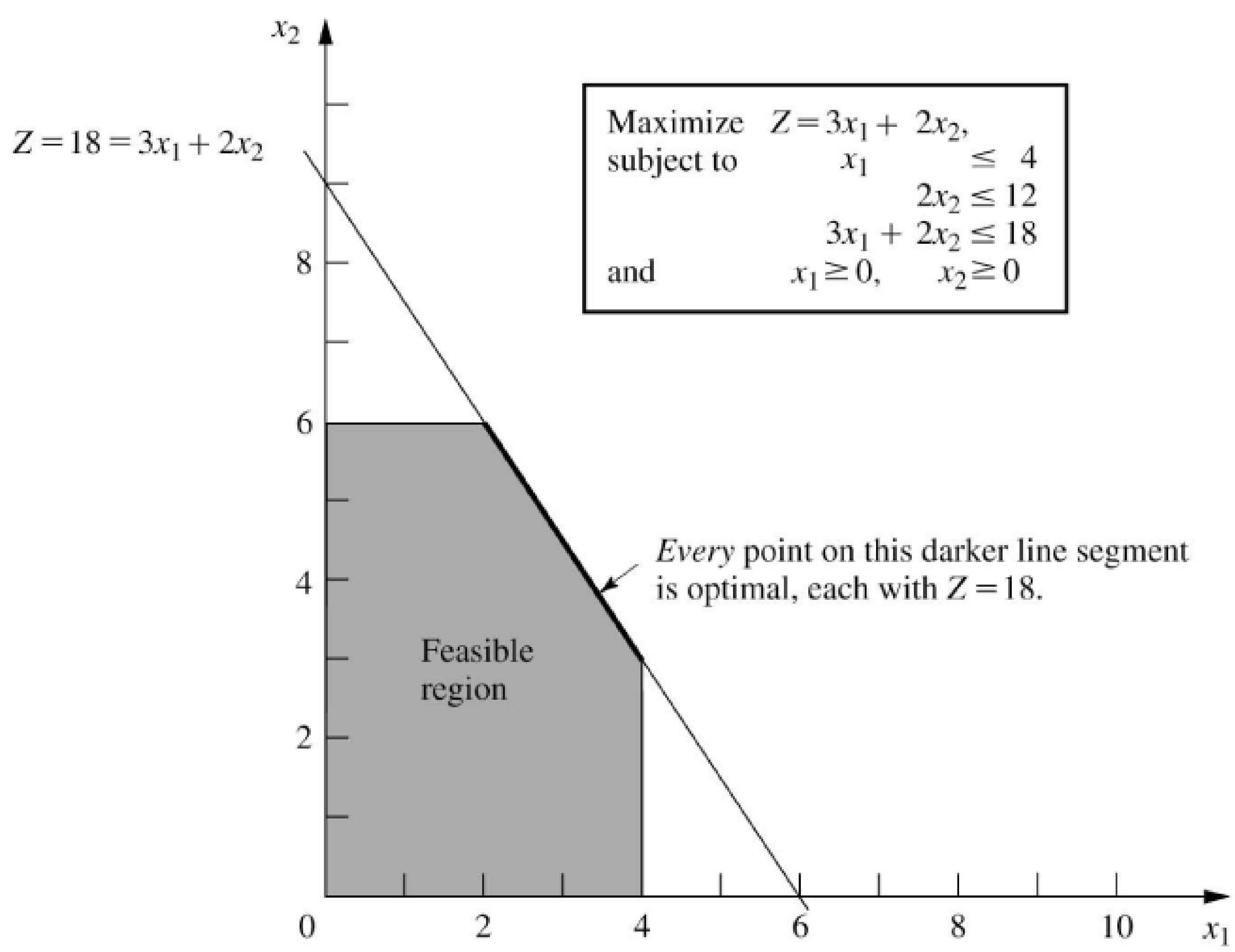
Another possibility is that a problem has **no optimal solutions.** This occurs only if (1) it has no feasible solutions or (2) the constraints do not prevent improving the value of the objective function (Z) indefinitely in the favorable direction (positive or negative). The latter case is referred to as having an **unbounded Z** or an *unbounded objective.* To illustrate, this case would result if the last two functional constraints were mistakenly deleted in the example, as illustrated in Fig. 3.6.

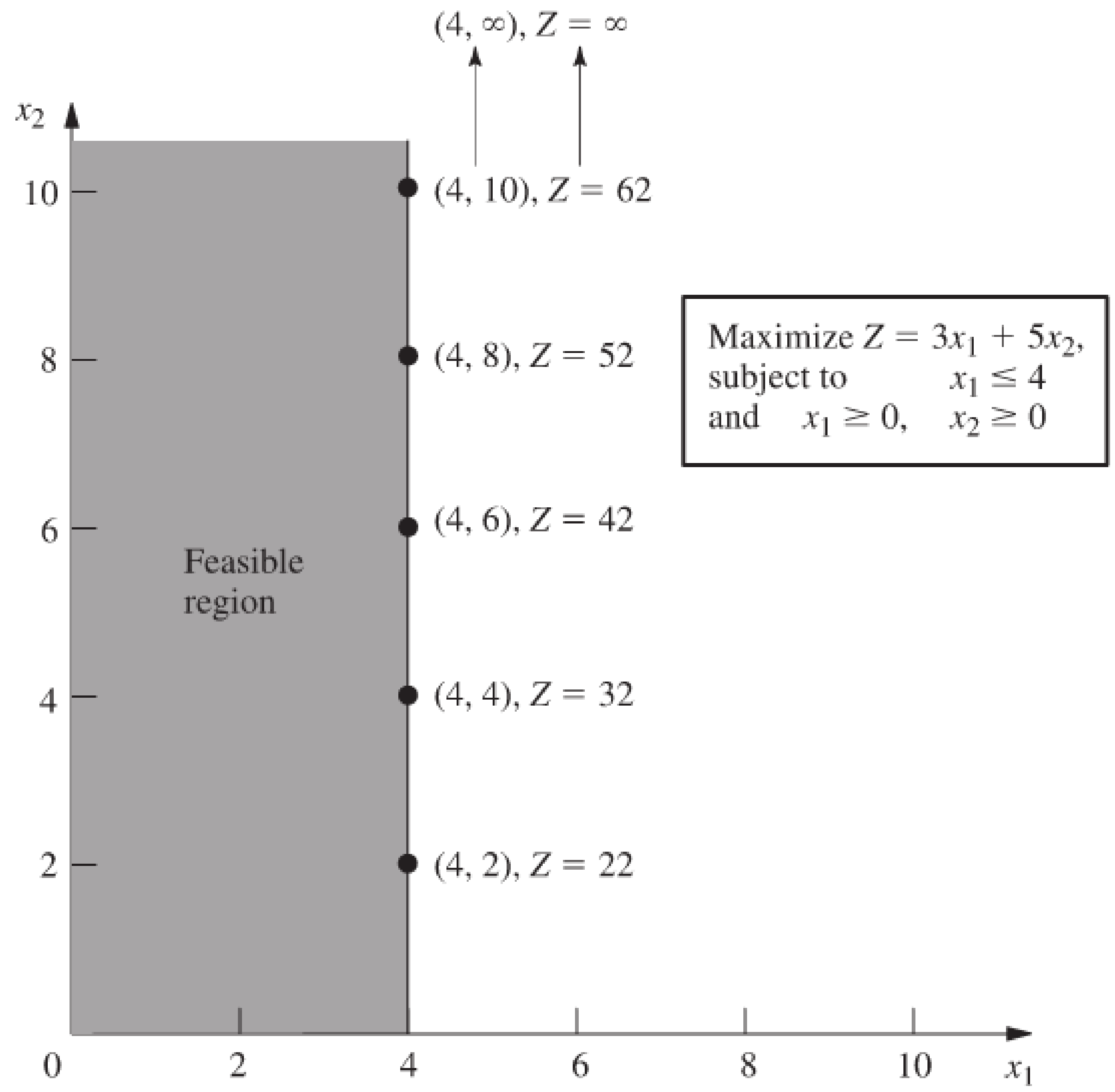
We next introduce a special type of feasible solution that plays the key role when the simplex method searches for an optimal solution.

A **corner-point feasible (CPF) solution** is a solution that lies at a corner of the feasible region.

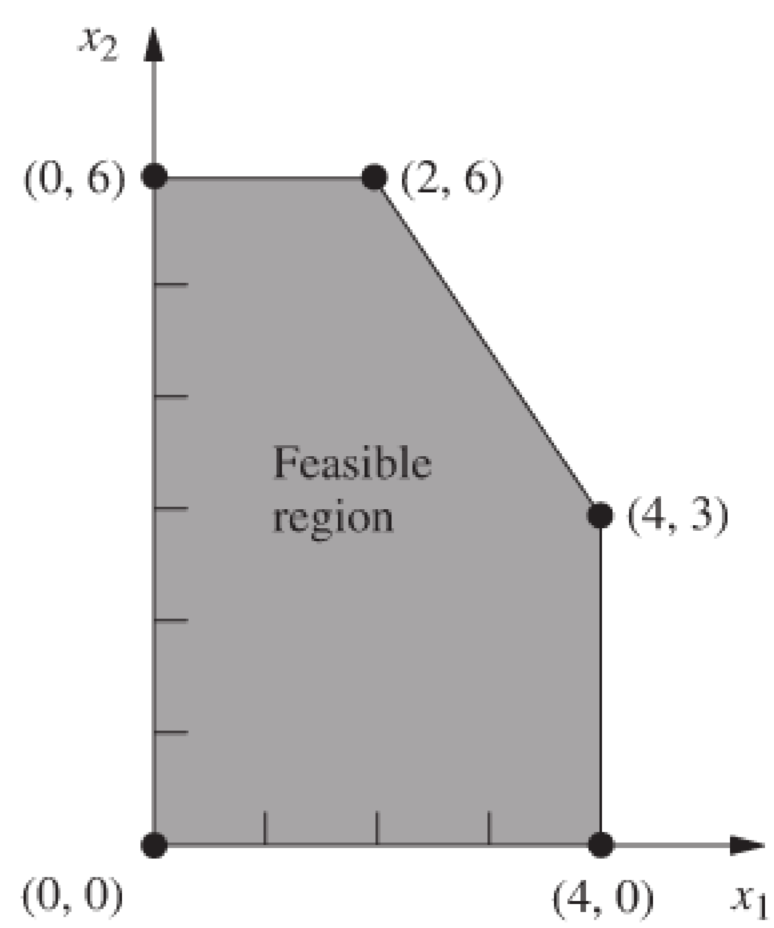
(CPF solutions also are commonly referred to as *extreme points* or *vertices,* but we prefer the more suggestive *corner-point* terminology.) Figure 3.7 highlights the five CPF solutions for the example.

FIGURE 3.5
The Wyndor Glass Co. problem would have multiple optimal solutions if the objective function were changed to $Z = 3x_1 + 2x_2$.





■ **FIGURE 3.6**
The Wyndor Glass Co. problem would have no optimal solutions if the only functional constraint were $x_1 \le 4$, because x_2 then could be increased indefinitely in the feasible region without ever reaching the maximum value of $Z = 3x_1 + 5x_2$.



■ **FIGURE 3.7**
The five dots are the five CPF solutions for the Wyndor Glass Co. problem.

Sections 4.1 and 5.1 will delve into the various useful properties of CPF solutions for problems of any size, including the following relationship with optimal solutions.

> **Relationship between optimal solutions and CPF solutions:** Consider any linear programming problem with feasible solutions and a bounded feasible region. The problem must possess CPF solutions and at least one optimal solution. Furthermore, the best CPF solution *must* be an optimal solution. Thus, if a problem has exactly one optimal solution, it *must* be a CPF solution. If the problem has multiple optimal solutions, at least two *must* be CPF solutions.

The example has exactly one optimal solution, $(x_1, x_2) = (2, 6)$, which is a CPF solution. (Think about how the graphical method leads to the one optimal solution being a CPF solution.) When the example is modified to yield multiple optimal solutions, as shown in Fig. 3.5, two of these optimal solutions—(2, 6) and (4, 3)—are CPF solutions.

3.3 ASSUMPTIONS OF LINEAR PROGRAMMING

All the assumptions of linear programming actually are implicit in the model formulation given in Sec. 3.2. In particular, from a mathematical viewpoint, the assumptions simply are that the model must have a linear objective function subject to linear constraints. However, from a modeling viewpoint, these mathematical properties of a linear programming model imply that certain assumptions must hold about the activities and data of the problem being modeled, including assumptions about the effect of varying the levels of the activities. It is good to highlight these assumptions so you can more easily evaluate how well linear programming applies to any given problem. Furthermore, we still need to see why the OR team for the Wyndor Glass Co. concluded that a linear programming formulation provided a satisfactory representation of the problem.

Proportionality

Proportionality is an assumption about both the objective function and the functional constraints, as summarized below.

> **Proportionality assumption:** The contribution of each activity to the *value of the objective function* Z is *proportional* to the *level of the activity* x_j, as represented by the $c_j x_j$ term in the objective function. Similarly, the contribution of each activity to the *left-hand side of each functional constraint* is *proportional* to the *level of the activity* x_j, as represented by the $a_{ij}x_j$ term in the constraint. Consequently, this assumption rules out any exponent other than 1 for any variable in any term of any function (whether the objective function or the function on the left-hand side of a functional constraint) in a linear programming model.[2]

To illustrate this assumption, consider the first term ($3x_1$) in the objective function ($Z = 3x_1 + 5x_2$) for the Wyndor Glass Co. problem. This term represents the profit generated per week (in thousands of dollars) by producing product 1 at the rate of x_1 batches per week. The *proportionality satisfied* column of Table 3.4 shows the case that was assumed in Sec. 3.1, namely, that this profit is indeed proportional to x_1 so that $3x_1$ is the appropriate term for the objective function. By contrast, the next three columns show different hypothetical cases where the proportionality assumption would be violated.

Refer first to the *Case 1* column in Table 3.4. This case would arise if there were *start-up costs* associated with initiating the production of product 1. For example, there might be costs involved with setting up the production facilities. There might also be costs associated with arranging the distribution of the new product. Because these are one-time costs, they would need to be amortized on a per-week basis to be commensurable with Z (profit in thousands of dollars per week). Suppose that this amortization were done and that the total start-up cost amounted to reducing Z by 1, but that the profit without considering the start-up cost would be $3x_1$. This would mean that the contribution from product 1 to Z should be $3x_1 - 1$ for $x_1 > 0$,

[2]When the function includes any *cross-product terms,* proportionality should be interpreted to mean that *changes* in the function value are proportional to *changes* in each variable (x_j) individually, given any fixed values for all the other variables. Therefore, a cross-product term satisfies proportionality as long as each variable in the term has an exponent of 1 (However, any cross-product term violates the *additivity assumption*, discussed next.)

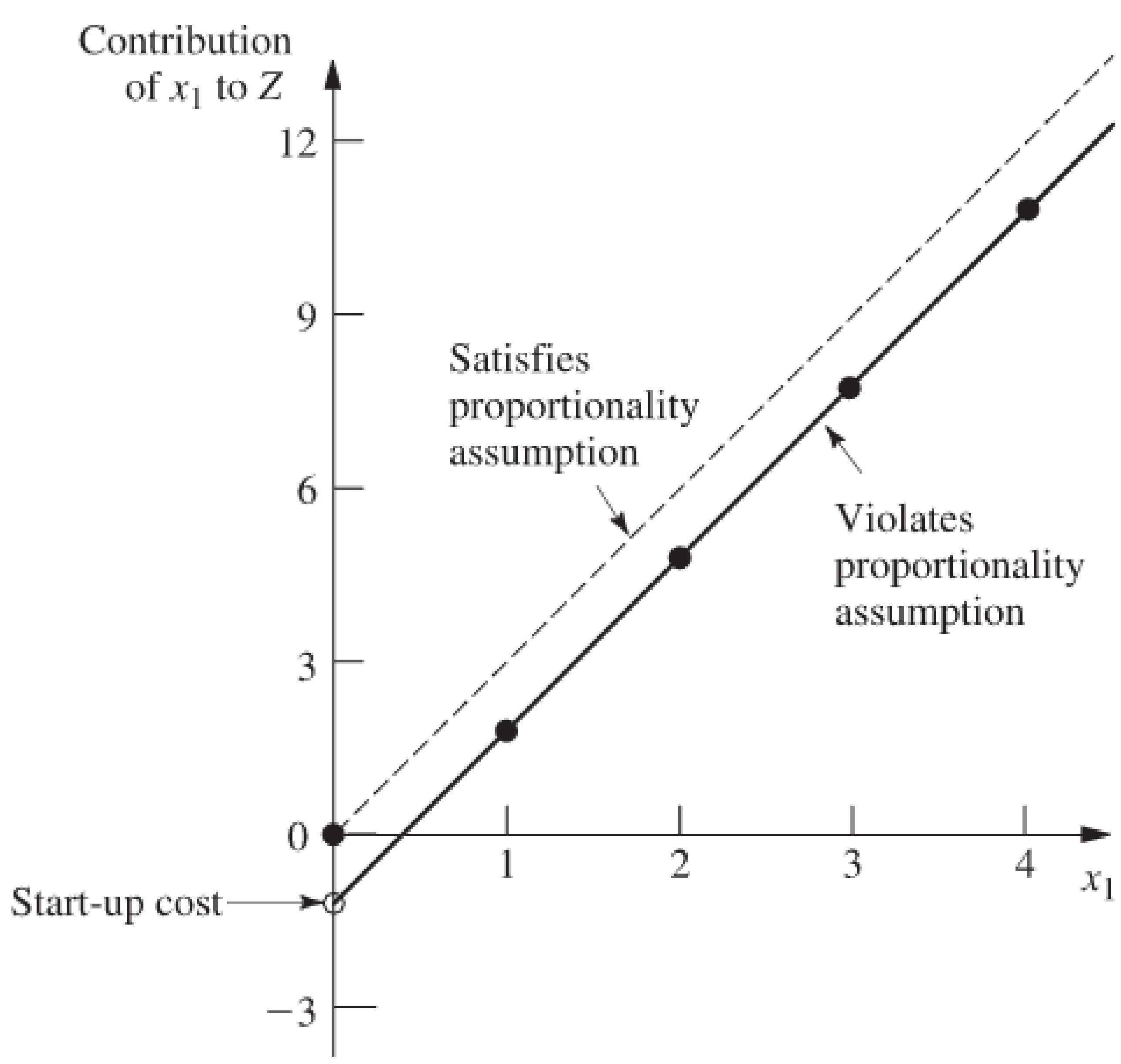
TABLE 3.4 Examples of satisfying or violating proportionality

	Profit from Product 1 ($000 per Week)			
		Proportionality Violated		
x_1	Proportionality Satisfied	Case 1	Case 2	Case 3
0	0	0	0	0
1	3	2	3	3
2	6	5	7	5
3	9	8	12	6
4	12	11	18	6

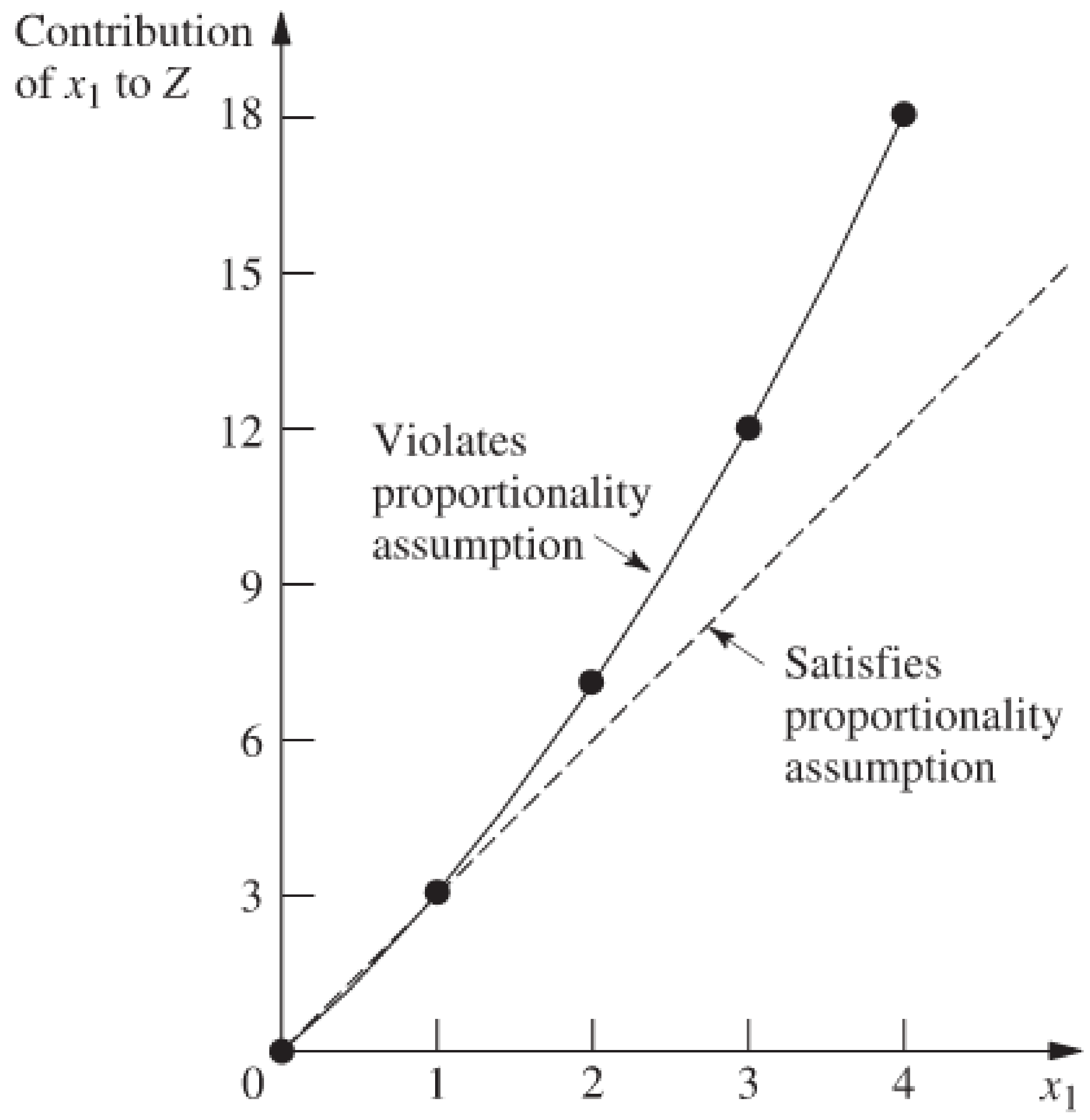
whereas the contribution would be $3x_1 = 0$ when $x_1 = 0$ (no start-up cost). This profit function,[3] which is given by the solid curve in Fig. 3.8, certainly is *not* proportional to x_1.

At first glance, it might appear that *Case 2* in Table 3.4 is quite similar to Case 1. However, Case 2 actually arises in a very different way. There no longer is a start-up cost, and the profit from the first unit of product 1 per week is indeed 3, as originally assumed. However, there now is an *increasing marginal return;* i.e., the *slope* of the *profit function* for product 1 (see the solid curve in Fig. 3.9) keeps increasing as x_1 is increased. This violation of proportionality might occur because of economies of scale that can sometimes be achieved at higher levels of production, e.g., through the use of more efficient high-volume machinery, longer production runs, quantity discounts for large purchases of raw materials, and the learning-curve effect whereby workers become more efficient as they gain experience with a particular mode of production. As the incremental cost goes down, the incremental profit will go up (assuming constant marginal revenue).

FIGURE 3.8
The solid curve violates the proportionality assumption because of the start-up cost that is incurred when x_1 is increased from 0. The values at the dots are given by the Case 1 column of Table 3.4.



[3]If the contribution from product 1 to Z were $3x_1 - 1$ for *all* $x_1 \geq 0$, including $x_1 = 0$, then the fixed constant, -1, could be deleted from the objective function without changing the optimal solution and proportionality would be restored. However, this "fix" does not work here because the -1 constant does not apply when $x_1 = 0$.

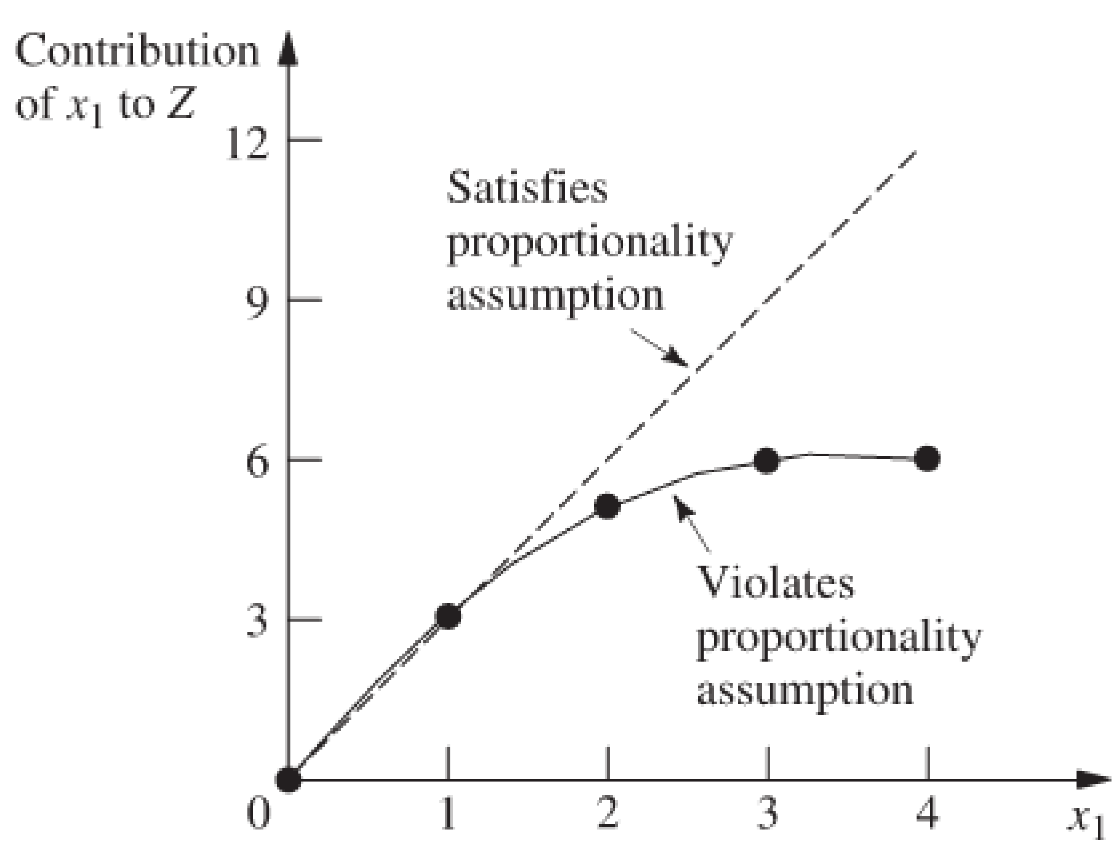


■ **FIGURE 3.9**
The solid curve violates the proportionality assumption because its slope (the *marginal return* from product 1) keeps increasing as x_1 is increased. The values at the dots are given by the Case 2 column of Table 3.4.

Referring again to Table 3.4, the reverse of Case 2 is *Case 3,* where there is a *decreasing marginal return.* In this case, the *slope* of the *profit function* for product 1 (given by the solid curve in Fig. 3.10) keeps decreasing as x_1 is increased. This violation of proportionality might occur because the *marketing costs* need to go up more than proportionally to attain increases in the level of sales. For example, it might be possible to sell product 1 at the rate of 1 per week ($x_1 = 1$) with no advertising, whereas attaining sales to sustain a production rate of $x_1 = 2$ might require a moderate amount of advertising, $x_1 = 3$ might necessitate an extensive advertising campaign, and $x_1 = 4$ might require also lowering the price.

All three cases are hypothetical examples of ways in which the proportionality assumption could be violated. What is the actual situation? The actual profit from producing product 1 (or any other product) is derived from the sales revenue minus various direct and indirect costs. Inevitably, some of these cost components are not strictly proportional to the production rate, perhaps for one of the reasons illustrated above. However, the real question

■ **FIGURE 3.10**
The solid curve violates the proportionality assumption because its slope (the *marginal return* from product 1) keeps decreasing as x_1 is increased. The values at the dots are given by the Case 3 column in Table 3.4.



is whether, after all the components of profit have been accumulated, proportionality is a reasonable approximation for practical modeling purposes. For the Wyndor Glass Co. problem, the OR team checked both the objective function and the functional constraints. The conclusion was that proportionality could indeed be assumed without serious distortion.

For other problems, what happens when the proportionality assumption does not hold even as a reasonable approximation? In most cases, this means you must use *nonlinear programming* instead (presented in Chap. 12). However, we do point out in Sec. 12.8 that a certain important kind of nonproportionality can still be handled by linear programming by reformulating the problem appropriately. Furthermore, if the assumption is violated only because of start-up costs, there is an extension of linear programming (*mixed integer programming*) that can be used, as discussed in Sec. 11.3 (the fixed-charge problem).

Additivity

Although the proportionality assumption rules out exponents other than 1, it does not prohibit *cross-product terms* (terms involving the product of two or more variables). The additivity assumption does rule out this latter possibility, as summarized below.

> **Additivity assumption:** *Every* function in a linear programming model (whether the objective function or the function on the left-hand side of a functional constraint) is the *sum* of the *individual contributions* of the respective activities.

To make this definition more concrete and clarify why we need to worry about this assumption, let us look at some examples. Table 3.5 shows some possible cases for the objective function for the Wyndor Glass Co. problem. In each case, the *individual contributions* from the products are just as assumed in Sec. 3.1, namely, $3x_1$ for product 1 and $5x_2$ for product 2. The difference lies in the last row, which gives the *function value* for Z when the two products are produced jointly. The *additivity satisfied* column shows the case where this *function value* is obtained simply by adding the first two rows ($3 + 5 = 8$), so that $Z = 3x_1 + 5x_2$ as previously assumed. By contrast, the next two columns show hypothetical cases where the additivity assumption would be violated (but not the proportionality assumption).

Referring to the *Case 1* column of Table 3.5, this case corresponds to an objective function of $Z = 3x_1 + 5x_2 + x_1x_2$, so that $Z = 3 + 5 + 1 = 9$ for $(x_1, x_2) = (1, 1)$, thereby violating the additivity assumption that $Z = 3 + 5$. (The proportionality assumption still is satisfied since after the value of one variable is fixed, the increment in Z from the other variable is proportional to the value of that variable.) This case would arise if the two products were *complementary* in some way that *increases* profit. For example, suppose that a major advertising campaign would be required to market either new product produced by itself, but that the same single campaign can effectively promote both products if the decision is made to produce both. Because a major cost is saved for the second

TABLE 3.5 Examples of satisfying or violating additivity for the objective function

	Value of Z		
		Additivity Violated	
(x_1, x_2)	Additivity Satisfied	Case 1	Case 2
(1, 0)	3	3	3
(0, 1)	5	5	5
(1, 1)	8	9	7

product, their joint profit is somewhat more than the *sum* of their individual profits when each is produced by itself.

Case 2 in Table 3.5 also violates the additivity assumption because of the extra term in the corresponding objective function, $Z = 3x_1 + 5x_2 - x_1x_2$, so that $Z = 3 + 5 - 1 = 7$ for $(x_1, x_2) = (1, 1)$. As the reverse of the first case, Case 2 would arise if the two products were *competitive* in some way that *decreased* their joint profit. For example, suppose that both products need to use the same machinery and equipment. If either product were produced by itself, this machinery and equipment would be dedicated to this one use. However, producing both products would require switching the production processes back and forth, with substantial time and cost involved in temporarily shutting down the production of one product and setting up for the other. Because of this major extra cost, their joint profit is somewhat less than the *sum* of their individual profits when each is produced by itself.

The same kinds of interaction between activities can affect the additivity of the constraint functions. For example, consider the third functional constraint of the Wyndor Glass Co. problem: $3x_1 + 2x_2 \leq 18$. (This is the only constraint involving both products.) This constraint concerns the production capacity of Plant 3, where 18 hours of production time per week is available for the two new products, and the function on the left-hand side $(3x_1 + 2x_2)$ represents the number of hours of production time per week that would be used by these products. The *additivity satisfied* column of Table 3.6 shows this case as is, whereas the next two columns display cases where the function has an extra cross-product term that violates additivity. For all three columns, the *individual contributions* from the products toward using the capacity of Plant 3 are just as assumed previously, namely, $3x_1$ for product 1 and $2x_2$ for product 2, or $3(2) = 6$ for $x_1 = 2$ and $2(3) = 6$ for $x_2 = 3$. As was true for Table 3.5, the difference lies in the last row, which now gives the *total function value* for production time used when the two products are produced jointly.

For Case 3 (see Table 3.6), the production time used by the two products is given by the function $3x_1 + 2x_2 + 0.5x_1x_2$, so the *total function value* is $6 + 6 + 3 = 15$ when $(x_1, x_2) = (2, 3)$, which violates the additivity assumption that the value is just $6 + 6 = 12$. This case can arise in exactly the same way as described for Case 2 in Table 3.5; namely, extra time is wasted switching the production processes back and forth between the two products. The extra cross-product term $(0.5x_1x_2)$ would give the production time wasted in this way. (Note that wasting time switching between products leads to a positive cross-product term here, where the total function is measuring production time used, whereas it led to a negative cross-product term for Case 2 because the total function there measures profit.)

For Case 4 in Table 3.6, the function for production time used is $3x_1 + 2x_2 - 0.1x_1^2x_2$, so the *function value* for $(x_1, x_2) = (2, 3)$ is $6 + 6 - 1.2 = 10.8$. This case could arise in the following way. As in Case 3, suppose that the two products require the same type of machinery and equipment. But suppose now that the time required to switch from one

TABLE 3.6 Examples of satisfying or violating additivity for a functional constraint

	Amount of Resource Used		
		Additivity Violated	
(x_1, x_2)	Additivity Satisfied	Case 3	Case 4
(2, 0)	6	6	6
(0, 3)	6	6	6
(2, 3)	12	15	10.8

product to the other would be relatively small. Because each product goes through a sequence of production operations, individual production facilities normally dedicated to that product would incur occasional idle periods. During these otherwise idle periods, these facilities can be used by the other product. Consequently, the total production time used (including idle periods) when the two products are produced jointly would be less than the *sum* of the production times used by the individual products when each is produced by itself.

After analyzing the possible kinds of interaction between the two products illustrated by these four cases, the OR team concluded that none played a major role in the actual Wyndor Glass Co. problem. Therefore, the additivity assumption was adopted as a reasonable approximation.

For other problems, if additivity is not a reasonable assumption, so that some of or all the mathematical functions of the model need to be *nonlinear* (because of the cross-product terms), you definitely enter the realm of nonlinear programming (Chap. 12).

Divisibility

Our next assumption concerns the values allowed for the decision variables.

> **Divisibility assumption:** Decision variables in a linear programming model are allowed to have *any* values, including *noninteger* values, that satisfy the functional and nonnegativity constraints. Thus, these variables are *not* restricted to just integer values. Since each decision variable represents the level of some activity, it is being assumed that the activities can be run at *fractional levels.*

For the Wyndor Glass Co. problem, the decision variables represent production rates (the number of batches of a product produced per week). Since these production rates can have *any* fractional values within the feasible region, the divisibility assumption does hold.

In certain situations, the divisibility assumption does not hold because some of or all the decision variables must be restricted to *integer values.* Mathematical models with this restriction are called *integer programming* models, and they are discussed in Chap. 11.

Certainty

Our last assumption concerns the *parameters* of the model, namely, the coefficients in the objective function c_j, the coefficients in the functional constraints a_{ij}, and the right-hand sides of the functional constraints b_i.

> **Certainty assumption:** The value assigned to each parameter of a linear programming model is assumed to be a *known constant.*

In real applications, the certainty assumption is seldom satisfied precisely. Linear programming models usually are formulated to select some future course of action. Therefore, the parameter values used would be based on a prediction of future conditions, which inevitably introduces some degree of uncertainty.

For this reason it is usually important to conduct **sensitivity analysis** after a solution is found that is optimal under the assumed parameter values. As discussed in Sec. 2.3, one purpose is to identify the *sensitive* parameters (those whose value cannot be changed without changing the optimal solution), since any later change in the value of a sensitive parameter immediately signals a need to change the solution being used.

Sensitivity analysis plays an important role in the analysis of the Wyndor Glass Co. problem, as you will see in Sec. 6.7. However, it is necessary to acquire some more background before we finish that story.

Occasionally, the degree of uncertainty in the parameters is too great to be amenable to sensitivity analysis. In this case, it is necessary to treat the parameters explicitly as *random variables*. Formulations of this kind have been developed, as discussed in Secs. 23.6 and 23.7 on the book's website.

The Assumptions in Perspective

We emphasized in Sec. 2.2 that a mathematical model is intended to be only an idealized representation of the real problem. Approximations and simplifying assumptions generally are required in order for the model to be tractable. Adding too much detail and precision can make the model too unwieldy for useful analysis of the problem. All that is really needed is that there be a reasonably high correlation between the prediction of the model and what would actually happen in the real problem.

This advice certainly is applicable to linear programming. It is very common in real applications of linear programming that almost *none* of the four assumptions hold completely. Except perhaps for the *divisibility assumption,* minor disparities are to be expected. This is especially true for the *certainty assumption,* so sensitivity analysis normally is a must to compensate for the violation of this assumption.

However, it is important for the OR team to examine the four assumptions for the problem under study and to analyze just how large the disparities are. If any of the assumptions are violated in a major way, then a number of useful alternative models are available, as presented in later chapters of the book. A disadvantage of these other models is that the algorithms available for solving them are not nearly as powerful as those for linear programming, but this gap has been closing in some cases. For some applications, the powerful linear programming approach is used for the initial analysis, and then a more complicated model is used to refine this analysis.

As you work through the examples in Sec. 3.4, you will find it good practice to analyze how well each of the four assumptions of linear programming applies.

3.4 ADDITIONAL EXAMPLES

The Wyndor Glass Co. problem is a prototype example of linear programming in several respects: It involves allocating limited resources among competing activities, its model fits our standard form, and its context is the traditional one of improved business planning. However, the applicability of linear programming is much wider. In this section we begin broadening our horizons. As you study the following examples, note that it is their underlying mathematical model rather than their context that characterizes them as linear programming problems. Then give some thought to how the same mathematical model could arise in many other contexts by merely changing the names of the activities and so forth.

These examples are scaled-down versions of actual applications. Like the Wyndor problem and the demonstration example for the graphical method in OR Tutor, the first of these examples has only two decision variables and so can be solved by the graphical method. The new features are that it is a minimization problem and has a mixture of forms for the functional constraints. (This example considerably simplifies the real situation when designing radiation therapy, but the first application vignette in this section describes the exciting impact that OR actually is having in this area.) The subsequent examples have considerably more than two decision variables and so are more challenging to formulate. Although we will mention their optimal solutions that are obtained by the simplex method, the focus here is on how to formulate the linear programming model for these larger problems. Subsequent sections and the next chapter will turn to the question of the software tools and the algorithm (the simplex method) that are used to solve such problems.

If you find that you need **additional examples** of formulating small and relatively straightforward linear programming models before dealing with these more challenging formulation examples, we suggest that you go back to the demonstration example for the graphical method in OR Tutor and to the examples in the Worked Examples section for this chapter on the book's website.

Design of Radiation Therapy

MARY has just been diagnosed as having a cancer at a fairly advanced stage. Specifically, she has a large malignant tumor in the bladder area (a "whole bladder lesion").

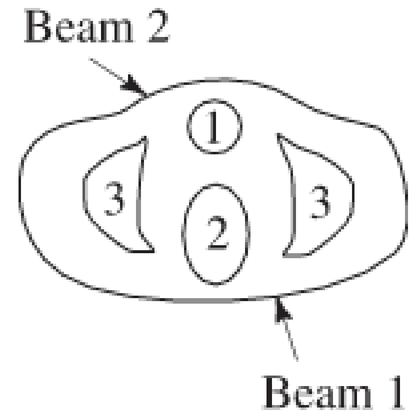
Mary is to receive the most advanced medical care available to give her every possible chance for survival. This care will include extensive *radiation therapy.*

Radiation therapy involves using an external beam treatment machine to pass ionizing radiation through the patient's body, damaging both cancerous and healthy tissues. Normally, several beams are precisely administered from different angles in a two-dimensional plane. Due to attenuation, each beam delivers more radiation to the tissue near the entry point than to the tissue near the exit point. Scatter also causes some delivery of radiation to tissue outside the direct path of the beam. Because tumor cells are typically microscopically interspersed among healthy cells, the radiation dosage throughout the tumor region must be large enough to kill the malignant cells, which are slightly more radiosensitive, yet small enough to spare the healthy cells. At the same time, the aggregate dose to critical tissues must not exceed established tolerance levels, in order to prevent complications that can be more serious than the disease itself. For the same reason, the total dose to the entire healthy anatomy must be minimized.

Because of the need to carefully balance all these factors, the design of radiation therapy is a very delicate process. The goal of the design is to select the combination of beams to be used, and the intensity of each one, to generate the best possible dose distribution. (The dose strength at any point in the body is measured in units called *kilorads.*) Once the treatment design has been developed, it is administered in many installments, spread over several weeks.

In Mary's case, the size and location of her tumor make the design of her treatment an even more delicate process than usual. Figure 3.11 shows a diagram of a cross section of the tumor viewed from above, as well as nearby critical tissues to avoid. These tissues include critical organs (e.g., the rectum) as well as bony structures (e.g., the femurs and pelvis) that will attenuate the radiation. Also shown are the entry point and direction for the only two beams that can be used with any modicum of safety in this case. (Actually, we are simplifying the example at this point, because normally dozens of possible beams must be considered.)

FIGURE 3.11
Cross section of Mary's tumor (viewed from above), nearby critical tissues, and the radiation beams being used.



1. Bladder and tumor
2. Rectum, coccyx, etc.
3. Femur, part of pelvis, etc.

For any proposed beam of given intensity, the analysis of what the resulting radiation absorption by various parts of the body would be requires a complicated process. In brief, based on careful anatomical analysis, the energy distribution within the two-dimensional cross section of the tissue can be plotted on an isodose map, where the contour lines represent the dose strength as a percentage of the dose strength at the entry point. A fine grid then is placed over the isodose map. By summing the radiation absorbed in the squares containing each type of tissue, the average dose that is absorbed by the tumor, healthy anatomy, and critical tissues can be calculated. With more than one beam (administered sequentially), the radiation absorption is additive.

After thorough analysis of this type, the medical team has carefully estimated the data needed to design Mary's treatment, as summarized in Table 3.7. The first column lists the areas of the body that must be considered, and then the next two columns give the fraction of the radiation dose at the entry point for each beam that is absorbed by the

An Application Vignette

Prostate cancer is the most common form of cancer diagnosed in men. There were an estimated 220,000 new cases in just the United States alone in 2007. Like many other forms of cancer, *radiation therapy* is a common method of treatment for prostate cancer, where the goal is to have a sufficiently high radiation dosage in the tumor region to kill the malignant cells while minimizing the radiation exposure to critical healthy structures near the tumor. This treatment can be applied through either *external beam* radiation therapy (as illustrated by the first example in this section) or *brachytherapy,* which involves placing approximately 100 radioactive "seeds" within the tumor region. The challenge is to determine the most effective three-dimensional geometric pattern for placing these seeds.

Memorial Sloan-Kettering Cancer Center (MSKCC) in New York City is the world's oldest private cancer center. An OR team from the *Center for Operations Research in Medicine and HealthCare* at Georgia Institute of Technology worked with physicians at MSKCC to develop a highly sophisticated *next-generation method* of optimizing the application of brachytherapy to prostrate cancer. The underlying model fits the structure for linear programming with one exception. In addition to having the usual continuous variables that fit linear programming, the model also has some *binary variables* (variables whose only possible values are 0 and 1). (This kind of extension of linear programming to what is called *mixed-integer programming* will be discussed in Chap. 11.) The optimization is done in a matter of minutes by an automated computerized planning system that can be operated readily by medical personnel when beginning the procedure of inserting the seeds into the patient's prostrate.

This breakthrough in optimizing the application of brachytherapy to prostrate cancer is having a profound impact on both health care costs and quality of life for treated patients because of its much greater effectiveness and the substantial reduction in side effects. When all U.S. clinics adopt this procedure, it is estimated that the annual cost savings will approximate **$500 million** due to eliminating the need for a pretreatment planning meeting and a postoperation CT scan, as well as providing a more efficient surgical procedure and reducing the need to treat subsequent side effects. It also is anticipated that this approach can be extended to other forms of brachytherapy, such as treatment of breast, cervix, esophagus, biliary tract, pancreas, head and neck, and eye.

This application of linear programming and its extensions led to the OR team winning the prestigious First Prize in the 2007 international competition for the Franz Edelman Award for Achievement in Operations Research and the Management Sciences.

Source: E. K. Lee and M. Zaider, "Operations Research Advances Cancer Therapeutics," *Interfaces,* **38**(1): 5–25, Jan.–Feb. 2008. (A link to this article is provided on our website, www.mhhe.com/hillier.)

respective areas on average. For example, if the dose level at the entry point for beam 1 is 1 kilorad, then an average of 0.4 kilorad will be absorbed by the entire healthy anatomy in the two-dimensional plane, an average of 0.3 kilorad will be absorbed by nearby critical tissues, an average of 0.5 kilorad will be absorbed by the various parts of the tumor, and 0.6 kilorad will be absorbed by the center of the tumor. The last column gives the restrictions on the total dosage from both beams that is absorbed on average by the respective areas of the body. In particular, the average dosage absorption for the

TABLE 3.7 Data for the design of Mary's radiation therapy

Area	Fraction of Entry Dose Absorbed by Area (Average): Beam 1	Beam 2	Restriction on Total Average Dosage, Kilorads
Healthy anatomy	0.4	0.5	Minimize
Critical tissues	0.3	0.1	$\leq$ 2.7
Tumor region	0.5	0.5	$=$ 6
Center of tumor	0.6	0.4	$\geq$ 6

healthy anatomy must be *as small as possible,* the critical tissues must *not exceed* 2.7 kilorads, the average over the entire tumor must *equal* 6 kilorads, and the center of the tumor must be *at least* 6 kilorads.

Formulation as a Linear Programming Problem. The decisions that need to be made are the dosages of radiation at the two entry points. Therefore, the two decision variables x_1 and x_2 represent the dose (in kilorads) at the entry point for beam 1 and beam 2, respectively. Because the total dosage reaching the healthy anatomy is to be minimized, let Z denote this quantity. The data from Table 3.7 can then be used directly to formulate the following linear programming model.[4]

$$\text{Minimize} \quad Z = 0.4x_1 + 0.5x_2,$$

subject to

$$0.3x_1 + 0.1x_2 \leq 2.7$$
$$0.5x_1 + 0.5x_2 = 6$$
$$0.6x_1 + 0.4x_2 \geq 6$$

and

$$x_1 \geq 0, \quad x_2 \geq 0.$$

Notice the differences between this model and the one in Sec. 3.1 for the Wyndor Glass Co. problem. The latter model involved *maximizing Z,* and all the functional constraints were in $\leq$ form. This new model does not fit this same standard form, but it does incorporate three other *legitimate* forms described in Sec. 3.2, namely, *minimizing Z,* functional constraints in $=$ form, and functional constraints in $\geq$ form.

However, both models have only two variables, so this new problem also can be solved by the *graphical method* illustrated in Sec. 3.1. Figure 3.12 shows the graphical solution. The *feasible region* consists of just the dark line segment between (6, 6) and (7.5, 4.5), because the points on this segment are the only ones that simultaneously satisfy all the constraints. (Note that the equality constraint limits the feasible region to the line containing this line segment, and then the other two functional constraints determine the two endpoints of the line segment.) The dashed line is the objective function line that passes through the optimal solution $(x_1, x_2) = (7.5, 4.5)$ with $Z = 5.25$. This solution is optimal rather than the point (6, 6) because *decreasing* Z (for positive values of Z) pushes the objective function line toward the origin (where $Z = 0$). And $Z = 5.25$ for (7.5, 4.5) is less than $Z = 5.4$ for (6, 6).

Thus, the optimal design is to use a total dose at the entry point of 7.5 kilorads for beam 1 and 4.5 kilorads for beam 2.

Regional Planning

The SOUTHERN CONFEDERATION OF KIBBUTZIM is a group of three kibbutzim (communal farming communities) in Israel. Overall planning for this group is done in its Coordinating Technical Office. This office currently is planning agricultural production for the coming year.

[4]This model is *much* smaller than normally would be needed for actual applications. For the best results, a realistic model might even need many tens of thousands of decision variables and constraints. For example, see H. E. Romeijn, R. K. Ahuja, J. F. Dempsey, and A. Kumar, "A New Linear Programming Approach to Radiation Therapy Treatment Planning Problems," *Operations Research,* **54**(2): 201–216, March–April 2006. For alternative approaches that combine linear programming with other OR techniques (like the first application vignette in this section), also see G. J. Lim, M. C. Ferris, S. J. Wright, D. M. Shepard, and M. A. Earl, "An Optimization Framework for Conformal Radiation Treatment Planning," *INFORMS Journal on Computing,* **19**(3): 366–380, Summer 2007.

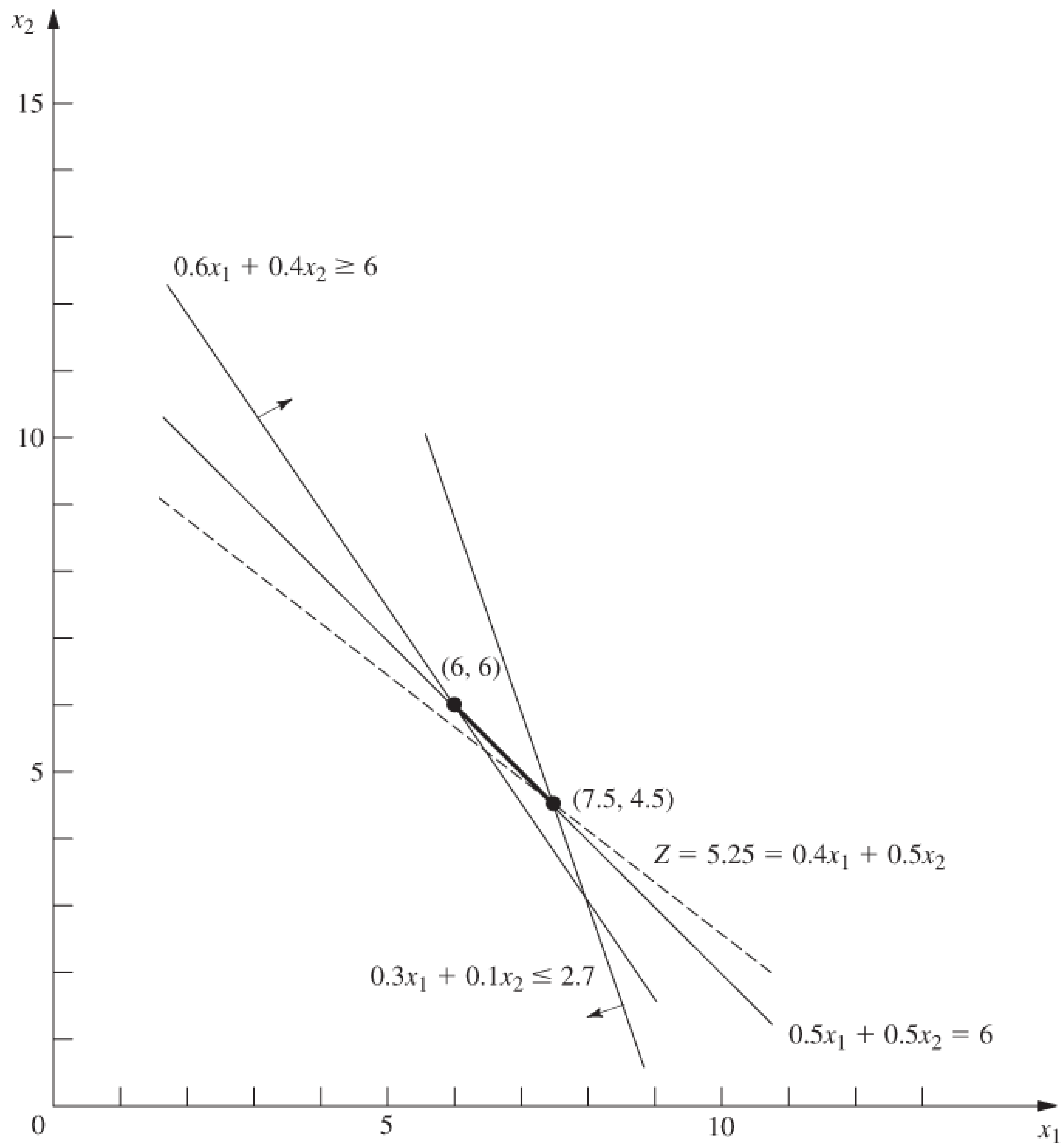


FIGURE 3.12 Graphical solution for the design of Mary's radiation therapy.

The agricultural output of each kibbutz is limited by both the amount of available irrigable land and the quantity of water allocated for irrigation by the Water Commissioner (a national government official). These data are given in Table 3.8.

The crops suited for this region include sugar beets, cotton, and sorghum, and these are the three being considered for the upcoming season. These crops differ primarily in their expected net return per acre and their consumption of water. In addition, the Ministry of Agriculture has set a maximum quota for the total acreage that can be devoted to each of these crops by the Southern Confederation of Kibbutzim, as shown in Table 3.9.

TABLE 3.8 Resource data for the Southern Confederation of Kibbutzim

Kibbutz	Usable Land (Acres)	Water Allocation (Acre Feet)
1	400	600
2	600	800
3	300	375

TABLE 3.9 Crop data for the Southern Confederation of Kibbutzim

Crop	Maximum Quota (Acres)	Water Consumption (Acre Feet/Acre)	Net Return ($/Acre)
Sugar beets	600	3	1,000
Cotton	500	2	750
Sorghum	325	1	250

Because of the limited water available for irrigation, the Southern Confederation of Kibbutzim will not be able to use all its irrigable land for planting crops in the upcoming season. To ensure equity between the three kibbutzim, it has been agreed that every kibbutz will plant the same proportion of its available irrigable land. For example, if kibbutz 1 plants 200 of its available 400 acres, then kibbutz 2 must plant 300 of its 600 acres, while kibbutz 3 plants 150 acres of its 300 acres. However, any combination of the crops may be grown at any of the kibbutzim. The job facing the Coordinating Technical Office is to plan how many acres to devote to each crop at the respective kibbutzim while satisfying the given restrictions. The objective is to maximize the total net return to the Southern Confederation of Kibbutzim as a whole.

Formulation as a Linear Programming Problem. The quantities to be decided upon are the number of acres to devote to each of the three crops at each of the three kibbutzim. The decision variables x_j ($j = 1, 2, \ldots, 9$) represent these nine quantities, as shown in Table 3.10.

Since the measure of effectiveness Z is the total net return, the resulting linear programming model for this problem is

$$\text{Maximize} \quad Z = 1{,}000(x_1 + x_2 + x_3) + 750(x_4 + x_5 + x_6) + 250(x_7 + x_8 + x_9),$$

subject to the following constraints:

1. Usable land for each kibbutz:

$$x_1 + x_4 + x_7 \leq 400$$
$$x_2 + x_5 + x_8 \leq 600$$
$$x_3 + x_6 + x_9 \leq 300$$

2. Water allocation for each kibbutz:

$$3x_1 + 2x_4 + x_7 \leq 600$$
$$3x_2 + 2x_5 + x_8 \leq 800$$
$$3x_3 + 2x_6 + x_9 \leq 375$$

TABLE 3.10 Decision variables for the Southern Confederation of Kibbutzim problem

	Allocation (Acres)		
	Kibbutz		
Crop	1	2	3
Sugar beets	x_1	x_2	x_3
Cotton	x_4	x_5	x_6
Sorghum	x_7	x_8	x_9

3. Total acreage for each crop:

$$x_1 + x_2 + x_3 \le 600$$
$$x_4 + x_5 + x_6 \le 500$$
$$x_7 + x_8 + x_9 \le 325$$

4. Equal proportion of land planted:

$$\frac{x_1 + x_4 + x_7}{400} = \frac{x_2 + x_5 + x_8}{600}$$
$$\frac{x_2 + x_5 + x_8}{600} = \frac{x_3 + x_6 + x_9}{300}$$
$$\frac{x_3 + x_6 + x_9}{300} = \frac{x_1 + x_4 + x_7}{400}$$

5. Nonnegativity:

$$x_j \ge 0, \qquad \text{for } j = 1, 2, \ldots, 9.$$

This completes the model, except that the equality constraints are not yet in an appropriate form for a linear programming model because some of the variables are on the right-hand side. Hence, their final form[5] is

$$3(x_1 + x_4 + x_7) - 2(x_2 + x_5 + x_8) = 0$$
$$(x_2 + x_5 + x_8) - 2(x_3 + x_6 + x_9) = 0$$
$$4(x_3 + x_6 + x_9) - 3(x_1 + x_4 + x_7) = 0$$

The Coordinating Technical Office formulated this model and then applied the simplex method (developed in Chap. 4) to find an optimal solution

$$(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9) = \left(133\frac{1}{3}, 100, 25, 100, 250, 150, 0, 0, 0\right),$$

as shown in Table 3.11. The resulting optimal value of the objective function is Z=633, $333\frac{1}{3}$, that is, a total net return of $633,333.33.

TABLE 3.11 Optimal solution for the Southern Confederation of Kibbutzim problem

	Best Allocation (Acres)		
	Kibbutz		
Crop	**1**	**2**	**3**
Sugar beets	$133\frac{1}{3}$	100	25
Cotton	100	250	150
Sorghum	0	0	0

[5]Actually, any one of these equations is redundant and can be deleted if desired. Also, because of these equations, any two of the usable land constraints also could be deleted because they automatically would be satisfied when both the remaining usable land constraint and these equations are satisfied. However, no harm is done (except a little more computational effort) by including unnecessary constraints, so you don't need to worry about identifying and deleting them in models you formulate.

Controlling Air Pollution

The NORI & LEETS CO., one of the major producers of steel in its part of the world, is located in the city of Steeltown and is the only large employer there. Steeltown has grown and prospered along with the company, which now employs nearly 50,000 residents. Therefore, the attitude of the townspeople always has been, What's good for Nori & Leets is good for the town. However, this attitude is now changing; uncontrolled air pollution from the company's furnaces is ruining the appearance of the city and endangering the health of its residents.

A recent stockholders' revolt resulted in the election of a new enlightened board of directors for the company. These directors are determined to follow socially responsible policies, and they have been discussing with Steeltown city officials and citizens' groups what to do about the air pollution problem. Together they have worked out stringent air quality standards for the Steeltown airshed.

The three main types of pollutants in this airshed are particulate matter, sulfur oxides, and hydrocarbons. The new standards require that the company reduce its annual emission of these pollutants by the amounts shown in Table 3.12. The board of directors has instructed management to have the engineering staff determine how to achieve these reductions in the most economical way.

The steelworks has two primary sources of pollution, namely, the blast furnaces for making pig iron and the open-hearth furnaces for changing iron into steel. In both cases the engineers have decided that the most effective types of abatement methods are (1) increasing the height of the smokestacks,[6] (2) using filter devices (including gas traps) in the smokestacks, and (3) including cleaner, high-grade materials among the fuels for the furnaces. Each of these methods has a technological limit on how heavily it can be used (e.g., a maximum feasible increase in the height of the smokestacks), but there also is considerable flexibility for using the method at a fraction of its technological limit.

Table 3.13 shows how much emission (in millions of pounds per year) can be eliminated from each type of furnace by fully using any abatement method to its technological limit. For purposes of analysis, it is assumed that each method also can be used less fully to achieve any fraction of the emission-rate reductions shown in this table. Furthermore, the fractions can be different for blast furnaces and for open-hearth furnaces. For either type of furnace, the emission reduction achieved by each method is not substantially affected by whether the other methods also are used.

After these data were developed, it became clear that no single method by itself could achieve all the required reductions. On the other hand, combining all three methods at full capacity on both types of furnaces (which would be prohibitively expensive if the company's

TABLE 3.12 Clean air standards for the Nori & Leets Co.

Pollutant	Required Reduction in Annual Emission Rate (Million Pounds)
Particulates	60
Sulfur oxides	150
Hydrocarbons	125

[6]Subsequent to this study, this particular abatement method has become a controversial one. Because its effect is to reduce ground-level pollution by spreading emissions over a greater distance, environmental groups contend that this creates more acid rain by keeping sulfur oxides in the air longer. Consequently, the U.S. Environmental Protection Agency adopted new rules in 1985 to remove incentives for using tall smokestacks.

TABLE 3.13 Reduction in emission rate (in millions of pounds per year) from the maximum feasible use of an abatement method for Nori & Leets Co.

Pollutant	Taller Smokestacks		Filters		Better Fuels	
	Blast Furnaces	Open-Hearth Furnaces	Blast Furnaces	Open-Hearth Furnaces	Blast Furnaces	Open-Hearth Furnaces
Particulates	12	9	25	20	17	13
Sulfur oxides	35	42	18	31	56	49
Hydrocarbons	37	53	28	24	29	20

products are to remain competitively priced) is much more than adequate. Therefore, the engineers concluded that they would have to use some combination of the methods, perhaps with fractional capacities, based upon the relative costs. Furthermore, because of the differences between the blast and the open-hearth furnaces, the two types probably should not use the same combination.

An analysis was conducted to estimate the total annual cost that would be incurred by each abatement method. A method's annual cost includes increased operating and maintenance expenses as well as reduced revenue due to any loss in the efficiency of the production process caused by using the method. The other major cost is the *start-up cost* (the initial capital outlay) required to install the method. To make this one-time cost commensurable with the ongoing annual costs, the time value of money was used to calculate the annual expenditure (over the expected life of the method) that would be equivalent in value to this start-up cost.

This analysis led to the total annual cost estimates (in millions of dollars) given in Table 3.14 for using the methods at their full abatement capacities. It also was determined that the cost of a method being used at a lower level is roughly proportional to the fraction of the abatement capacity given in Table 3.13 that is achieved. Thus, for any given fraction achieved, the total annual cost would be roughly that fraction of the corresponding quantity in Table 3.14.

The stage now was set to develop the general framework of the company's plan for pollution abatement. This plan specifies which types of abatement methods will be used and at what fractions of their abatement capacities for (1) the blast furnaces and (2) the open-hearth furnaces. Because of the combinatorial nature of the problem of finding a plan that satisfies the requirements with the smallest possible cost, an OR team was formed to solve the problem. The team adopted a linear programming approach, formulating the model summarized next.

Formulation as a Linear Programming Problem. This problem has six decision variables x_j, $j = 1, 2, \ldots, 6$, each representing the use of one of the three abatement methods for one of the two types of furnaces, expressed as a *fraction of the abatement capacity* (so x_j cannot exceed 1). The ordering of these variables is shown in Table 3.15. Because the

TABLE 3.14 Total annual cost from the maximum feasible use of an abatement method for Nori & Leets Co. ($ millions)

Abatement Method	Blast Furnaces	Open-Hearth Furnaces
Taller smokestacks	8	10
Filters	7	6
Better fuels	11	9

TABLE 3.15 Decision variables (fraction of the maximum feasible use of an abatement method) for Nori & Leets Co.

Abatement Method	Blast Furnaces	Open-Hearth Furnaces
Taller smokestacks	x_1	x_2
Filters	x_3	x_4
Better fuels	x_5	x_6

objective is to minimize total cost while satisfying the emission reduction requirements, the data in Tables 3.12, 3.13, and 3.14 yield the following model:

$$\text{Minimize} \quad Z = 8x_1 + 10x_2 + 7x_3 + 6x_4 + 11x_5 + 9x_6,$$

subject to the following constraints:

1. Emission reduction:

$$\begin{aligned} 12x_1 + 9x_2 + 25x_3 + 20x_4 + 17x_5 + 13x_6 &\geq 60 \\ 35x_1 + 42x_2 + 18x_3 + 31x_4 + 56x_5 + 49x_6 &\geq 150 \\ 37x_1 + 53x_2 + 28x_3 + 24x_4 + 29x_5 + 20x_6 &\geq 125 \end{aligned}$$

2. Technological limit:

$$x_j \leq 1, \quad \text{for } j = 1, 2, \ldots, 6$$

3. Nonnegativity:

$$x_j \geq 0, \quad \text{for } j = 1, 2, \ldots, 6.$$

The OR team used this model[7] to find a minimum-cost plan

$$(x_1, x_2, x_3, x_4, x_5, x_6) = (1, 0.623, 0.343, 1, 0.048, 1),$$

with $Z = 32.16$ (total annual cost of \$32.16 million). Sensitivity analysis then was conducted to explore the effect of making possible adjustments in the air standards given in Table 3.12, as well as to check on the effect of any inaccuracies in the cost data given in Table 3.14. (This story is continued in Case 6.1 at the end of Chap. 6.) Next came detailed planning and managerial review. Soon after, this program for controlling air pollution was fully implemented by the company, and the citizens of Steeltown breathed deep (cleaner) sighs of relief.

Reclaiming Solid Wastes

The SAVE-IT COMPANY operates a reclamation center that collects four types of solid waste materials and treats them so that they can be amalgamated into a salable product. (Treating and amalgamating are separate processes.) Three different grades of this product can be made (see the first column of Table 3.16), depending upon the mix of the materials used. Although there is some flexibility in the mix for each grade, quality standards may specify the minimum or maximum amount allowed for the proportion of a material in the product grade. (This proportion is the weight of the material expressed as a percentage of the total weight for the product grade.) For each of the two higher grades, a fixed percentage

[7]An equivalent formulation can express each decision variable in natural units for its abatement method; for example, x_1 and x_2 could represent the number of *feet* that the heights of the smokestacks are increased.

TABLE 3.16 Product data for Save-It Co.

Grade	Specification	Amalgamation Cost per Pound ($)	Selling Price per Pound ($)
A	Material 1: Not more than 30% of total Material 2: Not less than 40% of total Material 3: Not more than 50% of total Material 4: Exactly 20% of total	3.00	8.50
B	Material 1: Not more than 50% of total Material 2: Not less than 10% of total Material 4: Exactly 10% of total	2.50	7.00
C	Material 1: Not more than 70% of total	2.00	5.50

is specified for one of the materials. These specifications are given in Table 3.16 along with the cost of amalgamation and the selling price for each grade.

The reclamation center collects its solid waste materials from regular sources and so is normally able to maintain a steady rate for treating them. Table 3.17 gives the quantities available for collection and treatment each week, as well as the cost of treatment, for each type of material.

The Save-It Co. is solely owned by Green Earth, an organization devoted to dealing with environmental issues, so Save-It's profits are used to help support Green Earth's activities. Green Earth has raised contributions and grants, amounting to $30,000 per week, to be used exclusively to cover the entire treatment cost for the solid waste materials. The board of directors of Green Earth has instructed the management of Save-It to divide this money among the materials in such a way that *at least half* of the amount available of each material is actually collected and treated. These additional restrictions are listed in Table 3.17.

Within the restrictions specified in Tables 3.16 and 3.17, management wants to determine the *amount* of each product grade to produce *and* the exact *mix* of materials to be used for each grade. The objective is to maximize the net weekly profit (total sales income *minus* total amalgamation cost), exclusive of the fixed treatment cost of $30,000 per week that is being covered by gifts and grants.

Formulation as a Linear Programming Problem. Before attempting to construct a linear programming model, we must give careful consideration to the proper definition of the decision variables. Although this definition is often obvious, it sometimes becomes the crux of the entire formulation. After clearly identifying what information is really desired and the most convenient form for conveying this information by means of decision variables, we can develop the objective function and the constraints on the values of these decision variables.

TABLE 3.17 Solid waste materials data for the Save-It Co.

Material	Pounds per Week Available	Treatment Cost per Pound ($)	Additional Restrictions
1	3,000	3.00	1. For each material, at least half of the pounds per week available should be collected and treated.
2	2,000	6.00	
3	4,000	4.00	
4	1,000	5.00	2. $30,000 per week should be used to treat these materials.

In this particular problem, the decisions to be made are well defined, but the appropriate means of conveying this information may require some thought. (Try it and see if you first obtain the following *inappropriate* choice of decision variables.)

Because one set of decisions is the *amount* of each product grade to produce, it would seem natural to define one set of decision variables accordingly. Proceeding tentatively along this line, we define

$$y_i = \text{number of pounds of product grade } i \text{ produced per week} \qquad (i = A, B, C).$$

The other set of decisions is the *mix* of materials for each product grade. This mix is identified by the proportion of each material in the product grade, which would suggest defining the other set of decision variables as

$$z_{ij} = \text{proportion of material } j \text{ in product grade } i \qquad (i = A, B, C; j = 1, 2, 3, 4).$$

However, Table 3.17 gives both the treatment cost and the availability of the materials by *quantity* (pounds) rather than *proportion,* so it is this *quantity* information that needs to be recorded in some of the constraints. For material j ($j = 1, 2, 3, 4$),

$$\text{Number of pounds of material } j \text{ used per week} = z_{Aj}y_A + z_{Bj}y_B + z_{Cj}y_C.$$

For example, since Table 3.17 indicates that 3,000 pounds of material 1 is available per week, one constraint in the model would be

$$z_{A1}y_A + z_{B1}y_B + z_{C1}y_C \leq 3{,}000.$$

Unfortunately, this is not a legitimate linear programming constraint. The expression on the left-hand side is *not* a linear function because it involves products of variables. Therefore, a linear programming model cannot be constructed with these decision variables.

Fortunately, there is another way of defining the decision variables that will fit the linear programming format. (Do you see how to do it?) It is accomplished by merely replacing each *product* of the old decision variables by a single variable! In other words, define

$$\begin{aligned} x_{ij} &= z_{ij}y_i \qquad (\text{for } i = A, B, C; j = 1, 2, 3, 4) \\ &= \text{number of pounds of material } j \text{ allocated to product grade } i \text{ per week,} \end{aligned}$$

and then we let the x_{ij} be the decision variables. Combining the x_{ij} in different ways yields the following quantities needed in the model (for i = A, B, C; j = 1, 2, 3, 4).

$$\begin{aligned} x_{i1} + x_{i2} + x_{i3} + x_{i4} &= \text{number of pounds of product grade } i \text{ produced per week.} \\ x_{Aj} + x_{Bj} + x_{Cj} &= \text{number of pounds of material } j \text{ used per week.} \\ \frac{x_{ij}}{x_{i1} + x_{i2} + x_{i3} + x_{i4}} &= \text{proportion of material } j \text{ in product grade } i. \end{aligned}$$

The fact that this last expression is a *nonlinear* function does not cause a complication. For example, consider the first specification for product grade A in Table 3.16 (the proportion of material 1 should not exceed 30 percent). This restriction gives the nonlinear constraint

$$\frac{x_{A1}}{x_{A1} + x_{A2} + x_{A3} + x_{A4}} \leq 0.3.$$

However, multiplying through both sides of this inequality by the denominator yields an *equivalent* constraint

$$x_{A1} \leq 0.3(x_{A1} + x_{A2} + x_{A3} + x_{A4}),$$

so

$$0.7x_{A1} - 0.3x_{A2} - 0.3x_{A3} - 0.3x_{A4} \leq 0,$$

which is a legitimate linear programming constraint.

With this adjustment, the three quantities given above lead directly to all the functional constraints of the model. The objective function is based on management's objective of maximizing net weekly profit (total sales income *minus* total amalgamation cost) from the three product grades. Thus, for each product grade, the profit per pound is obtained by subtracting the amalgamation cost given in the third column of Table 3.16 from the selling price in the fourth column. These *differences* provide the coefficients for the objective function.

Therefore, the complete linear programming model is

$$\text{Maximize} \quad Z = 5.5(x_{A1} + x_{A2} + x_{A3} + x_{A4}) + 4.5(x_{B1} + x_{B2} + x_{B3} + x_{B4}) + 3.5(x_{C1} + x_{C2} + x_{C3} + x_{C4}),$$

subject to the following constraints:

1. Mixture specifications (second column of Table 3.16):

$$\begin{aligned}
x_{A1} &\leq 0.3(x_{A1} + x_{A2} + x_{A3} + x_{A4}) && \text{(grade } A\text{, material 1)}\\
x_{A2} &\geq 0.4(x_{A1} + x_{A2} + x_{A3} + x_{A4}) && \text{(grade } A\text{, material 2)}\\
x_{A3} &\leq 0.5(x_{A1} + x_{A2} + x_{A3} + x_{A4}) && \text{(grade } A\text{, material 3)}\\
x_{A4} &= 0.2(x_{A1} + x_{A2} + x_{A3} + x_{A4}) && \text{(grade } A\text{, material 4)}\\
x_{B1} &\leq 0.5(x_{B1} + x_{B2} + x_{B3} + x_{B4}) && \text{(grade } B\text{, material 1)}\\
x_{B2} &\geq 0.1(x_{B1} + x_{B2} + x_{B3} + x_{B4}) && \text{(grade } B\text{, material 2)}\\
x_{B4} &= 0.1(x_{B1} + x_{B2} + x_{B3} + x_{B4}) && \text{(grade } B\text{, material 4)}\\
x_{C1} &\leq 0.7(x_{C1} + x_{C2} + x_{C3} + x_{C4}) && \text{(grade } C\text{, material 1).}
\end{aligned}$$

2. Availability of materials (second column of Table 3.17):

$$\begin{aligned}
x_{A1} + x_{B1} + x_{C1} &\leq 3{,}000 && \text{(material 1)}\\
x_{A2} + x_{B2} + x_{C2} &\leq 2{,}000 && \text{(material 2)}\\
x_{A3} + x_{B3} + x_{C3} &\leq 4{,}000 && \text{(material 3)}\\
x_{A4} + x_{B4} + x_{C4} &\leq 1{,}000 && \text{(material 4).}
\end{aligned}$$

3. Restrictions on amounts treated (right side of Table 3.17):

$$\begin{aligned}
x_{A1} + x_{B1} + x_{C1} &\geq 1{,}500 && \text{(material 1)}\\
x_{A2} + x_{B2} + x_{C2} &\geq 1{,}000 && \text{(material 2)}\\
x_{A3} + x_{B3} + x_{C3} &\geq 2{,}000 && \text{(material 3)}\\
x_{A4} + x_{B4} + x_{C4} &\geq 500 && \text{(material 4).}
\end{aligned}$$

4. Restriction on treatment cost (right side of Table 3.17):

$$3(x_{A1} + x_{B1} + x_{C1}) + 6(x_{A2} + x_{B2} + x_{C2}) + 4(x_{A3} + x_{B3} + x_{C3}) + 5(x_{A4} + x_{B4} + x_{C4}) = 30{,}000.$$

5. Nonnegativity constraints:

$$x_{A1} \geq 0, \quad x_{A2} \geq 0, \quad \ldots, \quad x_{C4} \geq 0.$$

TABLE 3.18 Optimal solution for the Save-It Co. problem

	Pounds Used per Week				
	Material				
Grade	1	2	3	4	Number of Pounds Produced per Week
A	412.3 (19.2%)	859.6 (40%)	447.4 (20.8%)	429.8 (20%)	2149
B	2587.7 (50%)	517.5 (10%)	1552.6 (30%)	517.5 (10%)	5175
C	0	0	0	0	0
Total	3000	1377	2000	947	

This formulation completes the model, except that the constraints for the mixture specifications need to be rewritten in the proper form for a linear programming model by bringing all variables to the left-hand side and combining terms, as follows:

Mixture specifications:

$$
\begin{aligned}
0.7x_{A1} - 0.3x_{A2} - 0.3x_{A3} - 0.3x_{A4} &\leq 0 \quad &&(\text{grade } A, \text{ material } 1)\\
-0.4x_{A1} + 0.6x_{A2} - 0.4x_{A3} - 0.4x_{A4} &\geq 0 \quad &&(\text{grade } A, \text{ material } 2)\\
-0.5x_{A1} - 0.5x_{A2} + 0.5x_{A3} - 0.5x_{A4} &\leq 0 \quad &&(\text{grade } A, \text{ material } 3)\\
-0.2x_{A1} - 0.2x_{A2} - 0.2x_{A3} + 0.8x_{A4} &= 0 \quad &&(\text{grade } A, \text{ material } 4)\\
0.5x_{B1} - 0.5x_{B2} - 0.5x_{B3} - 0.5x_{B4} &\leq 0 \quad &&(\text{grade } B, \text{ material } 1)\\
-0.1x_{B1} + 0.9x_{B2} - 0.1x_{B3} - 0.1x_{B4} &\geq 0 \quad &&(\text{grade } B, \text{ material } 2)\\
-0.1x_{B1} - 0.1x_{B2} - 0.1x_{B3} + 0.9x_{B4} &= 0 \quad &&(\text{grade } B, \text{ material } 4)\\
0.3x_{C1} - 0.7x_{C2} - 0.7x_{C3} - 0.7x_{C4} &\leq 0 \quad &&(\text{grade } C, \text{ material } 1).
\end{aligned}
$$

An optimal solution for this model is shown in Table 3.18, and then these x_{ij} values are used to calculate the other quantities of interest given in the table. The resulting optimal value of the objective function is $Z = 35{,}109.65$ (a total weekly profit of \$35,109.65).

The Save-It Co. problem is an example of a **blending problem.** The objective for a blending problem is to find the best blend of ingredients into final products to meet certain specifications. Some of the earliest applications of linear programming were for *gasoline blending,* where petroleum ingredients were blended to obtain various grades of gasoline. Other blending problems involve such final products as steel, fertilizer, and animal feed.

Personnel Scheduling

UNION AIRWAYS is adding more flights to and from its hub airport, and so it needs to hire additional customer service agents. However, it is not clear just how many more should be hired. Management recognizes the need for cost control while also consistently providing a satisfactory level of service to customers. Therefore, an OR team is studying how to schedule the agents to provide satisfactory service with the smallest personnel cost.

Based on the new schedule of flights, an analysis has been made of the *minimum* number of customer service agents that need to be on duty at different times of the day to provide a satisfactory level of service. The rightmost column of Table 3.19 shows the number of agents needed for the time periods given in the first column. The other entries in this table reflect one of the provisions in the company's current contract with the union that

An Application Vignette

Cost control is essential for survival in the airline industry. Therefore, upper management of **United Airlines** initiated an operations research study to improve the utilization of personnel at the airline's reservations offices and airports by matching work schedules to customer needs more closely. The number of employees needed at each location to provide the required level of service varies greatly during the 24-hour day and might fluctuate considerably from one half-hour to the next.

Trying to design the work schedules for all the employees at a given location to meet these service requirements most efficiently is a nightmare of combinatorial considerations. Once an employee arrives, he or she will be there continuously for the entire shift (2 to 10 hours, depending on the employee), *except* for either a meal break or short rest breaks every two hours. Given the *minimum* number of employees needed on duty for *each* half-hour interval over a 24-hour day (this minimum changes from day to day over a seven-day week), *how many* employees of *each shift length* should begin work at *what start time* over *each* 24-hour day of a seven-day week? Fortunately, linear programming thrives on such combinatorial nightmares. The linear programming model for some of the locations scheduled involves over 20,000 decisions!

This application of linear programming was credited with *saving United Airlines more than* **$6 million** *annually* in just direct salary and benefit costs. Other benefits included improved customer service and reduced workloads for support staff.

Source: T. J. Holloran and J. E. Bryne, "United Airlines Station Manpower Planning System," *Interfaces,* **16**(1): 39–50, Jan.–Feb. 1986. (A link to this article is provided on our website, www.mhhe.com/hillier.)

represents the customer service agents. The provision is that each agent work an 8-hour shift 5 days per week, and the authorized shifts are

Shift 1: 6:00 A.M. to 2:00 P.M.
Shift 2: 8:00 A.M. to 4:00 P.M.
Shift 3: Noon to 8:00 P.M.
Shift 4: 4:00 P.M. to midnight
Shift 5: 10:00 P.M. to 6:00 A.M.

Checkmarks in the main body of Table 3.19 show the hours covered by the respective shifts. Because some shifts are less desirable than others, the wages specified in the contract differ by shift. For each shift, the daily compensation (including benefits) for each agent is shown in the bottom row. The problem is to determine how many agents should be

TABLE 3.19 Data for the Union Airways personnel scheduling problem

	Time Periods Covered					
	Shift					
Time Period	1	2	3	4	5	Minimum Number of Agents Needed
6:00 A.M. to 8:00 A.M.	✔					48
8:00 A.M. to 10:00 A.M.	✔	✔				79
10:00 A.M. to noon	✔	✔				65
Noon to 2:00 P.M.	✔	✔	✔			87
2:00 P.M. to 4:00 P.M.		✔	✔			64
4:00 P.M. to 6:00 P.M.			✔	✔		73
6:00 P.M. to 8:00 P.M.			✔	✔		82
8:00 P.M. to 10:00 P.M.				✔		43
10:00 P.M. to midnight				✔	✔	52
Midnight to 6:00 A.M.					✔	15
Daily cost per agent	$170	$160	$175	$180	$195	

assigned to the respective shifts each day to minimize the *total* personnel cost for agents, based on this bottom row, while meeting (or surpassing) the service requirements given in the rightmost column.

Formulation as a Linear Programming Problem. Linear programming problems always involve finding the best *mix of activity levels.* The key to formulating this particular problem is to recognize the nature of the activities.

Activities correspond to shifts, where the *level* of each activity is the number of agents assigned to that shift. Thus, this problem involves finding the *best mix of shift sizes.* Since the decision variables always are the levels of the activities, the five decision variables here are

$$x_j = \text{number of agents assigned to shift } j, \quad \text{for } j = 1, 2, 3, 4, 5.$$

The main restrictions on the values of these decision variables are that the number of agents working during each time period must satisfy the minimum requirement given in the rightmost column of Table 3.19. For example, for 2:00 P.M. to 4:00 P.M., the total number of agents assigned to the shifts that cover this time period (shifts 2 and 3) must be at least 64, so

$$x_2 + x_3 \geq 64$$

is the functional constraint for this time period.

Because the objective is to minimize the total cost of the agents assigned to the five shifts, the coefficients in the objective function are given by the last row of Table 3.19.

Therefore, the complete linear programming model is

$$\text{Minimize} \quad Z = 170x_1 + 160x_2 + 175x_3 + 180x_4 + 195x_5,$$

subject to

$$\begin{array}{lr}
x_1 \geq 48 & \text{(6–8 A.M.)} \\
x_1 + x_2 \geq 79 & \text{(8–10 A.M.)} \\
x_1 + x_2 \geq 65 & \text{(10 A.M. to noon)} \\
x_1 + x_2 + x_3 \geq 87 & \text{(Noon–2 P.M.)} \\
x_2 + x_3 \geq 64 & \text{(2–4 P.M.)} \\
x_3 + x_4 \geq 73 & \text{(4–6 P.M.)} \\
x_3 + x_4 \geq 82 & \text{(6–8 P.M.)} \\
x_4 \geq 43 & \text{(8–10 P.M.)} \\
x_4 + x_5 \geq 52 & \text{(10 P.M.–midnight)} \\
x_5 \geq 15 & \text{(Midnight–6 A.M.)}
\end{array}$$

and

$$x_j \geq 0, \quad \text{for } j = 1, 2, 3, 4, 5.$$

With a keen eye, you might have noticed that the third constraint, $x_1 + x_2 \geq 65$, actually is not necessary because the second constraint, $x_1 + x_2 \geq 79$, ensures that $x_1 + x_2$ will be larger than 65. Thus, $x_1 + x_2 \geq 65$ is a *redundant* constraint that can be deleted. Similarly, the sixth constraint, $x_3 + x_4 \geq 73$, also is a *redundant* constraint because the seventh constraint is $x_3 + x_4 \geq 82$. (In fact, three of the nonnegativity constraints—$x_1 \geq 0$, $x_4 \geq 0$, $x_5 \geq 0$—also are redundant constraints because of the first, eighth, and tenth functional constraints: $x_1 \geq 48$, $x_4 \geq 43$, and $x_5 \geq 15$. However, no computational advantage is gained by deleting these three nonnegativity constraints.)

The optimal solution for this model is $(x_1, x_2, x_3, x_4, x_5) = (48, 31, 39, 43, 15)$. This yields $Z = 30{,}610$, that is, a total daily personnel cost of \$30,610.

This problem is an example where the divisibility assumption of linear programming actually is not satisfied. The number of agents assigned to each shift needs to be an integer. Strictly speaking, the model should have an additional constraint for each decision variable specifying that the variable must have an integer value. Adding these constraints would convert the linear programming model to an integer programming model (the topic of Chap. 11).

Without these constraints, the optimal solution given above turned out to have integer values anyway, so no harm was done by not including the constraints. (The form of the functional constraints made this outcome a likely one.) If some of the variables had turned out to be noninteger, the easiest approach would have been to *round up* to integer values. (Rounding up is feasible for this example because all the functional constraints are in $\geq$ form with nonnegative coefficients.) Rounding up does not ensure obtaining an optimal solution for the integer programming model, but the error introduced by rounding up such large numbers would be negligible for most practical situations. Alternatively, integer programming techniques described in Chap. 11 could be used to solve exactly for an optimal solution with integer values.

The second application vignette in this section describes how United Airlines used linear programming to develop a personnel scheduling system on a vastly larger scale than this example.

Distributing Goods through a Distribution Network

The Problem. The DISTRIBUTION UNLIMITED CO. will be producing the same new product at two different factories, and then the product must be shipped to two warehouses, where either factory can supply either warehouse. The distribution network available for shipping this product is shown in Fig. 3.13, where F1 and F2 are the two factories, W1 and W2 are the two warehouses, and DC is a distribution center. The amounts to be shipped from F1 and F2 are shown to their left, and the amounts to be received at W1 and W2 are shown to their right. Each arrow represents a feasible shipping lane. Thus, F1 can ship directly to W1 and has three possible routes (F1 $\rightarrow$ DC $\rightarrow$ W2, F1 $\rightarrow$ F2 $\rightarrow$ DC $\rightarrow$ W2, and F1 $\rightarrow$ W1 $\rightarrow$ W2) for shipping to W2. Factory F2 has just one route to W2 (F2 $\rightarrow$ DC $\rightarrow$ W2) and one to W1 (F2 $\rightarrow$ DC $\rightarrow$ W2 $\rightarrow$ W1). The cost per unit shipped through each shipping lane is shown next to the arrow. Also shown next to F1 $\rightarrow$ F2 and DC $\rightarrow$ W2 are the maximum amounts that can be shipped through these lanes. The other lanes have sufficient shipping capacity to handle everything these factories can send.

The decision to be made concerns how much to ship through each shipping lane. The objective is to minimize the total shipping cost.

Formulation as a Linear Programming Problem. With seven shipping lanes, we need seven decision variables ($x_{\text{F1-F2}}$, $x_{\text{F1-DC}}$, $x_{\text{F1-W1}}$, $x_{\text{F2-DC}}$, $x_{\text{DC-W2}}$, $x_{\text{W1-W2}}$, $x_{\text{W2-W1}}$) to represent the amounts shipped through the respective lanes.

There are several restrictions on the values of these variables. In addition to the usual nonnegativity constraints, there are two *upper-bound constraints,* $x_{\text{F1-F2}} \leq 10$ and $x_{\text{DC-W2}} \leq 80$, imposed by the limited shipping capacities for the two lanes, F1 $\rightarrow$ F2 and DC $\rightarrow$ W2. All the other restrictions arise from five *net flow constraints,* one for each of the five locations. These constraints have the following form.

Net flow constraint for each location:

Amount shipped out $-$ amount shipped in $=$ required amount.

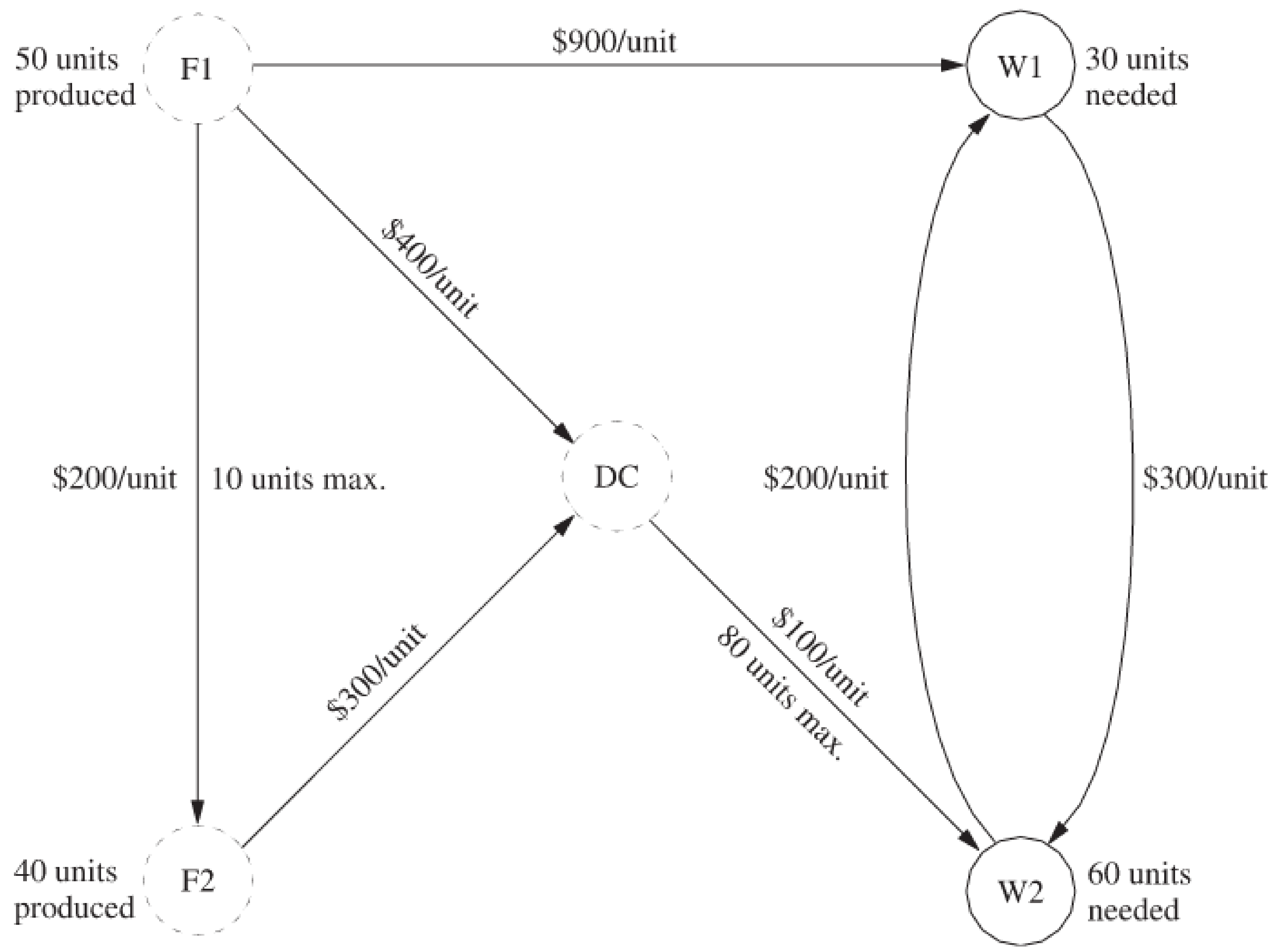


FIGURE 3.13
The distribution network for Distribution Unlimited Co.

As indicated in Fig. 3.13, these required amounts are 50 for F1, 40 for F2, -30 for W1, and -60 for W2.

What is the required amount for DC? All the units produced at the factories are ultimately needed at the warehouses, so any units shipped from the factories to the distribution center should be forwarded to the warehouses. Therefore, the total amount shipped from the distribution center to the warehouses should *equal* the total amount shipped from the factories to the distribution center. In other words, the *difference* of these two shipping amounts (the required amount for the net flow constraint) should be *zero.*

Since the objective is to minimize the total shipping cost, the coefficients for the objective function come directly from the unit shipping costs given in Fig. 3.13. Therefore, by using money units of hundreds of dollars in this objective function, the complete linear programming model is

$$\begin{aligned}\text{Minimize} \quad Z = {} & 2x_{\text{F1-F2}} + 4x_{\text{F1-DC}} + 9x_{\text{F1-W1}} + 3x_{\text{F2-DC}} + x_{\text{DC-W2}} \\ & + 3x_{\text{W1-W2}} + 2x_{\text{W2-W1}},\end{aligned}$$

subject to the following constraints:

1. Net flow constraints:

$$\begin{array}{lllllllcl}
x_{\text{F1-F2}} & + x_{\text{F1-DC}} & + x_{\text{F1-W1}} & & & & & = & 50 \text{ (factory 1)} \\
-x_{\text{F1-F2}} & & & + x_{\text{F2-DC}} & & & & = & 40 \text{ (factory 2)} \\
 & - x_{\text{F1-DC}} & & - x_{\text{F2-DC}} & + x_{\text{DC-W2}} & & & = & 0 \text{ (distribution center)} \\
 & & - x_{\text{F1-W1}} & & & + x_{\text{W1-W2}} & - x_{\text{W2-W1}} & = & -30 \text{ (warehouse 1)} \\
 & & & & - x_{\text{DC-W2}} & - x_{\text{W1-W2}} & + x_{\text{W2-W1}} & = & -60 \text{ (warehouse 2)}
\end{array}$$

2. Upper-bound constraints:

$$x_{\text{F1-F2}} \le 10, \qquad x_{\text{DC-W2}} \le 80$$

3. Nonnegativity constraints:

$$x_{\text{F1-F2}} \ge 0, \quad x_{\text{F1-DC}} \ge 0, \quad x_{\text{F1-W1}} \ge 0, \quad x_{\text{F2-DC}} \ge 0, \quad x_{\text{DC-W2}} \ge 0,$$
$$x_{\text{W1-W2}} \ge 0, \quad x_{\text{W2-W1}} \ge 0.$$

You will see this problem again in Sec. 9.6, where we focus on linear programming problems of this type (called the *minimum cost flow problem*). In Sec. 9.7, we will solve for its optimal solution:

$$x_{\text{F1-F2}} = 0, \quad x_{\text{F1-DC}} = 40, \quad x_{\text{F1-W1}} = 10, \quad x_{\text{F2-DC}} = 40, \quad x_{\text{DC-W2}} = 80,$$
$$x_{\text{W1-W2}} = 0, \quad x_{\text{W2-W1}} = 20.$$

The resulting total shipping cost is \$49,000.

3.5 FORMULATING AND SOLVING LINEAR PROGRAMMING MODELS ON A SPREADSHEET

Spreadsheet software, such as Excel, is a popular tool for analyzing and solving small linear programming problems. The main features of a linear programming model, including all its parameters, can be easily entered onto a spreadsheet. However, spreadsheet software can do much more than just display data. If we include some additional information, the spreadsheet can be used to quickly analyze potential solutions. For example, a potential solution can be checked to see if it is feasible and what Z value (profit or cost) it achieves. Much of the power of the spreadsheet lies in its ability to immediately reveal the results of any changes made in the solution.

In addition, the Excel Solver can quickly apply the simplex method to find an optimal solution for the model. We will describe how this is done in the latter part of this section.

To illustrate this process of formulating and solving linear programming models on a spreadsheet, we now return to the Wyndor example introduced in Sec. 3.1.

Formulating the Model on a Spreadsheet

Figure 3.14 displays the Wyndor problem by transferring the data from Table 3.1 onto a spreadsheet. (Columns E and F are being reserved for later entries described below.) We will refer to the cells showing the data as **data cells.** These cells are lightly shaded to distinguish them from other cells in the spreadsheet.[8]

You will see later that the spreadsheet is made easier to interpret by using range names. **A range name** is a descriptive name given to a block of cells that immediately identifies what is there. Thus, the data cells in the Wyndor problem are given the range names UnitProfit (C4:D4), HoursUsedPerBatchProduced (C7:D9), and HoursAvailable (G7:G9). Note that no spaces are allowed in a range name so each new word begins with a capital letter. Although optional, the range of cells being given each range name can be specified in parentheses following the name. (For example, the range C7:D9 is Excel shorthand for the *range from* C7 *to* D9; that is, the entire block of cells in column C or D and in row 7, 8, or 9.) To enter a range name, first select the range of cells, then choose

[8]Borders and cell shading can be added either by using the borders button and the fill color button on the formatting toolbar or by choosing Cells from the Format menu and then selecting the Borders tab and/or the Patterns tab.

An Application Vignette

Welch's, Inc., is the world's largest processor of Concord and Niagara grapes, with annual sales surpassing $550 million per year. Such products as Welch's grape jelly and Welch's grape juice have been enjoyed by generations of American consumers.

Every September, growers begin delivering grapes to processing plants that then press the raw grapes into juice. Time must pass before the grape juice is ready for conversion into finished jams, jellies, juices, and concentrates.

Deciding how to use the grape crop is a complex task given changing demand and uncertain crop quality and quantity. Typical decisions include what recipes to use for major product groups, the transfer of grape juice between plants, and the mode of transportation for these transfers.

Because Welch's lacked a formal system for optimizing raw material movement and the recipes used for production, an OR team developed a preliminary linear programming model. This was a large model with 8,000 decision variables that focused on the component level of detail. Small-scale testing proved that the model worked.

To make the model more useful, the team then revised it by aggregating demand by product group rather than by component. This reduced its size to 324 decision variables and 361 functional constraints. *The model then was incorporated into a spreadsheet.*

The company has run the continually updated version of this *spreadsheet model* each month since 1994 to provide senior management with information on the optimal logistics plan generated by the Solver. *The savings* from using and optimizing this model *were approximately* **$150,000** *in the first year alone.* A major advantage of incorporating the linear programming model into a spreadsheet has been the ease of explaining the model to managers with differing levels of mathematical understanding. This has led to a widespread appreciation of the operations research approach for both this application and others.

Source: E. W. Schuster and S. J. Allen, "Raw Material Management at Welch's, Inc.," *Interfaces,* **28**(5): 13–24, Sept.–Oct. 1998. (A link to this article is provided on our website, www.mhhe.com/hillier.)

Name\Define from the Insert menu and type a range name (or click in the name box on the left of the formula bar above the spreadsheet and type a name).

Three questions need to be answered to begin the process of using the spreadsheet to formulate a linear programming model for the problem.

1. What are the *decisions* to be made? For this problem, the necessary decisions are the *production rates* (number of batches produced per week) for the two new products.
2. What are the *constraints* on these decisions? The constraints here are that the number of hours of production time used per week by the two products in the respective plants cannot exceed the number of hours available.
3. What is the overall *measure of performance* for these decisions? Wyndor's overall measure of performance is the *total profit* per week from the two products, so the *objective* is to *maximize* this quantity.

Figure 3.15 shows how these answers can be incorporated into the spreadsheet. Based on the first answer, the *production rates* of the two products are placed in cells C12 and

FIGURE 3.14
The initial spreadsheet for the Wyndor problem after transferring the data from Table 3.1 into data cells.

	A	B	C	D	E	F	G
1	Wyndor Glass Co. Product-Mix Problem						
2							
3			Doors	Windows			
4		Profit Per Batch	$3,000	$5,000			
5							Hours
6			Hours Used Per Batch Produced				Available
7		Plant 1	1	0			4
8		Plant 2	0	2			12
9		Plant 3	3	2			18

	A	B	C	D	E	F	G
1	**Wyndor Glass Co. Product-Mix Problem**						
2							
3			Doors	Windows			
4		Profit Per Batch	$3,000	$5,000			
5					Hours		Hours
6			Hours Used Per Batch Produced		Used		Available
7		Plant 1	1	0	0	<=	4
8		Plant 2	0	2	0	<=	12
9		Plant 3	3	2	0	<=	18
10							
11			Doors	Windows			Total Profit
12		Batches Produced	0	0			$0

■ **FIGURE 3.15** The complete spreadsheet for the Wyndor problem with an initial trial solution (both production rates equal to zero) entered into the changing cells (C12 and D12).

D12 to locate them in the columns for these products just under the data cells. Since we don't know yet what these production rates should be, they are just entered as zeroes at this point. (Actually, any trial solution can be entered, although *negative* production rates should be excluded since they are impossible.) Later, these numbers will be changed while seeking the best mix of production rates. Therefore, these cells containing the decisions to be made are called **changing cells** (or *adjustable cells*). To highlight the changing cells, they are shaded and have a border. (In the spreadsheet files contained in OR Courseware, the changing cells appear in bright yellow on a color monitor.) The changing cells are given the range name BatchesProduced (C12:D12).

Using the answer to question 2, the total number of hours of production time used per week by the two products in the respective plants is entered in cells E7, E8, and E9, just to the right of the corresponding data cells. The Excel equations for these three cells are

$$E7 = C7*C12 + D7*D12$$

$$E8 = C8*C12 + D8*D12$$

$$E9 = C9*C12 + D9*D12$$

where each asterisk denotes multiplication. Since each of these cells provides output that depends on the changing cells (C12 and D12), they are called **output cells.**

Notice that each of the equations for the output cells involves the sum of two products. There is a function in Excel called SUMPRODUCT that will sum up the product of each of the individual terms in two different ranges of cells when the two ranges have the same number of rows and the same number of columns. Each product being summed is the product of a term in the first range and the term in the corresponding location in the second range. For example, consider the two ranges, C7:D7 and C12:D12, so that each range has one row and two columns. In this case, SUMPRODUCT (C7:D7, C12:D12) takes each of the individual terms in the range C7:D7, multiplies them by the corresponding term in the range C12:D12, and then sums up these individual products, as shown in the first equation above. Using the range name BatchesProduced (C12:D12), the formula becomes SUMPRODUCT (C7:D7, BatchesProduced). Although optional with such short equations, this function is especially handy as a shortcut for entering longer equations.

Next, ≤ signs are entered in cells F7, F8, and F9 to indicate that each total value to their left cannot be allowed to exceed the corresponding number in column G. The spreadsheet still will allow you to enter trial solutions that violate the ≤ signs. However, these ≤ signs serve as a reminder that such trial solutions need to be rejected if no changes are made in the numbers in column G.

Finally, since the answer to the third question is that the overall measure of performance is the total profit from the two products, this profit (per week) is entered in cell G12. Much like the numbers in column E, it is the sum of products,

G12 = SUMPRODUCT (C4:D4, C12:D12)

Utilizing range names of TotalProfit (G12), ProfitPerBatch (C4:D4), and BatchesProduced (C12:D12), this equation becomes

TotalProfit = SUMPRODUCT (ProfitPerBatch, BatchesProduced)

This is a good example of the benefit of using range names for making the resulting equation easier to interpret. Rather than needing to refer to the spreadsheet to see what is in cells G12, C4:D4, and C12:D12, the range names immediately reveal what the equation is doing.

TotalProfit (G12) is a special kind of output cell. It is the particular cell that is being targeted to be made as large as possible when making decisions regarding production rates. Therefore, TotalProfit (G12) is referred to as the **target cell** (or *objective cell*). The target cell is shaded darker than the changing cells and is further distinguished by having a heavy border. (In the spreadsheet files contained in OR Courseware, this cell appears in orange on a color monitor.)

The bottom of Fig. 3.16 summarizes all the formulas that need to be entered in the Hours Used column and in the Total Profit cell. Also shown is a summary of the range names (in alphabetical order) and the corresponding cell addresses.

This completes the formulation of the spreadsheet model for the Wyndor problem.

With this formulation, it becomes easy to analyze any trial solution for the production rates. Each time production rates are entered in cells C12 and D12, Excel immediately

FIGURE 3.16 The spreadsheet model for the Wyndor problem, including the formulas for the target cell TotalProfit (G12) and the other output cells in column E, where the objective is to maximize the target cell.

	A	B	C	D	E	F	G
1		Wyndor Glass Co. Product-Mix Problem					
2							
3			Doors	Windows			
4		Profit Per Batch	$3,000	$5,000			
5					Hours		Hours
6			Hours Used Per Batch Produced		Used		Available
7		Plant 1	1	0	0	<=	4
8		Plant 2	0	2	0	<=	12
9		Plant 3	3	2	0	<=	18
10							
11			Doors	Windows			Total Profit
12		Batches Produced	0	0			$0

Range Name	Cells
BatchesProduced	C12:D12
HoursAvailable	G7:G9
HoursUsed	E7:E9
HoursUsedPerBatchProduced	C7:D9
ProfitPerBatch	C4:D4
TotalProfit	G12

	E
5	Hours
6	Used
7	=SUMPRODUCT(C7:D7,BatchesProduced)
8	=SUMPRODUCT(C8:D8,BatchesProduced)
9	=SUMPRODUCT(C9:D9,BatchesProduced)

	G
11	Total Profit
12	=SUMPRODUCT(ProfitPerBatch,BatchesProduced)

calculates the output cells for hours used and total profit. However, it is not necessary to use trial and error. We shall describe next how the Excel Solver can be used to quickly find the optimal solution.

Using the Excel Solver to Solve the Model

Excel includes a tool called Solver that uses the simplex method to find an optimal solution. (A more powerful version of Solver, called *Premium Solver for Education,* also is available in your OR Courseware.)

To access Solver the first time, you need to install it by going to Excel's Add-in menu and adding Solver, after which you will find it on the Data tab (for Excel 2007) or in the Tools menu (for earlier versions of Excel).

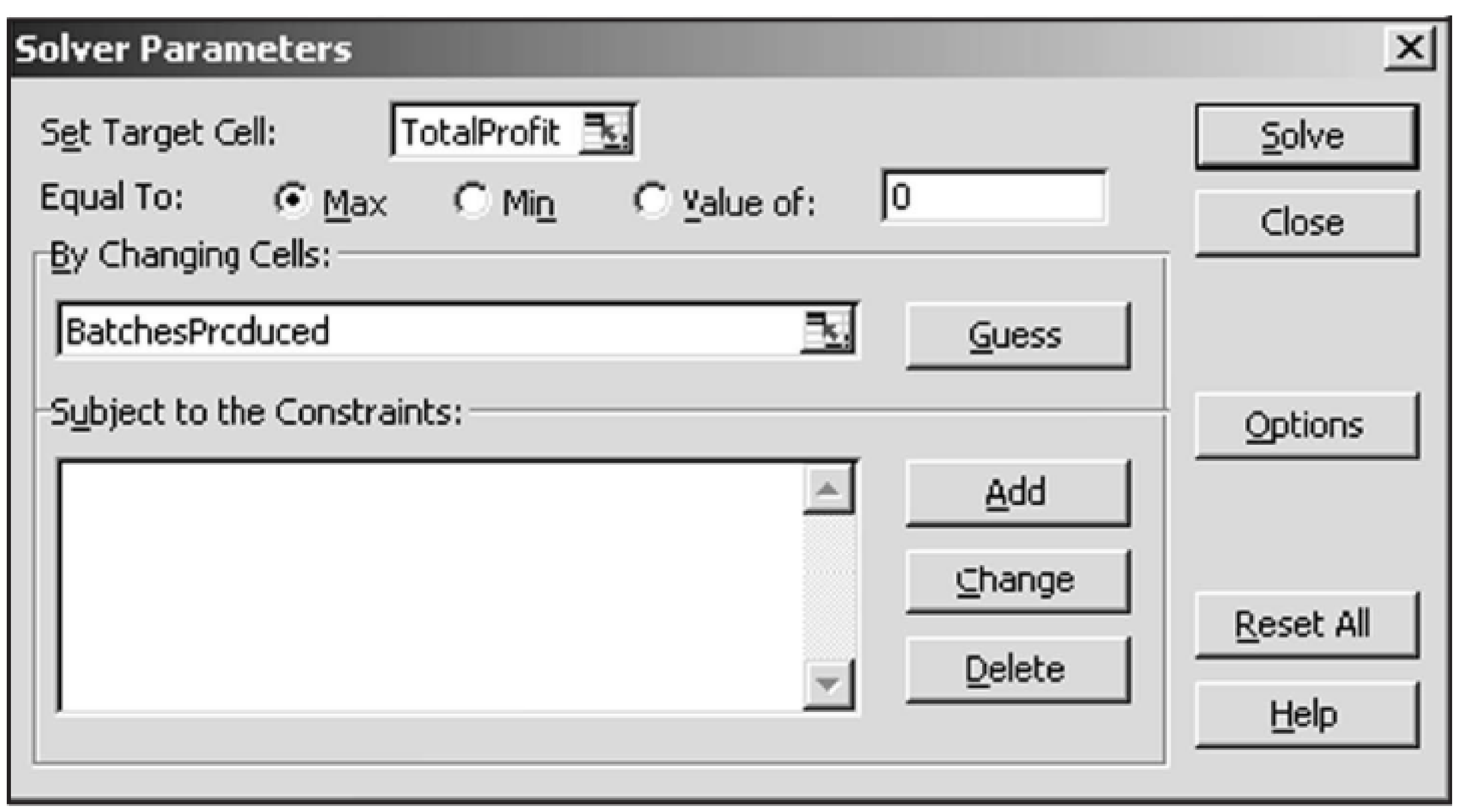
To get started, an arbitrary trial solution has been entered in Fig. 3.16 by placing zeroes in the changing cells. The Solver will then change these to the optimal values after solving the problem.

This procedure is started by choosing Solver. The Solver dialogue box is shown in Fig. 3.17.

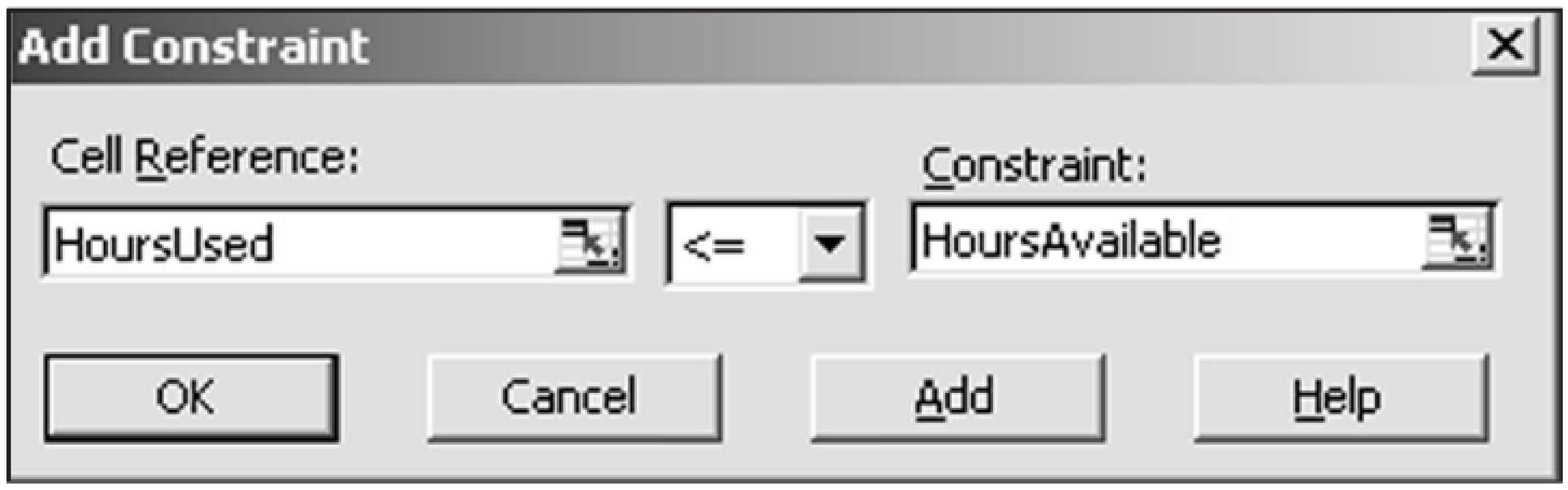
Before the Solver can start its work, it needs to know exactly where each component of the model is located on the spreadsheet. The Solver dialogue box is used to enter this information. You have the choice of typing the range names, typing in the cell addresses, or clicking on the cells in the spreadsheet.[9] Figure 3.17 shows the result of using the first choice, so TotalProfit (rather than G12) has been entered for the target cell and BatchesProduced (rather than the range C12:D12) has been entered for the changing cells. Since the goal is to maximize the target cell, Max also has been selected.

Next, the cells containing the functional constraints need to be specified. This is done by clicking on the Add button on the Solver dialogue box. This brings up the Add Constraint dialogue box shown in Fig. 3.18. The ≤ signs in cells F7, F8, and F9 of Fig. 3.16 are a reminder that the cells in HoursUsed (E7:E9) all need to be less than or equal to the

■ **FIGURE 3.17**
This Solver dialogue box specifies which cells in Fig. 3.16 are the target cell and the changing cells. It also indicates that the target cell is to be maximized.



[9]If you select cells by clicking on them, they will first appear in the dialogue box with their cell addresses and with dollar signs (e.g., $C$9:$D$9). You can ignore the dollar signs. Solver will eventually replace both the cell addresses and the dollar signs with the corresponding range name (if a range name has been defined for the given cell addresses), but only after either adding a constraint or closing and reopening the Solver dialogue box.

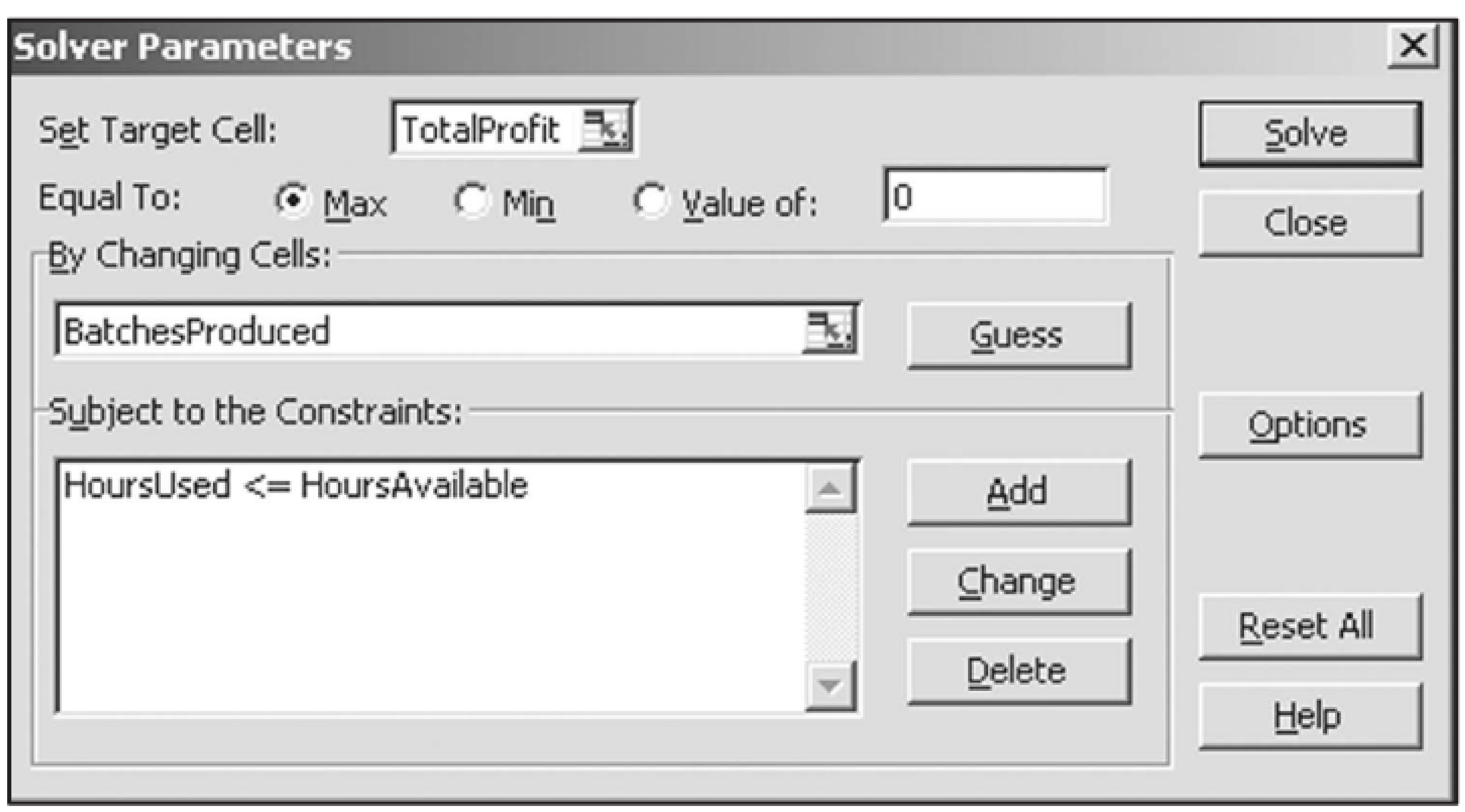


■ **FIGURE 3.18**
The Add Constraint dialogue box after entering the set of constraints, HoursUsed (E7:E9) ≤ HoursAvailable (G7:G9), which specifies that cells E7, E8, and E9 in Fig. 3.16 are required to be less than or equal to cells G7, G8, and G9, respectively.

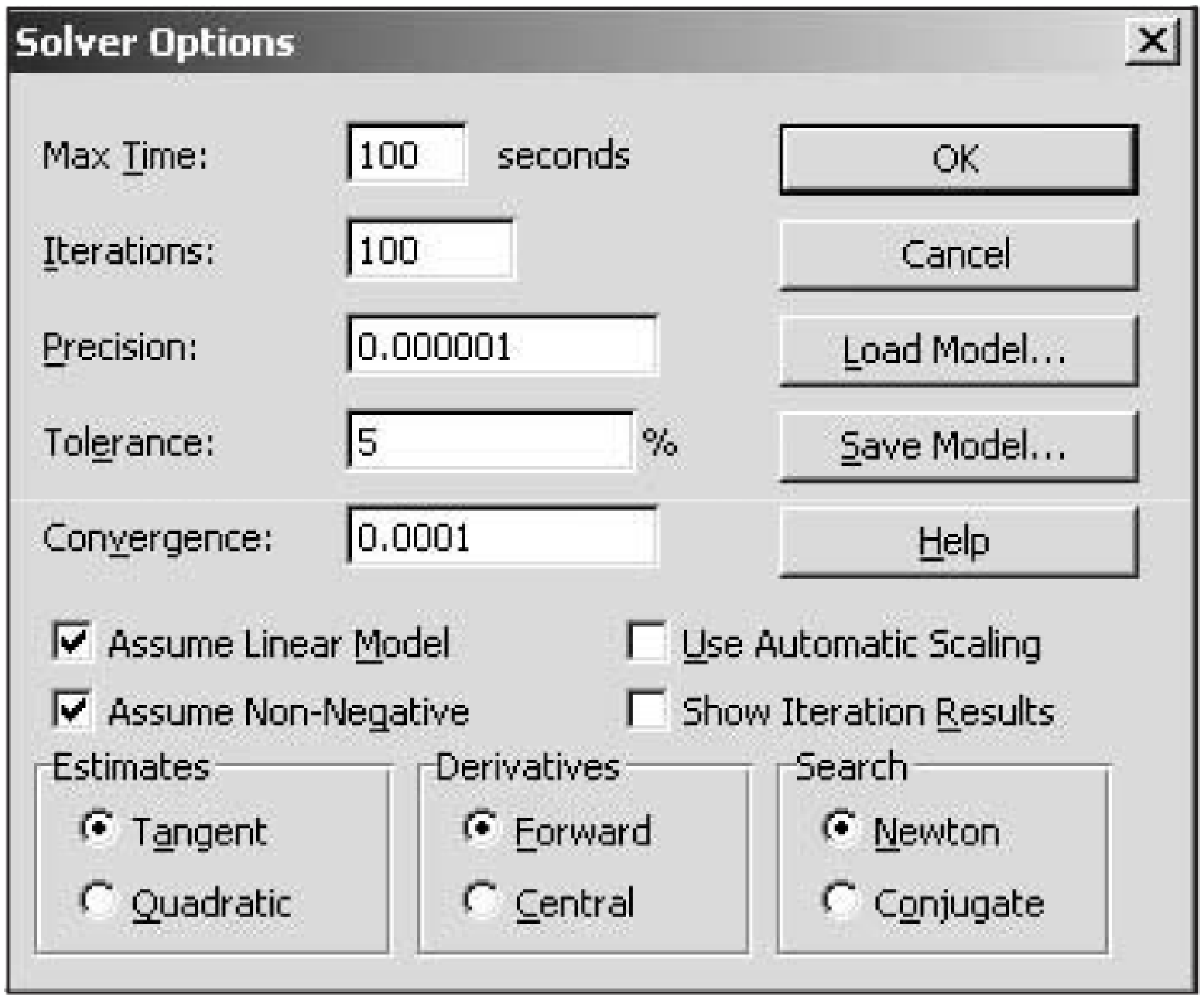
corresponding cells in HoursAvailable (G7:G9). These constraints are specified for the Solver by entering HoursUsed (or E7:E9) on the left-hand side of the Add Constraint dialogue box and HoursAvailable (or G7:G9) on the right-hand side. For the sign between these two sides, there is a menu to choose between <= (less than or equal), =, or >= (greater than or equal), so <= has been chosen. This choice is needed even though ≤ signs were previously entered in column F of the spreadsheet because the Solver only uses the functional constraints that are specified with the Add Constraint dialogue box.

If there were more functional constraints to add, you would click on Add to bring up a new Add Constraint dialogue box. However, since there are no more in this example, the next step is to click on OK to go back to the Solver dialogue box.

The Solver dialogue box now summarizes the complete model (see Fig. 3.19) in terms of the spreadsheet in Fig. 3.16. However, before asking Solver to solve the model, one more step should be taken. Clicking on the Options button brings up the dialogue box shown in Fig. 3.20. This box allows you to specify a number of options about how the problem will be solved. The most important of these are the Assume Linear Model option and the Assume Non-Negative option. Be sure that both options are checked as shown in the figure. This tells Solver that the problem is a *linear* programming problem and that nonnegativity constraints are needed for the changing cells to reject negative production rates. Regarding the other options, accepting the default values shown in the figure usually is fine for small problems. Clicking on the OK button then returns you to the Solver dialogue box.



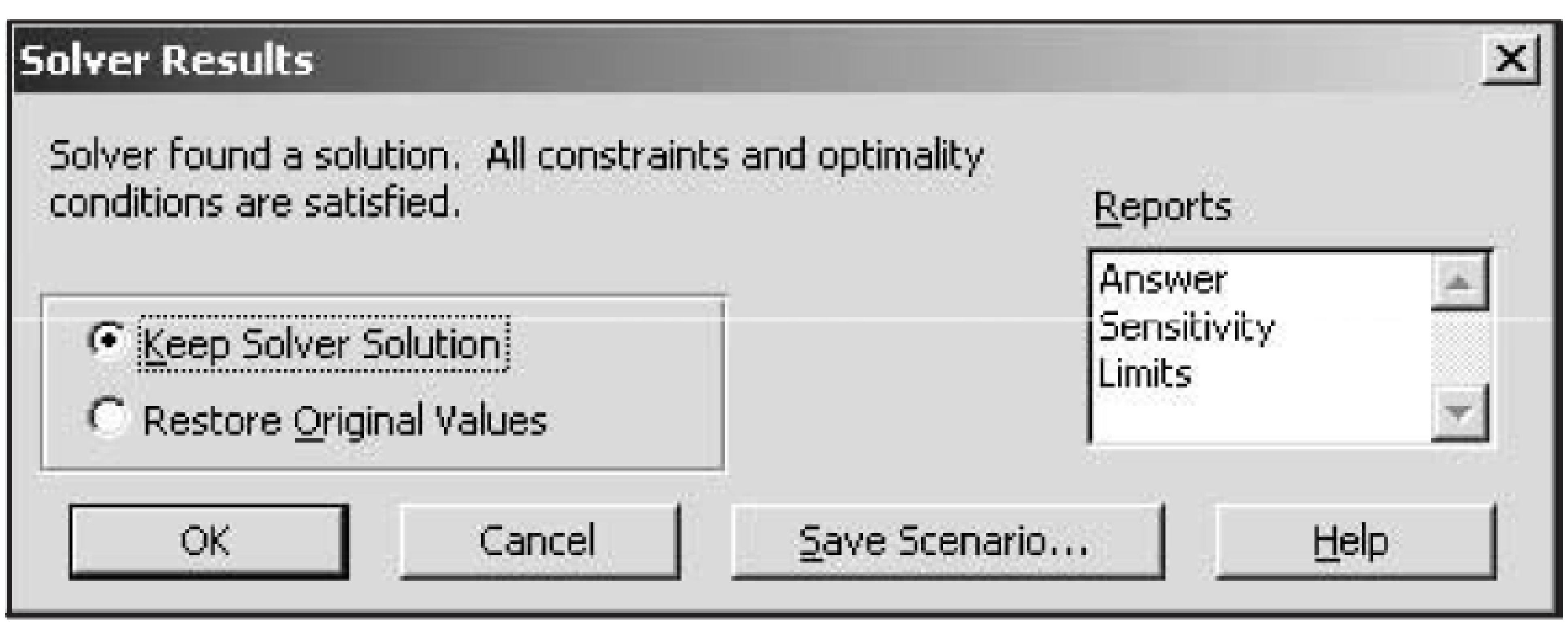
■ **FIGURE 3.19**
The Solver dialogue box after specifying the entire model in terms of the spreadsheet.



■ **FIGURE 3.20**
The Solver Options dialogue box after checking the Assume Linear Model and Assume Non-Negative options to indicate that we wish to solve a linear programming model that has nonnegativity constraints.

Now you are ready to click on Solve in the Solver dialogue box, which will start the process of solving the problem in the background. After a few seconds (for a small problem), Solver will then indicate the outcome. Typically, it will indicate that it has found an optimal solution, as specified in the Solver Results dialogue box shown in Fig. 3.21. If the model has no feasible solutions or no optimal solution, the dialogue box will indicate that instead by stating that "Solver could not find a feasible solution" or that "The Set Cell values do not converge." The dialogue box also presents the option of generating various reports. One of these (the Sensitivity Report) will be discussed later in Secs. 4.7 and 6.8.

■ **FIGURE 3.21**
The Solver Results dialogue box that indicates that an optimal solution has been found.



After solving the model, the Solver replaces the original numbers in the changing cells with the optimal numbers, as shown in Fig. 3.22. Thus, the optimal solution is to produce two batches of doors per week and six batches of windows per week, just as was found by the graphical method in Sec. 3.1. The spreadsheet also indicates the corresponding number in the target cell (a total profit of $36,000 per week), as well as the numbers in the output cells HoursUsed (E7:E9).

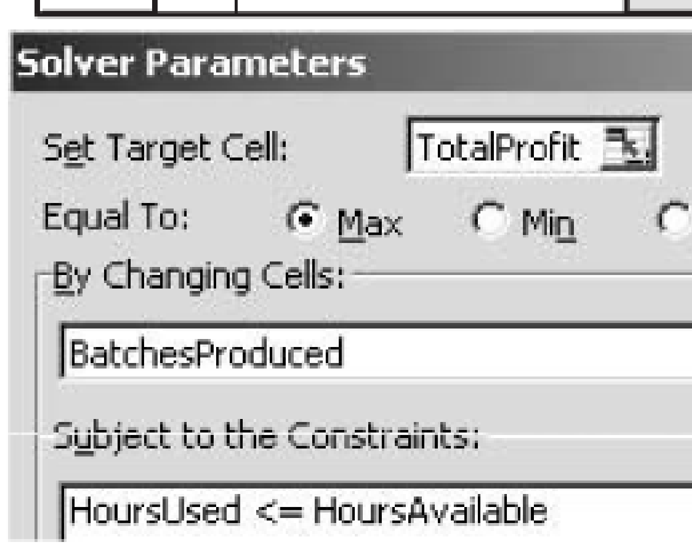
At this point, you might want to check what would happen to the optimal solution if any of the numbers in the data cells were changed to other possible values. This is easy to do because Solver saves all the addresses for the target cell, changing cells, constraints, and so on when you save the file. All you need to do is make the changes you want in the data cells and then click on Solve in the Solver dialogue box again. (Sections 4.7 and 6.8 will focus on this kind of *sensitivity analysis,* including how to use the Solver's Sensitivity Report to expedite this type of what-if analysis.)

To assist you with experimenting with these kinds of changes, your OR Courseware includes Excel files for this chapter (as for others) that provide a complete formulation and solution of the examples here (the Wyndor problem and the ones in Sec. 3.4) in a spreadsheet format. We encourage you to "play" with these examples to see what happens with different data, different solutions, and so forth. You might also find these spreadsheets useful as templates for solving homework problems.

In addition, we suggest that you use this chapter's Excel files to take a careful look at the spreadsheet formulations for some of the examples in Sec. 3.4. This will demonstrate

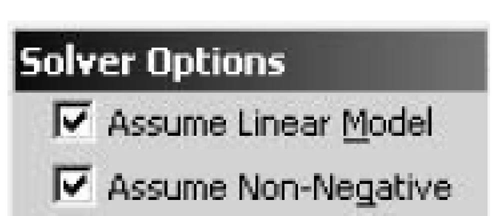
■ **FIGURE 3.22**
The spreadsheet obtained after solving the Wyndor problem.

	A	B	C	D	E	F	G
1		**Wyndor Glass Co. Product-Mix Problem**					
2							
3			Doors	Windows			
4		Profit Per Batch	$3,000	$5,000			
5					Hours		Hours
6			Hours Used Per Batch Produced		Used		Available
7		Plant 1	1	0	2	<=	4
8		Plant 2	0	2	12	<=	12
9		Plant 3	3	2	18	<=	18
10							
11			Doors	Windows			Total Profit
12		Batches Produced	2	6			$36,000



	E
5	Hours
6	Used
7	=SUMPRODUCT(C7:D7,BatchesProduced)
8	=SUMPRODUCT(C8:D8,BatchesProduced)
9	=SUMPRODUCT(C9:D9,BatchesProduced)

	G
11	Total Profit
12	=SUMPRODUCT(ProfitPerBatch,BatchesProduced)



Range Name	Cells
BatchesProduced	C12:D12
HoursAvailable	G7:G9
HoursUsed	E7:E9
HoursUsedPerBatchProduced	C7:D9
ProfitPerBatch	C4:D4
TotalProfit	G12

how to formulate linear programming models in a spreadsheet that are larger and more complicated than for the Wyndor problem.

You will see other examples of how to formulate and solve various kinds of OR models in a spreadsheet in later chapters. The supplementary chapters on the book's website also include a complete chapter (Chap. 21) that is devoted to the art of modeling in spreadsheets. That chapter describes in detail both the general process and the basic guidelines for building a spreadsheet model. It also presents some techniques for debugging such models.

3.6 FORMULATING VERY LARGE LINEAR PROGRAMMING MODELS

Linear programming models come in many different sizes. For the examples in Secs. 3.1 and 3.4, the model sizes range from three functional constraints and two decision variables (for the Wyndor and radiation therapy problems) up to 17 functional constraints and 12 decision variables (for the Save-It Company problem). The latter case may seem like a rather large model. After all, it does take a substantial amount of time just to write down a model of this size. However, by contrast, the models for the application vignettes presented in this chapter are much, much larger. For example, the model for the United Airlines application in Sec. 3.4 often has over 20,000 decision variables.

Such model sizes are not at all unusual. Linear programming models in practice commonly have many hundreds or thousands of functional constraints. In fact, they occasionally will have even millions of functional constraints. The number of decision variables frequently is even larger than the number of functional constraints, and occasionally will range well into the millions.

Formulating such monstrously large models can be a daunting task. Even a "medium-sized" model with a thousand functional constraints and a thousand decision variables has over a million parameters (including the million coefficients in these constraints). It simply is not practical to write out the algebraic formulation, or even to fill in the parameters on a spreadsheet, for such a model.

So how are these very large models formulated in practice? It requires the use of a *modeling language.*

Modeling Languages

A mathematical modeling language is software that has been specifically designed for efficiently formulating large mathematical models, including linear programming models. Even with millions of functional constraints, they typically are of a relatively few types. Similarly, the decision variables will fall into a small number of categories. Therefore, using large blocks of data in databases, a modeling language will use a single expression to simultaneously formulate all the constraints of the same type in terms of the variables of each type. We will illustrate this process soon.

In addition to efficiently formulating large models, a modeling language will expedite a number of model management tasks, including accessing data, transforming data into model parameters, modifying the model whenever desired, and analyzing solutions from the model. It also may produce summary reports in the vernacular of the decision makers, as well as document the model's contents.

Several excellent modeling languages have been developed over the last couple of decades. These include AMPL, MPL, OPL, GAMS, and LINGO.

The student version of one of these, **MPL** (short for Mathematical Programming Language), is provided for you on the book's website along with extensive tutorial material. As subsequent versions are released in future years, the latest student version also can be downloaded from the website, maximalsoftware.com. MPL is a product of Maximal Software, Inc. One feature is extensive support for Excel in MPL. This includes both importing and exporting Excel ranges from MPL. Full support also is provided for the Excel VBA macro language through OptiMax 2000. (The student version of OptiMax 2000 is on the book's website as well.) This product allows the user to fully integrate MPL models into Excel and solve with any of the powerful solvers that MPL supports, including **CPLEX** (described in Sec. 4.8).

LINGO is a product of LINDO Systems, Inc., which also markets a spreadsheet-add-in optimizer called What's*Best!* that is designed for large industrial problems, as well as a callable subroutine library called the LINDO API. The LINGO software includes as a subset the LINDO interface that has been a popular introduction to linear programming for many people. The student version of LINGO with the LINDO interface is part of the software included on the book's website. All of the LINDO Systems products can also be downloaded from www.lindo.com. Like MPL, LINGO is a powerful general-purpose modeling language. A notable feature of LINGO is its great flexibility for dealing with a wide variety of OR problems in addition to linear programming. For example, when dealing with highly nonlinear models, it contains a global optimizer that will find a globally optimal solution. (More about this in Sec. 12.10.) New to the current edition of this book, the latest LINGO also has a built-in compatible programming language so that you can do things like solve several different optimization problems as part of one run, which is particularly useful when doing parametric analysis.

The book's website includes MPL, LINGO and LINDO formulations for essentially every example in this book to which these modeling languages and optimizers can be applied.

Now let us look at a simplified example that illustrates how a very large linear programming model can arise.

An Example of a Problem with a Huge Model

Management of the WORLDWIDE CORPORATION needs to address a *product-mix problem,* but one that is vastly more complex than the Wyndor product-mix problem introduced in Sec. 3.1. This corporation has 10 plants in various parts of the world. Each of these plants produces the same 10 products and then sells them within its region. The *demand* (sales potential) for each of these products from each plant is known for each of the next 10 months. Although the amount of a product sold by a plant in a given month cannot exceed the demand, the amount produced can be larger, where the excess amount would be stored in inventory (at some unit cost per month) for sale in a later month. Each unit of each product takes the same amount of space in inventory, and each plant has some upper limit on the total number of units that can be stored (the *inventory capacity*).

Each plant has the same 10 production processes (we'll refer to them as *machines*), each of which can be used to produce any of the 10 products. Both the production cost per unit of a product and the production rate of the product (number of units produced per day devoted to that product) depend on the combination of plant and machine involved (but not the month). The number of working days (*production days available*) varies somewhat from month to month.

Since some plants and machines can produce a particular product either less expensively or at a faster rate than other plants and machines, it is sometimes worthwhile to ship some units of the product from one plant to another for sale by the latter plant. For each combination of a plant being shipped from (the *fromplant*) and a plant being shipped to (the *toplant*), there is a certain cost per unit shipped of any product, where this unit shipping cost is the same for all the products.

Management now needs to determine how much of each product should be produced by each machine in each plant during each month, as well as how much each plant should sell of each product in each month and how much each plant should ship of each product in each month to each of the other plants. Considering the worldwide price for each product, the objective is to find the feasible plan that maximizes the total profit (total sales revenue *minus* the sum of the total production costs, inventory costs, and shipping costs).

We should note again that this is a simplified example in a number of ways. We have assumed that the number of plants, machines, products, and months are exactly the same (10). In most real situations, the number of products probably will be far larger and the planning horizon is likely to be considerably longer than 10 months, whereas the number of "machines" (types of production processes) may be less than 10. We also have assumed that every plant has all the same types of machines (production processes) and every machine type can produce every product. In reality, the plants may have some differences in terms of their machine types and the products they are capable of producing. The net result is that the corresponding model for some corporations may be smaller than the one for this example, but the model for other corporations may be considerably larger (perhaps even vastly larger) than this one.

The Structure of the Resulting Model

Because of the inventory costs and the limited inventory capacities, it is necessary to keep track of the amount of each product kept in inventory in each plant during each month. Consequently, the linear programming model has four types of decision variables: production quantities, inventory quantities, sales quantities, and shipping quantities. With 10 plants, 10 machines, 10 products, and 10 months, this gives a total of 21,000 decision variables, as outlined below.

Decision Variables.

10,000 production variables: one for each combination of a plant, machine, product, and month
1,000 inventory variables: one for each combination of a plant, product, and month
1,000 sales variables: one for each combination of a plant, product, and month
9,000 shipping variables: one for each combination of a product, month, plant (the fromplant), and another plant (the toplant)

Multiplying each of these decision variables by the corresponding unit cost or unit revenue, and then summing over each type, the following objective function can be calculated:

Objective Function.

$$\text{Maximize} \quad \text{profit} = \text{total sales revenue} - \text{total cost},$$

where

$$\text{Total cost} = \text{total production cost} + \text{total inventory cost} + \text{total shipping cost}.$$

When maximizing this objective function, the 21,000 decision variables need to satisfy nonnegativity constraints as well as four types of functional constraints—production capacity constraints, plant balance constraints (equality constraints that provide appropriate values to the inventory variables), maximum inventory constraints, and maximum sales constraints. As enumerated below, there are a total of 3,100 functional constraints, but all the constraints of each type follow the same pattern.

Functional Constraints.

1,000 production capacity constraints (one for each combination of a plant, machine, and month):

Production days used ≤ production days available,

where the left-hand side is the sum of 10 fractions, one for each product, where each fraction is that product's production quantity (a decision variable) *divided* by the product's production rate (a given constant).

1,000 plant balance constraints (one for each combination of a plant, product, and month):

Amount produced + inventory last month + amount shipped in = sales + current inventory + amount shipped out,

where the *amount produced* is the sum of the decision variables representing the production quantities at the machines, the *amount shipped in* is the sum of the decision variables representing the shipping quantities in from the other plants, and the *amount shipped out* is the sum of the decision variables representing the shipping quantities out to the other plants.

100 maximum inventory constraints (one for each combination of a plant and month):

Total inventory ≤ inventory capacity,

where the left-hand side is the sum of the decision variables representing the inventory quantities for the individual products.

1,000 maximum sales constraints (one for each combination of a plant, product, and month):

Sales ≤ demand.

Now let us see how the MPL Modeling Language can formulate this huge model very compactly.

Formulation of the Model in MPL

The modeler begins by assigning a title to the model and listing an *index* for each of the entities of the problem, as illustrated below.

```
TITLE
  Production_Planning;

INDEX
   product   := (A1, A2, A3, A4, A5, A6, A7, A8, A9, A10);
   month     := (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct);
   plant     := (p1, p2, p3, p4, p5, p6, p7, p8, p9, p10);
   fromplant := plant;
   toplant   := plant;
   machine   := (m1, m2, m3, m4, m5, m6, m7, m8, m9, m10);
```

Except for the months, the entries on the right-hand side are arbitrary labels for the respective products, plants, and machines, where these same labels are used in the data files. Note that a colon is placed after the name of each entry and a semicolon is placed at the end of each statement (but a statement is allowed to extend over more than one line).

A big job with any large model is collecting and organizing the various types of data into data files. A data file can be in either dense format or sparse format. In *dense format,* the file will contain an entry for every combination of all possible values of the respective indexes. For example, suppose that the data file contains the production rates for producing the various products with the various machines (production processes) in the various plants. In dense format, the file will contain an entry for every combination of a plant, a machine, and a product. However, the entry may need to be zero for most of the combinations because that particular plant may not have that particular machine or, even if it does, that particular machine may not be capable of producing that particular product in that particular plant. The percentage of the entries in dense format that are *nonzero* is referred to as the *density* of the data set. In practice, it is common for large data sets to have a density under 5 percent, and it frequently is under 1 percent. Data sets with such a low density are referred to as being *sparse.* In such situations, it is more efficient to use a data file in *sparse format.* In this format, only the nonzero values (and an identification of the index values they refer to) are entered into the data file. Generally, data are entered in sparse format either from a text file or from corporate databases. The ability to handle sparse data sets efficiently is one key for successfully formulating and solving large-scale optimization models. MPL can readily work with data in either dense format or sparse format.

In the Worldwide Corp. example, eight data files are needed to hold the product prices, demands, production costs, production rates, production days available, inventory costs, inventory capacities, and shipping costs. We assume that these data files are available in sparse format. The next step is to give a brief suggestive name to each one and to identify (inside square brackets) the index or indexes for that type of data, as shown below.

```
DATA
Price[product]      := SPARSEFILE("Price.dat");
Demand[plant,   product,   month] := SPARSEFILE("Demand.dat");
ProdCost[plant, machine, product] := SPARSEFILE("Produce.dat", 4);
ProdRate[plant, machine, product] := SPARSEFILE("Produce.dat", 5);
ProdDaysAvail[month]     := SPARSEFILE("ProdDays.dat");
InvtCost[plant, product] := SPARSEFILE("InvtCost.dat");
InvtCapacity[plant]      := SPARSEFILE("InvtCap.dat");
ShipCost[fromplant, toplant]     := SPARSEFILE ("ShipCost.dat");
```

To illustrate the contents of these data files, consider the one that provides production costs and production rates. Here is a sample of the first few entries of SPARSEFILE produce.dat:

```
!
   ! Produce.dat - Production Cost and Rate
   !
   ! ProdCost[plant, machine, product]:
   ! ProdRate[plant, machine, product]:
   !
     p1, m11, A1, 73.30, 500,
     p1, m11, A2, 52.90, 450,
     p1, m12, A3, 65.40, 550,
     p1, m13, A3, 47.60, 350,
```

Next, the modeler gives a short name to each type of decision variable. Following the name, inside square brackets, is the index or indexes over which the subscripts run.

```
VARIABLES
  Produce[plant, machine, product, month]        -> Prod;
  Inventory[plant, product, month]               -> Invt;
  Sales[plant, product, month]                   -> Sale;
  Ship[product, month, fromplant, toplant]
      WHERE (fromplant <> toplant);
```

In the case of the decision variables with names longer than four letters, the arrows on the right point to four-letter abbreviations to fit the size limitations of many solvers. The last line indicates that the fromplant subscript and toplant subscript are not allowed to have the same value.

There is one more step before writing down the model. To make the model easier to read, it is useful first to introduce *macros* to represent the summations in the objective function.

```
MACROS
  Total Revenue   := SUM(plant, product, month: Price*Sales);
  TotalProdCost   := SUM(plant, machine, product, month:
                     ProdCost*Produce);
  TotalInvtCost   := SUM(plant, product, month:
                     InvtCost*Inventory);
  TotalShipCost   := SUM(product, month, fromplant, toplant:
                     ShipCost*Ship);
  TotalCost       := TotalProdCost + TotalInvtCost + TotalShipCost;
```

The first four macros use the MPL keyword SUM to execute the summation involved. Following each SUM keyword (inside the parentheses) is, first, the index or indexes over which the summation runs. Next (after the colon) is the vector product of a data vector (one of the data files) times a variable vector (one of the four types of decision variables).

Now this model with 3,100 functional constraints and 21,000 decision variables can be written down in the following compact form.

```
MODEL

  MAX Profit = TotalRevenue - TotalCost;

SUBJECT TO
  ProdCapacity[plant, machine, month] -> PCap:
    SUM(product: Produce/ProdRate) <= ProdDaysAvail;
  PlantBal[plant, product, month] -> PBal:
       SUM(machine: Produce) + Inventory [month - 1]
     + SUM(fromplant: Ship[fromplant, toplant:= plant])
  =
       Sales + Inventory
     + SUM(toplant: Ship[fromplant:= plant, toplant]);

  MaxInventory [plant, month] -> MaxI:
    SUM(product: Inventory) <= InvtCapacity;

BOUNDS
    Sales <= Demand;
END
```

For each of the four types of constraints, the first line gives the name for this type. There is one constraint of this type for each combination of values for the indexes inside the square brackets following the name. To the right of the brackets, the arrow points to a four-letter abbreviation of the name that a solver can use. Below the first line, the general form of constraints of this type is shown by using the SUM operator.

For each production capacity constraint, each term in the summation consists of a decision variable (the production quantity of that product on that machine in that plant during that month) divided by the corresponding production rate, which gives the number of production days being used. Summing over the products then gives the total number of production days being used on that machine in that plant during that month, so this number must not exceed the number of production days available.

The purpose of the plant balance constraint for each plant, product, and month is to give the correct value to the current inventory variable, given the values of all the other decision variables including the inventory level for the preceding month. Each of the SUM operators in these constraints involves simply a sum of decision variables rather than a vector product. This is the case also for the SUM operator in the maximum inventory constraints. By contrast, the left-hand side of the maximum sales constraints is just a single decision variable for each of the 1,000 combinations of a plant, product, and month. (Separating these upper-bound constraints on individual variables from the regular functional constraints is advantageous because of the computational efficiencies that can be obtained by using the *upper bound technique* described in Sec. 7.3.) No lower-bound constraints are shown here because MPL automatically assumes that all 21,000 decision variables have nonnegativity constraints unless nonzero lower bounds are specified. For each of the 3,100 functional constraints, note that the left-hand side is a linear function of the decision variables and the right-hand side is a constant taken from the appropriate data file. Since the objective function also is a linear function of the decision variables, this model is a legitimate linear programming model.

To solve the model, MPL supports various leading **solvers** (software packages for solving linear programming models and related models) that can be installed into MPL. As discussed in Sec. 4.8, **CPLEX** is a particularly prominent and powerful solver. The version of MPL in your OR Courseware already has installed the student version of CPLEX, which uses the simplex method to solve linear programming models. Therefore, to solve such a model formulated with MPL, all you have to do is choose *Solve CPLEX* from the *Run* menu or press the *Run Solve* button in the *Toolbar.* You then can display the solution file in a view window by pressing the *View* button at the bottom of the *Status Window.*

This brief introduction to MPL illustrates the ease with which modelers can use modeling languages to formulate huge linear programming models in a clear, concise way. To assist you in using MPL, an MPL Tutorial is included on the book's website. This tutorial goes through all the details of formulating smaller versions of the production planning example considered here. You also can see elsewhere on the book's website how all the other linear programming examples in this chapter and subsequent chapters would be formulated with MPL and solved by CPLEX.

The LINGO Modeling Language

LINGO is another popular modeling language featured in this book. The company, LINDO Systems, that produces LINGO first became known for the easy-to-use optimizer, **LINDO,** which is a subset of the LINGO software. LINDO Systems also produces a spreadsheet solver, **What's*Best!*,** and a callable solver library, the **LINDO API.** The student version of LINGO is provided to you on the book's website. (The latest trial versions of all of the above can be downloaded from www.lindo.com.) Both LINDO and What's*Best!* share the LINDO API as the solver engine. The LINDO API has solvers based on the simplex method and interior-point/barrier solvers (such as discussed in Secs. 4.9 and 7.4), plus a global solver for solving nonlinear models.

Like MPL, LINGO enables a modeler to efficiently formulate a huge model in a clear compact fashion that separates the data from the model formulation. This separation means that as changes occur in the data describing the problem that needs to be solved from day to day (or even minute to minute), the user needs to change only the data and not be concerned with the model formulation. You can develop a model on a small data set and then when you supply the model with a large data set, the model formulation adjusts automatically to the new data set.

LINGO uses *sets* as a fundamental concept. For example, in the Worldwide Corp. production planning problem, the simple or "primitive" sets of interest are products, plants, machines, and months. Each member of a set may have one or more *attributes* associated with it, such as the price of a product, the inventory capacity of a plant, the production rate of a machine, and the number of production days available in a month. Some of these attributes are input data, while others, such as production and shipping quantities, are decision variables for the model. One can also define derived sets that are built from combinations of other sets. As with MPL, the SUM operator is commonly used to write the objective function and constraints in a compact form.

There is a hard copy manual available for LINGO. This entire manual also is available directly in LINGO via the Help command and can be searched in a variety of ways.

A supplement to this chapter on the book's website describes LINGO further and illustrates its use on a couple of small examples. A second supplement shows how LINGO can be used to formulate the model for the Worldwide Corp. production planning example. A LINGO tutorial on the website provides the details needed for doing basic modeling with this modeling language. The LINGO formulations and solutions for the various examples in both this chapter and many other chapters also are included on the website.

3.7 CONCLUSIONS

Linear programming is a powerful technique for dealing with the problem of allocating limited resources among competing activities as well as other problems having a similar mathematical formulation. It has become a standard tool of great importance for numerous business and industrial organizations. Furthermore, almost any social organization is concerned with allocating resources in some context, and there is a growing recognition of the extremely wide applicability of this technique.

However, not all problems of allocating limited resources can be formulated to fit a linear programming model, even as a reasonable approximation. When one or more of the assumptions of linear programming is violated seriously, it may then be possible to apply another mathematical programming model instead, e.g., the models of integer programming (Chap. 11) or nonlinear programming (Chap. 12).

SELECTED REFERENCES

1. Baker, K. R.: *Optimization Modeling with Spreadsheets,* Duxbury, Pacific Grove, CA, 2006.
2. Hillier, F. S., and M. S. Hillier: *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets,* 3rd ed., McGraw-Hill/Irwin, Burr Ridge, IL, 2008, chaps. 2, 3.
3. *LINGO User's Guide,* LINDO Systems, Inc., Chicago, IL, 2008.
4. *MPL Modeling System* (*Release* 4.2) manual, Maximal Software, Inc., Arlington, VA, e-mail: info@maximalsoftware.com, 2008.
5. Schrage, L.: *Optimization Modeling with LINGO,* LINDO Systems Press, Chicago, IL, 2008.
6. Williams, H. P.: *Model Building in Mathematical Programming,* 4th ed., Wiley, New York, 1999.

Some Award-Winning Applications of Linear Programming:

(A link to all these articles is provided on our website, www.mhhe.com/hillier.)

A1. Ambs, K., S. Cwilich, M. Deng, D. J. Houck, D. F. Lynch, and D. Yan: "Optimizing Restoration Capacity in the AT&T Network," *Interfaces,* **30**(1): 26–44, January–February 2000.
A2. Caixeta-Filho, J. V., J. M. van Swaay-Neto, and A. de P. Wagemaker: "Optimization of the Production Planning and Trade of Lily Flowers at Jan de Wit Company," *Interfaces,* **32**(1): 35–46, January–February 2002.

A3. Chalermkraivuth, K. C., S. Bollapragada, M. C. Clark, J. Deaton, L. Kiaer, J. P. Murdzek, W. Neeves, B. J. Scholz, and D. Toledano: "GE Asset Management, Genworth Financial, and GE Insurance Use a Sequential-Linear-Programming Algorithm to Optimize Portfolios, *Interfaces,* **35**(5): 370–380, September–October 2005.

A4. Elimam, A. A., M. Girgis, and S. Kotob: "A Solution to Post Crash Debt Entanglements in Kuwait's al-Manakh Stock Market," *Interfaces,* **27**(1): 89–106, January–February 1997.

A5. Epstein, R., R. Morales, J. Serón, and A. Weintraub: "Use of OR Systems in the Chilean Forest Industries," *Interfaces,* **29**(1): 7–29, January–February 1999.

A6. Geraghty, M. K., and E. Johnson: "Revenue Management Saves National Car Rental," *Interfaces,* **27**(1): 107–127, January–February 1997.

A7. Leachman, R. C., R. F. Benson, C. Liu, and D. J. Raar: "IMPReSS: An Automated Production-Planning and Delivery-Quotation System at Harris Corporation—Semiconductor Sector," *Interfaces,* **26**(1): 6–37, January–February 1996.

A8. Mukuch, W. M., J. L. Dodge, J. G. Ecker, D. C. Granfors, and G. J. Hahn: "Managing Consumer Credit Delinquency in the U.S. Economy: A Multi-Billion Dollar Management Science Application," *Interfaces,* **22**(1): 90–109, January–February 1992.

A9. Murty, K. G., Y.-w. Wan, J. Liu, M. M. Tseng, E. Leung, K.-K. Lai, and H. W. C. Chiu: "Hongkong International Terminals Gains Elastic Capacity Using a Data-Intensive Decision-Support System," *Interfaces,* **35**(1): 61–75, January–February 2005.

A10. Yoshino, T., T. Sasaki, and T. Hasegawa: "The Traffic-Control System on the Hanshin Expressway," *Interfaces,* **25**(1): 94–108, January–February 1995.

LEARNING AIDS FOR THIS CHAPTER ON OUR WEBSITE (www.mhhe.com/hillier)

Worked Examples:

Examples for Chapter 3

A Demonstration Example in OR Tutor:

Graphical Method

Procedures in IOR Tutorial:

Interactive Graphical Method
Graphical Method and Sensitivity Analysis

An Excel Add-In:

Premium Solver for Education

"Ch. 3—Intro to LP" Files for Solving the Examples:

Excel Files
LINGO/LINDO File
MPL/CPLEX File

Glossary for Chapter 3

Supplements to This Chapter:

The LINGO Modeling Language
More About LINGO.

See Appendix 1 for documentation of the software.

PROBLEMS

The symbols to the left of some of the problems (or their parts) have the following meaning:

D: The demonstration example listed above may be helpful.
I: You may find it helpful to use the corresponding procedure in IOR Tutorial (the printout records your work).
C: Use the computer to solve the problem by applying the simplex method. The available software options for doing this include the Excel Solver or Premium Solver (Sec. 3.5), MPL/CPLEX (Sec. 3.6), LINGO (Supplements 1 and 2 to this chapter on the book's website and Appendix 4.1), and LINDO (Appendix 4.1), but follow any instructions given by your instructor regarding the option to use.

An asterisk on the problem number indicates that at least a partial answer is given in the back of the book.

3.1-1. Read the referenced article that fully describes the OR study summarized in the application vignette presented in Sec. 3.1. Briefly describe how linear programming was applied in this study. Then list the various financial and nonfinancial benefits that resulted from this study.

D **3.1-2.*** For each of the following constraints, draw a separate graph to show the nonnegative solutions that satisfy this constraint.
(a) $x_1 + 3x_2 \le 6$
(b) $4x_1 + 3x_2 \le 12$
(c) $4x_1 + x_2 \le 8$
(d) Now combine these constraints into a single graph to show the feasible region for the entire set of functional constraints plus nonnegativity constraints.

D **3.1-3.** Consider the following objective function for a linear programming model:

$$\text{Maximize } Z = 2x_1 + 3x_2$$

(a) Draw a graph that shows the corresponding objective function lines for $Z = 6$, $Z = 12$, and $Z = 18$.
(b) Find the slope-intercept form of the equation for each of these three objective function lines. Compare the slope for these three lines. Also compare the intercept with the x_2 axis.

3.1-4. Consider the following equation of a line:

$$60x_1 + 40x_2 = 600$$

(a) Find the slope-intercept form of this equation.
(b) Use this form to identify the slope and the intercept with the x_2 axis for this line.
(c) Use the information from part (*b*) to draw a graph of this line.

D,I **3.1-5.*** Use the graphical method to solve the problem:

$$\text{Maximize} \quad Z = 2x_1 + x_2,$$

subject to

$$\begin{aligned} x_2 &\le 10 \\ 2x_1 + 5x_2 &\le 60 \\ x_1 + x_2 &\le 18 \\ 3x_1 + x_2 &\le 44 \end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

D,I **3.1-6.** Use the graphical method to solve the problem:

$$\text{Maximize} \quad Z = 10x_1 + 20x_2,$$

subject to

$$\begin{aligned} -x_1 + 2x_2 &\le 15 \\ x_1 + x_2 &\le 12 \\ 5x_1 + 3x_2 &\le 45 \end{aligned}$$

and

$$x_1 \ge 0, \ x_2 \ge 0.$$

3.1-7. The Whitt Window Company is a company with only thr employees which makes two different kinds of hand-crafted wi dows: a wood-framed and an aluminum-framed window. They ea $180 profit for each wood-framed window and $90 profit for ea aluminum-framed window. Doug makes the wood frames, and ca make 6 per day. Linda makes the aluminum frames, and can mal 4 per day. Bob forms and cuts the glass, and can make 48 squa feet of glass per day. Each wood-framed window uses 6 square fe of glass and each aluminum-framed window uses 8 square feet glass.

The company wishes to determine how many windows of ea type to produce per day to maximize total profit.
(a) Describe the analogy between this problem and the Wynd Glass Co. problem discussed in Sec. 3.1. Then construct a fill in a table like Table 3.1 for this problem, identifying bo the activities and the resources.
(b) Formulate a linear programming model for this problem.
D,I **(c)** Use the graphical method to solve this model.
I **(d)** A new competitor in town has started making wood-fram windows as well. This may force the company to lower tl price they charge and so lower the profit made for each woo framed window. How would the optimal solution change (if all) if the profit per wood-framed window decreases from $18 to $120? From $180 to $60? (You may find it helpful to u the Graphical Analysis and Sensitivity Analysis procedure IOR Tutorial.)
I **(e)** Doug is considering lowering his working hours, which wou decrease the number of wood frames he makes per day. Ho would the optimal solution change if he makes only 5 woo frames per day? (You may find it helpful to use the Graphic Analysis and Sensitivity Analysis procedure in IOR Tutoria

3.1-8. The WorldLight Company produces two light fixtur (products 1 and 2) that require both metal frame parts and electric components. Management wants to determine how many units of each product to produce so as to maximize profit. For each unit of product 1, 1 unit of frame parts and 2 units of electrical components are required. For each unit of product 2, 3 units of frame parts and 2 units of electrical components are required. The company has 200 units of frame parts and 300 units of electrical components. Each unit of product 1 gives a profit of \$1, and each unit of product 2, up to 60 units, gives a profit of \$2. Any excess over 60 units of product 2 brings no profit, so such an excess has been ruled out.

(a) Formulate a linear programming model for this problem.

D,I **(b)** Use the graphical method to solve this model. What is the resulting total profit?

3.1-9. The Primo Insurance Company is introducing two new product lines: special risk insurance and mortgages. The expected profit is \$5 per unit on special risk insurance and \$2 per unit on mortgages.

Management wishes to establish sales quotas for the new product lines to maximize total expected profit. The work requirements are as follows:

	Work-Hours per Unit		
Department	**Special Risk**	**Mortgage**	**Work-Hours Available**
Underwriting	3	2	2400
Administration	0	1	800
Claims	2	0	1200

(a) Formulate a linear programming model for this problem.

D,I **(b)** Use the graphical method to solve this model.

(c) Verify the exact value of your optimal solution from part (*b*) by solving algebraically for the simultaneous solution of the relevant two equations.

3.1-10. Weenies and Buns is a food processing plant which manufactures hot dogs and hot dog buns. They grind their own flour for the hot dog buns at a maximum rate of 200 pounds per week. Each hot dog bun requires 0.1 pound of flour. They currently have a contract with Pigland, Inc., which specifies that a delivery of 800 pounds of pork product is delivered every Monday. Each hot dog requires $\frac{1}{4}$ pound of pork product. All the other ingredients in the hot dogs and hot dog buns are in plentiful supply. Finally, the labor force at Weenies and Buns consists of 5 employees working full time (40 hours per week each). Each hot dog requires 3 minutes of labor, and each hot dog bun requires 2 minutes of labor. Each hot dog yields a profit of \$0.80, and each bun yields a profit of \$0.30.

Weenies and Buns would like to know how many hot dogs and how many hot dog buns they should produce each week so as to achieve the highest possible profit.

(a) Formulate a linear programming model for this problem.

D,I **(b)** Use the graphical method to solve this model.

3.1-11.* The Omega Manufacturing Company has discontinued the production of a certain unprofitable product line. This act created considerable excess production capacity. Management is considering devoting this excess capacity to one or more of three products; call them products 1, 2, and 3. The available capacity on the machines that might limit output is summarized in the following table:

Machine Type	Available Time (Machine Hours per Week)
Milling machine	500
Lathe	350
Grinder	150

The number of machine hours required for each unit of the respective products is

Productivity coefficient (in machine hours per unit)

Machine Type	Product 1	Product 2	Product 3
Milling machine	9	3	5
Lathe	5	4	0
Grinder	3	0	2

The sales department indicates that the sales potential for products 1 and 2 exceeds the maximum production rate and that the sales potential for product 3 is 20 units per week. The unit profit would be \$50, \$20, and \$25, respectively, on products 1, 2, and 3. The objective is to determine how much of each product Omega should produce to maximize profit.

(a) Formulate a linear programming model for this problem.

C **(b)** Use a computer to solve this model by the simplex method.

D **3.1-12.** Consider the following problem, where the value of c_1 has not yet been ascertained.

$$\text{Maximize} \quad Z = c_1x_1 + x_2,$$

subject to

$$x_1 + x_2 \le 6$$
$$x_1 + 2x_2 \le 10$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

Use graphical analysis to determine the optimal solution(s) for (x_1, x_2) for the various possible values of $c_1 (-\infty < c_1 < \infty)$.

D **3.1-13.** Consider the following problem, where the value of k has not yet been ascertained.

$$\text{Maximize} \quad Z = x_1 + 2x_2,$$

subject to

$$-x_1 + x_2 \le 2$$
$$x_2 \le 3$$
$$kx_1 + x_2 \le 2k + 3, \quad \text{where } k \ge 0$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

The solution currently being used is $x_1 = 2$, $x_2 = 3$. Use graphical analysis to determine the values of k such that this solution actually is optimal.

D **3.1-14.** Consider the following problem, where the values of c_1 and c_2 have not yet been ascertained.

$$\text{Maximize} \quad Z = c_1x_1 + c_2x_2,$$

subject to

$$2x_1 + x_2 \le 11$$
$$-x_1 + 2x_2 \le 2$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

Use graphical analysis to determine the optimal solution(s) for (x_1, x_2) for the various possible values of c_1 and c_2. (*Hint:* Separate the cases where $c_2 = 0$, $c_2 > 0$, and $c_2 < 0$. For the latter two cases, focus on the ratio of c_1 to c_2.)

3.2-1. The following table summarizes the key facts about two products, A and B, and the resources, Q, R, and S, required to produce them.

	Resource Usage per Unit Produced		
Resource	**Product A**	**Product B**	**Amount of Resource Available**
Q	2	1	2
R	1	2	2
S	3	3	4
Profit per unit	3	2	

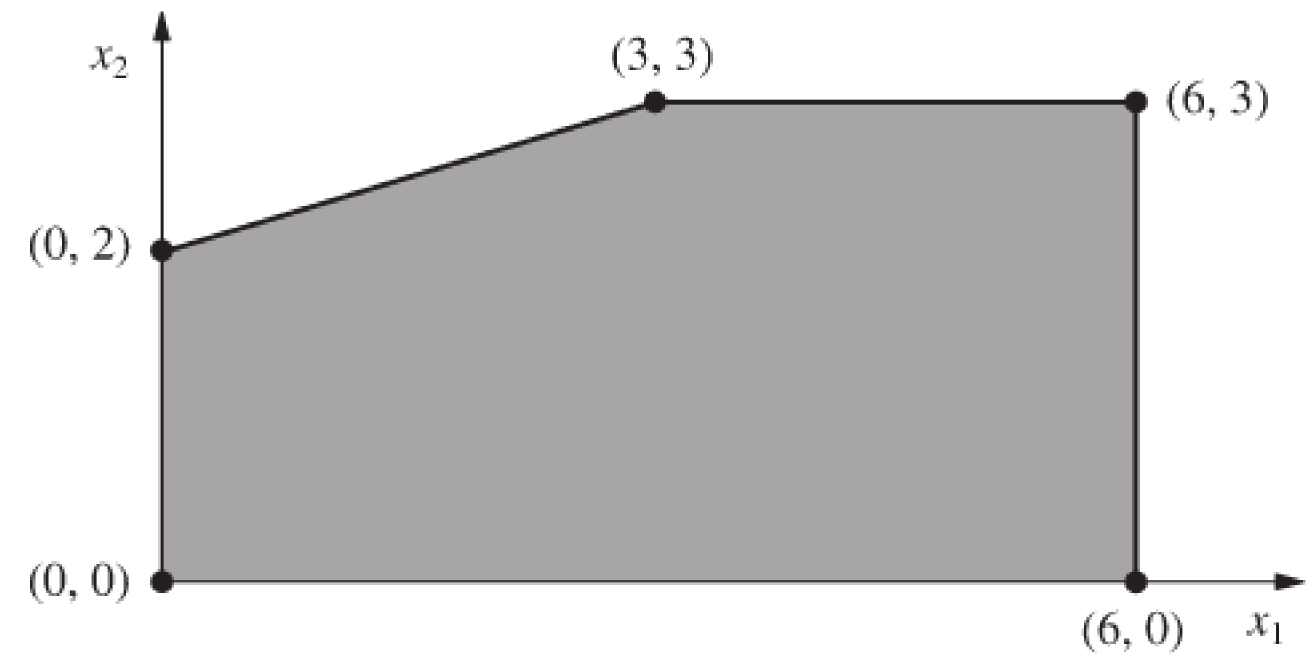
All the assumptions of linear programming hold.

(a) Formulate a linear programming model for this problem.

D,I **(b)** Solve this model graphically.

(c) Verify the exact value of your optimal solution from part (*b*) by solving algebraically for the simultaneous solution of the relevant two equations.

3.2-2. The shaded area in the following graph represents the feasible region of a linear programming problem whose objective function is to be maximized.



Label each of the following statements as True or False, and then justify your answer based on the graphical method. In each case, give an example of an objective function that illustrates your answer.

(a) If (3, 3) produces a larger value of the objective function than (0, 2) and (6, 3), then (3, 3) must be an optimal solution.

(b) If (3, 3) is an optimal solution and multiple optimal solutions exist, then either (0, 2) or (6, 3) must also be an optimal solution.

(c) The point (0, 0) cannot be an optimal solution.

3.2-3.* This is your lucky day. You have just won a \$10,000 prize. You are setting aside \$4,000 for taxes and partying expenses, but you have decided to invest the other \$6,000. Upon hearing this news, two different friends have offered you an opportunity to become a partner in two different entrepreneurial ventures, one planned by each friend. In both cases, this investment would involve expending some of your time next summer as well as putting up cash. Becoming a *full* partner in the first friend's venture would require an investment of \$5,000 and 400 hours, and your estimated profit (ignoring the value of your time) would be \$4,500. The corresponding figures for the second friend's venture are \$4,000 and 500 hours, with an estimated profit to you of \$4,500. However, both friends are flexible and would allow you to come in at any *fraction* of a full partnership you would like. If you choose a fraction of a full partnership, all the above figures given for a full partnership (money investment, time investment, and your profit) would be multiplied by this same fraction.

Because you were looking for an interesting summer job anyway (maximum of 600 hours), you have decided to participate in one or both friends' ventures in whichever combination would maximize your total estimated profit. You now need to solve the problem of finding the best combination.

(a) Describe the analogy between this problem and the Wyndor Glass Co. problem discussed in Sec. 3.1. Then construct and fill in a table like Table 3.1 for this problem, identifying both the activities and the resources.

(b) Formulate a linear programming model for this problem.

D,I **(c)** Use the graphical method to solve this model. What is your total estimated profit?

D,I **3.2-4.** Use the graphical method to find all optimal solutions for the following model:

Maximize $Z = 500x_1 + 300x_2$,

subject to

$$15x_1 + 5x_2 \le 300$$
$$10x_1 + 6x_2 \le 240$$
$$8x_1 + 12x_2 \le 450$$

and

$x_1 \ge 0, \quad x_2 \ge 0.$

D **3.2-5.** Use the graphical method to demonstrate that the following model has no feasible solutions.

Maximize $Z = 5x_1 + 7x_2$,

subject to

$$2x_1 - x_2 \le -1$$
$$-x_1 + 2x_2 \le -1$$

and

$x_1 \ge 0, \quad x_2 \ge 0.$

D **3.2-6.** Suppose that the following constraints have been provided for a linear programming model.

$$-x_1 + 2x_2 \le 50$$
$$-2x_1 + x_2 \le 50$$

and

$x_1 \ge 0, \quad x_2 \ge 0.$

(a) Demonstrate that the feasible region is unbounded.
(b) If the objective is to maximize $Z = -x_1 + x_2$, does the model have an optimal solution? If so, find it. If not, explain why not.
(c) Repeat part (*b*) when the objective is to maximize $Z = x_1 - x_2$.
(d) For objective functions where this model has no optimal solution, does this mean that there are no good solutions according to the model? Explain. What probably went wrong when formulating the model?

3.3-1. Reconsider Prob. 3.2-3. Indicate why each of the four assumptions of linear programming (Sec. 3.3) appears to be reasonably satisfied for this problem. Is one assumption more doubtful than the others? If so, what should be done to take this into account?

3.3-2. Consider a problem with two decision variables, x_1 and x_2, which represent the levels of activities 1 and 2, respectively. For each variable, the permissible values are 0, 1, and 2, where the feasible combinations of these values for the two variables are determined from a variety of constraints. The objective is to maximize a certain measure of performance denoted by Z. The values of Z for the possibly feasible values of (x_1, x_2) are estimated to be those given in the following table:

	x_2		
x_1	**0**	**1**	**2**
0	0	4	8
1	3	8	13
2	6	12	18

Based on this information, indicate whether this problem completely satisfies each of the four assumptions of linear programming. Justify your answers.

3.4-1. Read the referenced article that fully describes the OR study summarized in the first application vignette presented in Sec. 3.4. Briefly describe how linear programming was applied in this study. Then list the various financial and nonfinancial benefits that resulted from this study.

3.4-2. Read the referenced article that fully describes the OR study summarized in the second application vignette presented in Sec. 3.4. Briefly describe how linear programming was applied in this study. Then list the various financial and nonfinancial benefits that resulted from this study.

3.4-3.* For each of the four assumptions of linear programming discussed in Sec. 3.3, write a one-paragraph analysis of how well you feel it applies to each of the following examples given in Sec. 3.4:
(a) Design of radiation therapy (Mary).
(b) Regional planning (Southern Confederation of Kibbutzim).
(c) Controlling air pollution (Nori & Leets Co.).

3.4-4. For each of the four assumptions of linear programming discussed in Sec. 3.3, write a one-paragraph analysis of how well it applies to each of the following examples given in Sec. 3.4.
(a) Reclaiming solid wastes (Save-It Co.).
(b) Personnel scheduling (Union Airways).
(c) Distributing goods through a distribution network (Distribution Unlimited Co.).

D,I **3.4-5.** Use the graphical method to solve this problem:

Minimize $Z = 15x_1 + 20x_2$,

subject to

$$x_1 + 2x_2 \ge 10$$
$$2x_1 - 3x_2 \le 6$$
$$x_1 + x_2 \ge 6$$

and

$x_1 \ge 0, \quad x_2 \ge 0.$

D,I **3.4-6.** Use the graphical method to solve this problem:

Minimize $Z = 3x_1 + 2x_2$,

subject to

$$x_1 + 2x_2 \le 12$$

$$2x_1 + 3x_2 = 12$$
$$2x_1 + x_2 \geq 8$$

and

$$x_1 \geq 0, \quad x_2 \geq 0.$$

D **3.4-7.** Consider the following problem, where the value of c_1 has not yet been ascertained.

$$\text{Maximize} \quad Z = c_1x_1 + 2x_2,$$

subject to

$$4x_1 + x_2 \leq 12$$
$$x_1 - x_2 \geq 2$$

and

$$x_1 \geq 0, \quad x_2 \geq 0.$$

Use graphical analysis to determine the optimal solution(s) for (x_1, x_2) for the various possible values of c_1.

D,I **3.4-8.** Consider the following model:

$$\text{Minimize} \quad Z = 40x_1 + 50x_2,$$

subject to

$$2x_1 + 3x_2 \geq 30$$
$$x_1 + x_2 \geq 12$$
$$2x_1 + x_2 \geq 20$$

and

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Use the graphical method to solve this model.
(b) How does the optimal solution change if the objective function is changed to $Z = 40x_1 + 70x_2$? (You may find it helpful to use the Graphical Analysis and Sensitivity Analysis procedure in IOR Tutorial.)
(c) How does the optimal solution change if the third functional constraint is changed to $2x_1 + x_2 \geq 15$? (You may find it helpful to use the Graphical Analysis and Sensitivity Analysis procedure in IOR Tutorial.)

3.4-9. Ralph Edmund loves steaks and potatoes. Therefore, he has decided to go on a steady diet of only these two foods (plus some liquids and vitamin supplements) for all his meals. Ralph realizes that this isn't the healthiest diet, so he wants to make sure that he eats the right quantities of the two foods to satisfy some key nutritional requirements. He has obtained the nutritional and cost information shown at the top of the next column.

Ralph wishes to determine the number of daily servings (may be fractional) of steak and potatoes that will meet these requirements at a minimum cost.
(a) Formulate a linear programming model for this problem.
D,I **(b)** Use the graphical method to solve this model.
C **(c)** Use a computer to solve this model by the simplex method.

	Grams of Ingredient per Serving		
Ingredient	**Steak**	**Potatoes**	**Daily Requirement (Grams)**
Carbohydrates	5	15	≥ 50
Protein	20	5	≥ 40
Fat	15	2	≤ 60
Cost per serving	\$4	\$2	

3.4-10. Web Mercantile sells many household products through an online catalog. The company needs substantial warehouse space for storing its goods. Plans now are being made for leasing warehouse storage space over the next 5 months. Just how much space will be required in each of these months is known. However, since these space requirements are quite different, it may be most economical to lease only the amount needed each month on a month-by-month basis. On the other hand, the additional cost for leasing space for additional months is much less than for the first month, so it may be less expensive to lease the maximum amount needed for the entire 5 months. Another option is the intermediate approach of changing the total amount of space leased (by adding a new lease and/or having an old lease expire) at least once but not every month.

The space requirement and the leasing costs for the various leasing periods are as follows:

Month	Required Space (Sq. Ft.)
1	30,000
2	20,000
3	40,000
4	10,000
5	50,000

Leasing Period (Months)	Cost per Sq. Ft. Leased
1	\$ 65
2	\$100
3	\$135
4	\$160
5	\$190

The objective is to minimize the total leasing cost for meeting the space requirements.
(a) Formulate a linear programming model for this problem.
C **(b)** Solve this model by the simplex method.

3.4-11. Larry Edison is the director of the Computer Center for Buckly College. He now needs to schedule the staffing of the center. It is open from 8 A.M. until midnight. Larry has monitored the usage of the center at various times of the day, and determined that the following number of computer consultants are required:

Time of Day	Minimum Number of Consultants Required to Be on Duty
8 A.M.–noon	4
Noon–4 P.M.	8
4 P.M.–8 P.M.	10
8 P.M.–midnight	6

Two types of computer consultants can be hired: full-time and part-time. The full-time consultants work for 8 consecutive hours in any of the following shifts: morning (8 A.M.–4 P.M.), afternoon (noon–8 P.M.), and evening (4 P.M.–midnight). Full-time consultants are paid \$40 per hour.

Part-time consultants can be hired to work any of the four shifts listed in the above table. Part-time consultants are paid \$30 per hour.

An additional requirement is that during every time period, there must be at least 2 full-time consultants on duty for every part-time consultant on duty.

Larry would like to determine how many full-time and how many part-time workers should work each shift to meet the above requirements at the minimum possible cost.

(a) Formulate a linear programming model for this problem.

C **(b)** Solve this model by the simplex method.

3.4-12.* The Medequip Company produces precision medical diagnostic equipment at two factories. Three medical centers have placed orders for this month's production output. The table below shows what the cost would be for shipping each unit from each factory to each of these customers. Also shown are the number of units that will be produced at each factory and the number of units ordered by each customer.

To / From	Unit Shipping Cost			
	Customer 1	Customer 2	Customer 3	Output
Factory 1	\$600	\$800	\$700	400 units
Factory 2	\$400	\$900	\$600	500 units
Order size	300 units	200 units	400 units	

A decision now needs to be made about the shipping plan for how many units to ship from each factory to each customer.

(a) Formulate a linear programming model for this problem.

C **(b)** Solve this model by the simplex method.

3.4-13.* Al Ferris has \$60,000 that he wishes to invest now in order to use the accumulation for purchasing a retirement annuity in 5 years. After consulting with his financial adviser, he has been offered four types of fixed-income investments, which we will label as investments A, B, C, D.

Investments A and B are available at the beginning of each of the next 5 years (call them years 1 to 5). Each dollar invested in A at the beginning of a year returns \$1.40 (a profit of \$0.40) 2 years later (in time for immediate reinvestment). Each dollar invested in B at the beginning of a year returns \$1.70 three years later.

Investments C and D will each be available at one time in the future. Each dollar invested in C at the beginning of year 2 returns \$1.90 at the end of year 5. Each dollar invested in D at the beginning of year 5 returns \$1.30 at the end of year 5.

Al wishes to know which investment plan maximizes the amount of money that can be accumulated by the beginning of year 6.

(a) All the functional constraints for this problem can be expressed as equality constraints. To do this, let A_t, B_t, C_t, and D_t be the amount invested in investment A, B, C, and D, respectively, at the beginning of year t for each t where the investment is available and will mature by the end of year 5. Also let R_t be the number of available dollars *not* invested at the beginning of year t (and so available for investment in a later year). Thus, the amount invested at the beginning of year t *plus* R_t must equal the number of dollars available for investment at that time. Write such an equation in terms of the relevant variables above for the beginning of each of the 5 years to obtain the five functional constraints for this problem.

(b) Formulate a complete linear programming model for this problem.

C **(c)** Solve this model by the simplex model.

3.4-14. The Metalco Company desires to blend a new alloy of 40 percent tin, 35 percent zinc, and 25 percent lead from several available alloys having the following properties:

	Alloy				
Property	1	2	3	4	5
Percentage of tin	60	25	45	20	50
Percentage of zinc	10	15	45	50	40
Percentage of lead	30	60	10	30	10
Cost (\$/lb)	77	70	88	84	94

The objective is to determine the proportions of these alloys that should be blended to produce the new alloy at a minimum cost.

(a) Formulate a linear programming model for this problem.

C **(b)** Solve this model by the simplex method.

3.4-15* A cargo plane has three compartments for storing cargo: front, center, and back. These compartments have capacity limits on both *weight* and *space,* as summarized below:

Compartment	Weight Capacity (Tons)	Space Capacity (Cubic Feet)
Front	12	7,000
Center	18	9,000
Back	10	5,000

Furthermore, the weight of the cargo in the respective compartments must be the same proportion of that compartment's weight capacity to maintain the balance of the airplane.

The following four cargoes have been offered for shipment on an upcoming flight as space is available:

Cargo	Weight (Tons)	Volume (Cubic Feet/Ton)	Profit ($/Ton)
1	20	500	320
2	16	700	400
3	25	600	360
4	13	400	290

Any portion of these cargoes can be accepted. The objective is to determine how much (if any) of each cargo should be accepted and how to distribute each among the compartments to maximize the total profit for the flight.

(a) Formulate a linear programming model for this problem.

C **(b)** Solve this model by the simplex method to find one of its multiple optimal solutions.

3.4-16. Oxbridge University maintains a powerful mainframe computer for research use by its faculty, Ph.D. students, and research associates. During all working hours, an operator must be available to operate and maintain the computer, as well as to perform some programming services. Beryl Ingram, the director of the computer facility, oversees the operation.

It is now the beginning of the fall semester, and Beryl is confronted with the problem of assigning different working hours to her operators. Because all the operators are currently enrolled in the university, they are available to work only a limited number of hours each day, as shown in the following table.

		Maximum Hours of Availability				
Operators	Wage Rate	Mon.	Tue.	Wed.	Thurs.	Fri.
K. C.	$25/hour	6	0	6	0	6
D. H.	$26/hour	0	6	0	6	0
H. B.	$24/hour	4	8	4	0	4
S. C.	$23/hour	5	5	5	0	5
K. S.	$28/hour	3	0	3	8	0
N. K.	$30/hour	0	0	0	6	2

There are six operators (four undergraduate students and two graduate students). They all have different wage rates because of differences in their experience with computers and in their programming ability. The above table shows their wage rates, along with the maximum number of hours that each can work each day.

Each operator is guaranteed a certain minimum number of hours per week that will maintain an adequate knowledge of the operation. This level is set arbitrarily at 8 hours per week for the undergraduate students (K. C., D. H., H. B., and S. C.) and 7 hours per week for the graduate students (K. S. and N. K.).

The computer facility is to be open for operation from 8 A.M. to 10 P.M. Monday through Friday with exactly one operator on duty during these hours. On Saturdays and Sundays, the computer is to be operated by other staff.

Because of a tight budget, Beryl has to minimize cost. She wishes to determine the number of hours she should assign to each operator on each day.

(a) Formulate a linear programming model for this problem.

C **(b)** Solve this model by the simplex method.

3.4-17. Joyce and Marvin run a day care for preschoolers. They are trying to decide what to feed the children for lunches. They would like to keep their costs down, but also need to meet the nutritional requirements of the children. They have already decided to go with peanut butter and jelly sandwiches, and some combination of graham crackers, milk, and orange juice. The nutritional content of each food choice and its cost are given in the table below.

Food Item	Calories from Fat	Total Calories	Vitamin C (mg)	Protein (g)	Cost (¢)
Bread (1 slice)	10	70	0	3	5
Peanut butter (1 tbsp)	75	100	0	4	4
Strawberry jelly (1 tbsp)	0	50	3	0	7
Graham cracker (1 cracker)	20	60	0	1	8
Milk (1 cup)	70	150	2	8	15
Juice (1 cup)	0	100	120	1	35

The nutritional requirements are as follows. Each child should receive between 400 and 600 calories. No more than 30 percent of the total calories should come from fat. Each child should consume at least 60 milligrams (mg) of vitamin C and 12 grams (g) of protein. Furthermore, for practical reasons, each child needs exactly 2 slices of bread (to make the sandwich), at least twice as much peanut butter as jelly, and at least 1 cup of liquid (milk and/or juice).

Joyce and Marvin would like to select the food choices for each child which minimize cost while meeting the above requirements.

(a) Formulate a linear programming model for this problem.

C **(b)** Solve this model by the simplex method.

3.5-1. Read the referenced article that fully describes the OR study summarized in the application vignette presented in Sec. 3.5. Briefly describe how linear programming was applied in this study. Then list the various financial and nonfinancial benefits that resulted from this study

3.5-2.* You are given the following data for a linear programming problem where the objective is to maximize the profit from allocating three resources to two nonnegative activities.

Resource	Resource Usage per Unit of Each Activity		Amount of Resource Available
	Activity 1	Activity 2	
1	2	1	10
2	3	3	20
3	2	4	20
Contribution per unit	$20	$30	

Contribution per unit = profit per unit of the activity.

(a) Formulate a linear programming model for this problem.
D,I **(b)** Use the graphical method to solve this model.
(c) Display the model on an Excel spreadsheet.
(d) Use the spreadsheet to check the following solutions: $(x_1, x_2) = (2, 2), (3, 3), (2, 4), (4, 2), (3, 4), (4, 3)$. Which of these solutions are feasible? Which of these feasible solutions has the best value of the objective function?
C **(e)** Use the Excel Solver to solve the model by the simplex method.

3.5-3. Ed Butler is the production manager for the Bilco Corporation, which produces three types of spare parts for automobiles. The manufacture of each part requires processing on each of two machines, with the following processing times (in hours):

Machine	Part		
	A	B	C
1	0.02	0.03	0.05
2	0.05	0.02	0.04

Each machine is available 40 hours per month. Each part manufactured will yield a unit profit as follows:

	Part		
	A	B	C
Profit	$300	$250	$200

Ed wants to determine the mix of spare parts to produce in order to maximize total profit.
(a) Formulate a linear programming model for this problem.
(b) Display the model on an Excel spreadsheet.
(c) Make three guesses of your own choosing for the optimal solution. Use the spreadsheet to check each one for feasibility and, if feasible, to find the value of the objective function. Which feasible guess has the best objective function value?
(d) Use the Excel Solver to solve the model by the simplex method.

3.5-4. You are given the following data for a linear programming problem where the objective is to minimize the cost of conducting two nonnegative activities so as to achieve three benefits that do not fall below their minimum levels.

Benefit	Benefit Contribution per Unit of Each Activity		Minimum Acceptable Level
	Activity 1	Activity 2	
1	5	3	60
2	2	2	30
3	7	9	126
Unit cost	$60	$50	

(a) Formulate a linear programming model for this problem.
D,I **(b)** Use the graphical method to solve this model.
(c) Display the model on an Excel spreadsheet.
(d) Use the spreadsheet to check the following solutions: $(x_1, x_2) = (7, 7), (7, 8), (8, 7), (8, 8), (8, 9), (9, 8)$. Which of these solutions are feasible? Which of these feasible solutions has the best value of the objective function?
C **(e)** Use the Excel Solver to solve this model by the simplex method.

3.5-5.* Fred Jonasson manages a family-owned farm. To supplement several food products grown on the farm, Fred also raises pigs for market. He now wishes to determine the quantities of the available types of feed (corn, tankage, and alfalfa) that should be given to each pig. Since pigs will eat any mix of these feed types, the objective is to determine which mix will meet certain nutritional requirements at a *minimum cost*. The number of units of each type of basic nutritional ingredient contained within a kilogram of each feed type is given in the following table, along with the daily nutritional requirements and feed costs:

Nutritional Ingredient	Kilogram of Corn	Kilogram of Tankage	Kilogram of Alfalfa	Minimum Daily Requirement
Carbohydrates	90	20	40	200
Protein	30	80	60	180
Vitamins	10	20	60	150
Cost (¢)	84	72	60	

(a) Formulate a linear programming model for this problem.
(b) Display the model on an Excel spreadsheet.
(c) Use the spreadsheet to check if $(x_1, x_2, x_3) = (1, 2, 2)$ is a feasible solution and, if so, what the daily cost would be for this

diet. How many units of each nutritional ingredient would this diet provide daily?

(d) Take a few minutes to use a trial-and-error approach with the spreadsheet to develop your best guess for the optimal solution. What is the daily cost for your solution?

C **(e)** Use the Excel Solver to solve the model by the simplex method.

3.5-6. Maureen Laird is the chief financial officer for the Alva Electric Co., a major public utility in the midwest. The company has scheduled the construction of new hydroelectric plants 5, 10, and 20 years from now to meet the needs of the growing population in the region served by the company. To cover at least the construction costs, Maureen needs to invest some of the company's money now to meet these future cash-flow needs. Maureen may purchase only three kinds of financial assets, each of which costs \$1 million per unit. Fractional units may be purchased. The assets produce income 5, 10, and 20 years from now, and that income is needed to cover at least minimum cash-flow requirements in those years. (Any excess income above the minimum requirement for each time period will be used to increase dividend payments to shareholders rather than saving it to help meet the minimum cash-flow requirement in the next time period.) The following table shows both the amount of income generated by each unit of each asset and the minimum amount of income needed for each of the future time periods when a new hydroelectric plant will be constructed.

Year	Income per Unit of Asset			Minimum Cash Flow Required
	Asset 1	Asset 2	Asset 3	
5	\$2 million	\$1 million	\$0.5 million	\$400 million
10	\$0.5 million	\$0.5 million	\$1 million	\$100 million
20	0	\$1.5 million	\$2 million	\$300 million

Maureen wishes to determine the mix of investments in these assets that will cover the cash-flow requirements while minimizing the total amount invested.

(a) Formulate a linear programming model for this problem.

(b) Display the model on a spreadsheet.

(c) Use the spreadsheet to check the possibility of purchasing 100 units of Asset 1, 100 units of Asset 2, and 200 units of Asset 3. How much cash flow would this mix of investments generate 5, 10, and 20 years from now? What would be the total amount invested?

(d) Take a few minutes to use a trial-and-error approach with the spreadsheet to develop your best guess for the optimal solution. What is the total amount invested for your solution?

C **(e)** Use the Excel Solver to solve the model by the simplex method.

3.6-1. The Philbrick Company has two plants on opposite sides of the United States. Each of these plants produces the same two products and then sells them to wholesalers within its half of the country. The orders from wholesalers have already been received for the next 2 months (February and March), where the number of units requested are shown below. (The company is not obligated to completely fill these orders but will do so if it can without decreasing its profits.)

Product	Plant 1		Plant 2	
	February	March	February	March
1	3,600	6,300	4,900	4,200
2	4,500	5,400	5,100	6,000

Each plant has 20 production days available in February and 23 production days available in March to produce and ship these products. Inventories are depleted at the end of January, but each plant has enough inventory capacity to hold 1,000 units total of the two products if an excess amount is produced in February for sale in March. In either plant, the cost of holding inventory in this way is \$3 per unit of product 1 and \$4 per unit of product 2.

Each plant has the same two production processes, each of which can be used to produce either of the two products. The production cost per unit produced of each product is shown below for each process in each plant.

Product	Plant 1		Plant 2	
	Process 1	Process 2	Process 1	Process 2
1	\$62	\$59	\$61	\$65
2	\$78	\$85	\$89	\$86

The production rate for each product (number of units produced per day devoted to that product) also is given for each process in each plant below.

Product	Plant 1		Plant 2	
	Process 1	Process 2	Process 1	Process 2
1	100	140	130	110
2	120	150	160	130

The net sales revenue (selling price minus normal shipping costs) the company receives when a plant sells the products to its own customers (the wholesalers in its half of the country) is \$83 per unit of product 1 and \$112 per unit of product 2. However, it also is possible (and occasionally desirable) for a plant to make a shipment to the other half of the country to help fill the sales of the other plant. When this happens, an extra shipping cost of \$9 per unit of product 1 and \$7 per unit of product 2 is incurred.

Management now needs to determine how much of each product should be produced by each production process in each plant during each month, as well as how much each plant should sell of each product in each month and how much each plant should ship of each product in each month to the other plant's customers. The objective is to determine which feasible plan would maximize the total profit (total net sales revenue minus the sum of the production costs, inventory costs, and extra shipping costs).

(a) Formulate a complete linear programming model in algebraic form that shows the individual constraints and decision variables for this problem.

C **(b)** Formulate this same model on an Excel spreadsheet instead. Then use the Excel Solver to solve the model.

C **(c)** Use MPL to formulate this model in a compact form. Then use the MPL solver CPLEX to solve the model.

C **(d)** Use LINGO to formulate this model in a compact form. Then use the LINGO solver to solve the model.

C **3.6-2.** Reconsider Prob. 3.1-11.

(a) Use MPL/CPLEX to formulate and solve the model for this problem.

(b) Use LINGO to formulate and solve this model.

C **3.6-3.** Reconsider Prob. 3.4-12.

(a) Use MPL/CPLEX to formulate and solve the model for this problem.

(b) Use LINGO to formulate and solve this model.

C **3.6-4.** Reconsider Prob. 3.4-16.

(a) Use MPL/CPLEX to formulate and solve the model for this problem.

(b) Use LINGO to formulate and solve this model.

C **3.6-5.** Reconsider Prob. 3.5-5.

(a) Use MPL/CPLEX to formulate and solve the model for this problem.

(b) Use LINGO to formulate and solve this model.

C **3.6-6.** Reconsider Prob. 3.5-6.

(a) Use MPL/CPLEX to formulate and solve the model for this problem.

(b) Use LINGO to formulate and solve this model.

3.6-7. A large paper manufacturing company, the Quality Paper Corporation, has 10 paper mills from which it needs to supply 1,000 customers. It uses three alternative types of machines and four types of raw materials to make five different types of paper. Therefore, the company needs to develop a detailed production distribution plan on a monthly basis, with an objective of minimizing the total cost of producing and distributing the paper during the month. Specifically, it is necessary to determine jointly the amount of each type of paper to be made at each paper mill on each type of machine *and* the amount of each type of paper to be shipped from each paper mill to each customer.

The relevant data can be expressed symbolically as follows:

D_{jk} = number of units of paper type k demanded by customer j,

r_{klm} = number of units of raw material m needed to produce 1 unit of paper type k on machine type l,

R_{im} = number of units of raw material m available at paper mill i,

c_{kl} = number of capacity units of machine type l that will produce 1 unit of paper type k,

C_{il} = number of capacity units of machine type l available at paper mill i,

P_{ikl} = production cost for each unit of paper type k produced on machine type l at paper mill i,

T_{ijk} = transportation cost for each unit of paper type k shipped from paper mill i to customer j.

(a) Using these symbols, formulate a linear programming model for this problem by hand.

(b) How many functional constraints and decision variables does this model have?

C **(c)** Use MPL to formulate this problem.

C **(d)** Use LINGO to formulate this problem.

3.7-1. From the bottom part of the selected references given at the end of the chapter, select one of these award-winning applications of linear programming. Read this article and then write a two-page summary of the application and the benefits (including nonfinancial benefits) it provided.

3.7-2. From the bottom part of the selected references given at the end of the chapter, select three of these award-winning applications of linear programming. For each one, read the article and then write a one-page summary of the application and the benefits (including nonfinancial benefits) it provided.

CASES

CASE 3.1 Auto Assembly

Automobile Alliance, a large automobile manufacturing company, organizes the vehicles it manufactures into three families: a family of trucks, a family of small cars, and a family of midsized and luxury cars. One plant outside Detroit, MI, assembles two models from the family of midsized and luxury cars. The first model, the Family Thrillseeker, is a four-door sedan with vinyl seats, plastic interior, standard features, and excellent gas mileage. It is marketed as a smart buy for middle-class families with tight budgets, and each Family Thrillseeker sold generates a modest profit of $3,600 for the company. The second model, the Classy Cruiser, is a two-door luxury sedan with leather seats, wooden interior, custom features, and navigational capabilities. It is marketed as a privilege of affluence for upper-middle-class families, and each Classy Cruiser sold generates a healthy profit of $5,400 for the company.

Rachel Rosencrantz, the manager of the assembly plant, is currently deciding the production schedule for the next month. Specifically, she must decide how many Family Thrillseekers and how many Classy Cruisers to assemble in the plant to maximize profit for the company. She knows that the plant possesses a capacity of 48,000 labor-hours during the month. She also knows that it takes 6 labor-hours to assemble one Family Thrillseeker and 10.5 labor-hours to assemble one Classy Cruiser.

Because the plant is simply an assembly plant, the parts required to assemble the two models are not produced at the plant. They are instead shipped from other plants around the Michigan area to the assembly plant. For example, tires, steering wheels, windows, seats, and doors all arrive from various supplier plants. For the next month, Rachel knows that she will be able to obtain only 20,000 doors (10,000 left-hand doors and 10,000 right-hand doors) from the door supplier. A recent labor strike forced the shutdown of that particular supplier plant for several days, and that plant will not be able to meet its production schedule for the next month. Both the Family Thrillseeker and the Classy Cruiser use the same door part.

In addition, a recent company forecast of the monthly demands for different automobile models suggests that the demand for the Classy Cruiser is limited to 3,500 cars. There is no limit on the demand for the Family Thrillseeker within the capacity limits of the assembly plant.

(a) Formulate and solve a linear programming problem to determine the number of Family Thrillseekers and the number of Classy Cruisers that should be assembled.

Before she makes her final production decisions, Rachel plans to explore the following questions independently except where otherwise indicated.

(b) The marketing department knows that it can pursue a targeted $500,000 advertising campaign that will raise the demand for the Classy Cruiser next month by 20 percent. Should the campaign be undertaken?

(c) Rachel knows that she can increase next month's plant capacity by using overtime labor. She can increase the plant's labor-hour capacity by 25 percent. With the new assembly plant capacity, how many Family Thrillseekers and how many Classy Cruisers should be assembled?

(d) Rachel knows that overtime labor does not come without an extra cost. What is the maximum amount she should be willing to pay for all overtime labor beyond the cost of this labor at regular time rates? Express your answer as a lump sum.

(e) Rachel explores the option of using both the targeted advertising campaign and the overtime labor-hours. The advertising campaign raises the demand for the Classy Cruiser by 20 percent, and the overtime labor increases the plant's labor-hour capacity by 25 percent. How many Family Thrillseekers and how many Classy Cruisers should be assembled using the advertising campaign and overtime labor-hours if the profit from each Classy Cruiser sold continues to be 50 percent more than for each Family Thrillseeker sold?

(f) Knowing that the advertising campaign costs $500,000 and the maximum usage of overtime labor-hours costs $1,600,000 beyond regular time rates, is the solution found in part (*e*) a wise decision compared to the solution found in part (*a*)?

(g) Automobile Alliance has determined that dealerships are actually heavily discounting the price of the Family Thrillseekers to move them off the lot. Because of a profit-sharing agreement with its dealers, the company is therefore not making a profit of $3,600 on the Family Thrillseeker but is instead making a profit of $2,800. Determine the number of Family Thrillseekers and the number of Classy Cruisers that should be assembled given this new discounted price.

(h) The company has discovered quality problems with the Family Thrillseeker by randomly testing Thrillseekers at the end of the assembly line. Inspectors have discovered that in over 60 percent of the cases, two of the four doors on a Thrillseeker do not seal properly. Because the percentage of defective Thrillseekers determined by the random testing is so high, the floor supervisor has decided to perform quality control tests on every Thrillseeker at the end of the line. Because of the added tests, the time it takes to assemble one Family Thrillseeker has increased from 6 to 7.5 hours. Determine the number of units of each model that should be assembled given the new assembly time for the Family Thrillseeker.

(i) The board of directors of Automobile Alliance wishes to capture a larger share of the luxury sedan market and therefore would like to meet the full demand for Classy Cruisers. They ask Rachel to determine by how much the profit of her assembly plant would decrease as compared to the profit found in part (*a*). They then ask her to meet the full demand for Classy Cruisers if the decrease in profit is not more than $2,000,000.

(j) Rachel now makes her final decision by combining all the new considerations described in parts (*f*), (*g*), and (*h*). What are her final decisions on whether to undertake the advertising campaign, whether to use overtime labor, the number of Family Thrillseekers to assemble, and the number of Classy Cruisers to assemble?

PREVIEWS OF ADDED CASES ON OUR WEBSITE (www.mhhe.com/hillier)

CASE 3.2 Cutting Cafeteria Costs

This case focuses on a subject that is dear to the heart of many students. How should the manager of a college cafeteria choose the ingredients of a casserole dish to make it sufficiently tasty for the students while also minimizing costs? In this case, linear programming models with only two decision variables can be used to address seven specific issues being faced by the manager.

CASE 3.3 Staffing a Call Center

California Children's Hospital currently uses a confusing, decentralized appointment and registration process for its patients. Therefore, the decision has been made to centralize the process by establishing one call center devoted exclusively to appointments and registration. The hospital manager now needs to develop a plan for how many employees of each kind (full-time or part-time, English speaking, Spanish speaking, or bilingual) should be hired for each of several possible work shifts. Linear programming is needed to find a plan that minimizes the total cost of providing a satisfactory level of service throughout the 14 hours that the call center will be open each weekday. The model requires more than two decision variables, so a software package such as described in Sec. 3.5 or Sec. 3.6 will be needed to solve the two versions of the model.

CASE 3.4 Promoting a Breakfast Cereal

The vice president for marketing of the Super Grain Corporation needs to develop a promotional campaign for the company's new breakfast cereal. Three advertising media have been chosen for the campaign, but decisions now need to be made regarding how much of each medium should be used. Constraints include a limited advertising budget, a limited planning budget, and a limited number of TV commercial spots available, as well as requirements for effectively reaching two special target audiences (young children and parents of young children) and for making full use of a rebate program. The corresponding linear programming model requires more than two decision variables, so a software package such as described in Sec. 3.5 or Sec. 3.6 will be needed to solve the model. This case also asks for an analysis of how well the four assumptions of linear programming are satisfied for this problem. Does linear programming actually provide a reasonable basis for managerial decision making in this situation? (Case 12.3 will provide a continuation of this case.)

CHAPTER 4

Solving Linear Programming Problems: The Simplex Method

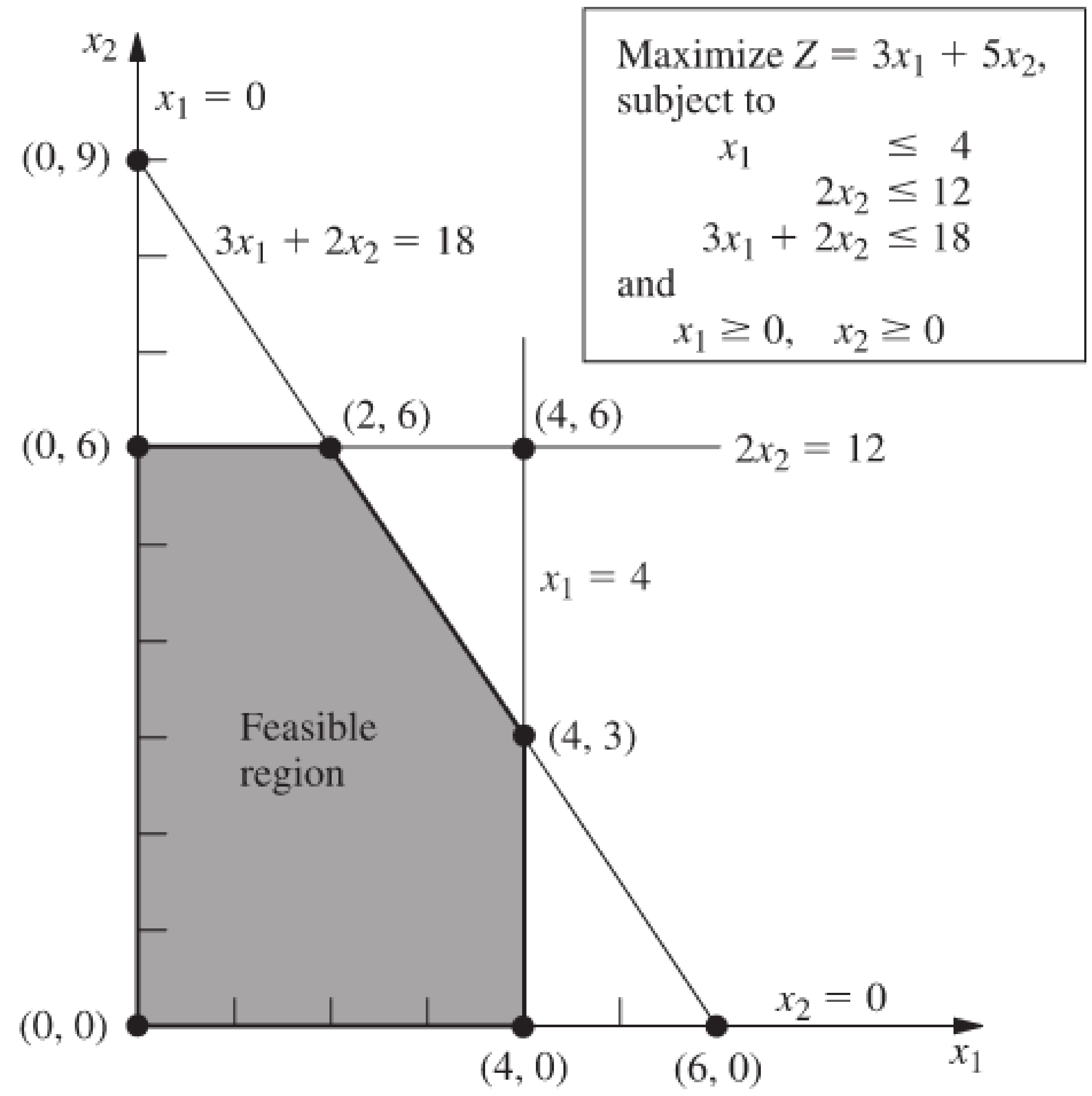
We now are ready to begin studying the *simplex method,* a general procedure for solving linear programming problems. Developed by the brilliant George Dantzig[1] in 1947, it has proved to be a remarkably efficient method that is used routinely to solve huge problems on today's computers. Except for its use on tiny problems, this method is always executed on a computer, and sophisticated software packages are widely available. Extensions and variations of the simplex method also are used to perform *postoptimality analysis* (including sensitivity analysis) on the model.

This chapter describes and illustrates the main features of the simplex method. The first section introduces its general nature, including its geometric interpretation. The following three sections then develop the procedure for solving any linear programming model that is in our standard form (maximization, all functional constraints in $\leq$ form, and nonnegativity constraints on all variables) and has only *nonnegative* right-hand sides b_i in the functional constraints. Certain details on resolving ties are deferred to Sec. 4.5, and Sec. 4.6 describes how to adapt the simplex method to other model forms. Next we discuss postoptimality analysis (Sec. 4.7), and describe the computer implementation of the simplex method (Sec. 4.8). Section 4.9 then introduces an alternative to the simplex method (the interior-point approach) for solving large linear programming problems.

4.1 THE ESSENCE OF THE SIMPLEX METHOD

The simplex method is an *algebraic* procedure. However, its underlying concepts are *geometric*. Understanding these geometric concepts provides a strong intuitive feeling for how the simplex method operates and what makes it so efficient. Therefore, before delving into algebraic details, we focus in this section on the big picture from a geometric viewpoint.

[1]Widely revered as perhaps the most important pioneer of operations research, George Dantzig is commonly referred to as the *father of linear programming* because of the development of the simplex method and many key subsequent contributions. The authors had the privilege of being his faculty colleagues in the Department of Operations Research at Stanford University for nearly 30 years. Dr. Dantzig remained professionally active right up until he passed away in 2005 at the age of 90.



■ **FIGURE 4.1**
Constraint boundaries and corner-point solutions for the Wyndor Glass Co. problem.

To illustrate the general geometric concepts, we shall use the Wyndor Glass Co. example presented in Sec. 3.1. (Sections 4.2 and 4.3 use the *algebra* of the simplex method to solve this same example.) Section 5.1 will elaborate further on these geometric concepts for larger problems.

To refresh your memory, the model and graph for this example are repeated in Fig. 4.1. The five constraint boundaries and their points of intersection are highlighted in this figure because they are the keys to the analysis. Here, each **constraint boundary** is a line that forms the boundary of what is permitted by the corresponding constraint. The points of intersection are the **corner-point solutions** of the problem. The five that lie on the corners of the *feasible region*—(0, 0), (0, 6), (2, 6), (4, 3), and (4, 0)—are the *corner-point feasible solutions* (**CPF solutions**). [The other three—(0, 9), (4, 6), and (6, 0)—are called *corner-point infeasible solutions*.]

In this example, each corner-point solution lies at the intersection of *two* constraint boundaries. (For a linear programming problem with n decision variables, each of its corner-point solutions lies at the intersection of n constraint boundaries.[2]) Certain pairs of the CPF solutions in Fig. 4.1 share a constraint boundary, and other pairs do not. It will be important to distinguish between these cases by using the following general definitions.

> For any linear programming problem with n decision variables, two CPF solutions are **adjacent** to each other if they share $n - 1$ constraint boundaries. The two adjacent CPF solutions are connected by a line segment that lies on these same shared constraint boundaries. Such a line segment is referred to as an **edge** of the feasible region.

Since $n = 2$ in the example, two of its CPF solutions are adjacent if they share *one* constraint boundary; for example, (0, 0) and (0, 6) are adjacent because they share the $x_1 = 0$ constraint boundary. The feasible region in Fig. 4.1 has five edges, consisting of the five line segments forming the boundary of this region. Note that two edges emanate from

[2]Although a corner-point solution is defined in terms of n constraint boundaries whose intersection gives this solution, it also is possible that one or more *additional* constraint boundaries pass through this same point.

TABLE 4.1 Adjacent CPF solutions for each CPF solution of the Wyndor Glass Co. problem

CPF Solution	Its Adjacent CPF Solutions
(0, 0)	(0, 6) and (4, 0)
(0, 6)	(2, 6) and (0, 0)
(2, 6)	(4, 3) and (0, 6)
(4, 3)	(4, 0) and (2, 6)
(4, 0)	(0, 0) and (4, 3)

each CPF solution. Thus, each CPF solution has two adjacent CPF solutions (each lying at the other end of one of the two edges), as enumerated in Table 4.1. (In each row of this table, the CPF solution in the first column is adjacent to each of the two CPF solutions in the second column, but the two CPF solutions in the second column are *not* adjacent to each other.)

One reason for our interest in adjacent CPF solutions is the following general property about such solutions, which provides a very useful way of checking whether a CPF solution is an optimal solution.

> **Optimality test:** Consider any linear programming problem that possesses at least one optimal solution. If a CPF solution has no *adjacent* CPF solutions that are *better* (as measured by Z), then it *must* be an *optimal* solution.

Thus, for the example, (2, 6) must be optimal simply because its $Z = 36$ is larger than $Z = 30$ for (0, 6) and $Z = 27$ for (4, 3). (We will delve further into why this property holds in Sec. 5.1.) This optimality test is the one used by the simplex method for determining when an optimal solution has been reached.

Now we are ready to apply the simplex method to the example.

Solving the Example

Here is an outline of what the simplex method does (from a geometric viewpoint) to solve the Wyndor Glass Co. problem. At each step, first the conclusion is stated and then the reason is given in parentheses. (Refer to Fig. 4.1 for a visualization.)

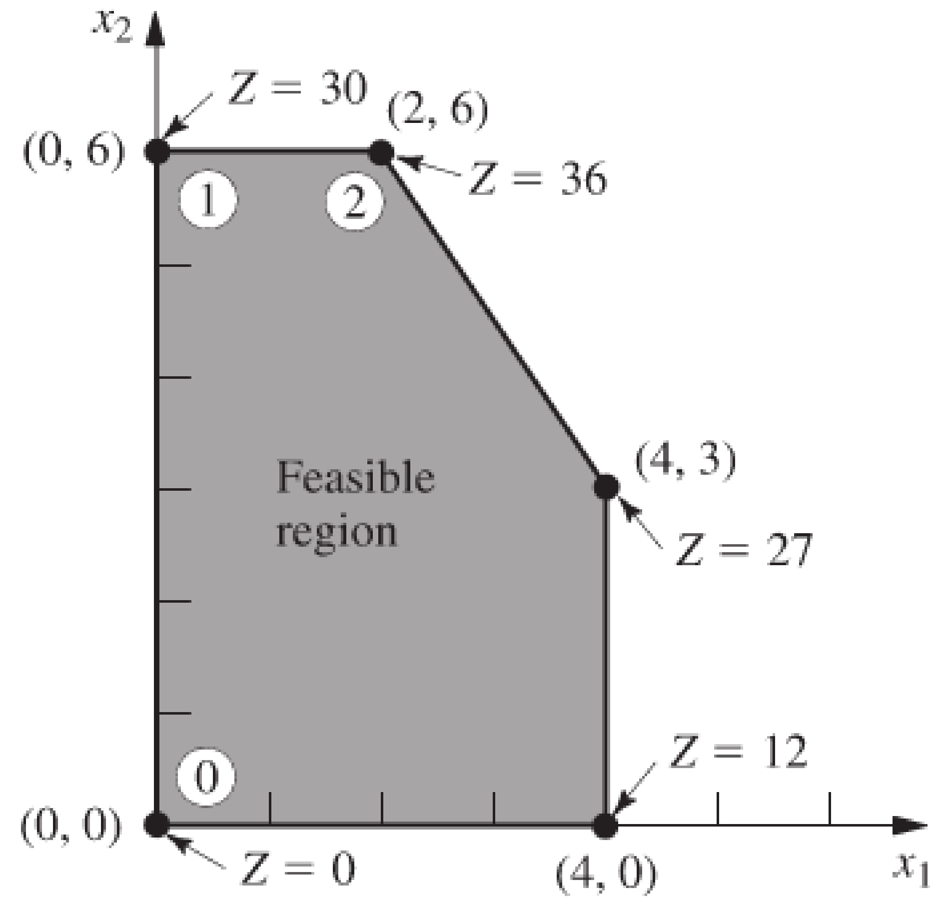
Initialization: Choose (0, 0) as the *initial* CPF solution to examine. (This is a convenient choice because no calculations are required to identify this CPF solution.)

Optimality Test: Conclude that (0, 0) is *not* an optimal solution. (Adjacent CPF solutions are better.)

Iteration 1: Move to a better *adjacent* CPF solution, (0, 6), by performing the following three steps.

1. Considering the two edges of the feasible region that emanate from (0, 0), choose to move along the edge that leads up the x_2 axis. (With an objective function of $Z = 3x_1 + 5x_2$, moving up the x_2 axis increases Z at a faster rate than moving along the x_1 axis.)
2. Stop at the first new constraint boundary: $2x_2 = 12$. [Moving farther in the direction selected in step 1 leaves the feasible region; e.g., moving to the second new constraint boundary hit when moving in that direction gives (0, 9), which is a corner-point *infeasible* solution.]
3. Solve for the intersection of the new set of constraint boundaries: (0, 6). (The equations for these constraint boundaries, $x_1 = 0$ and $2x_2 = 12$, immediately yield this solution.)

Optimality Test: Conclude that (0, 6) is *not* an optimal solution. (An adjacent CPF solution is better.)



■ **FIGURE 4.2**
This graph shows the sequence of CPF solutions (0, 1, 2) examined by the simplex method for the Wyndor Glass Co. problem. The optimal solution (2, 6) is found after just three solutions are examined.

Iteration 2: Move to a better adjacent CPF solution, (2, 6), by performing the following three steps.

1. Considering the two edges of the feasible region that emanate from (0, 6), choose to move along the edge that leads to the right. (Moving along this edge increases Z, whereas backtracking to move back down the x_2 axis decreases Z.)
2. Stop at the first new constraint boundary encountered when moving in that direction: $3x_1 + 2x_2 = 12$. (Moving farther in the direction selected in step 1 leaves the feasible region.)
3. Solve for the intersection of the new set of constraint boundaries: (2, 6). (The equations for these constraint boundaries, $3x_1 + 2x_2 = 18$ and $2x_2 = 12$, immediately yield this solution.)

Optimality Test: Conclude that (2, 6) *is* an optimal solution, so stop. (None of the adjacent CPF solutions are better.)

This sequence of CPF solutions examined is shown in Fig. 4.2, where each circled number identifies which iteration obtained that solution. (See the Worked Examples section on the book's website for **another example** of how the simplex method marches through a sequence of CPF solutions to reach the optimal solution.)

Now let us look at the six key solution concepts of the simplex method that provide the rationale behind the above steps. (Keep in mind that these concepts also apply for solving problems with more than two decision variables where a graph like Fig. 4.2 is not available to help quickly find an optimal solution.)

The Key Solution Concepts

The first solution concept is based directly on the relationship between optimal solutions and CPF solutions given at the end of Sec. 3.2.

> **Solution concept 1:** The simplex method focuses solely on CPF solutions. For any problem with at least one optimal solution, finding one requires only finding a best CPF solution.[3]

[3]The only restriction is that the problem must possess CPF solutions. This is ensured if the feasible region is bounded.

Since the number of feasible solutions generally is infinite, reducing the number of solutions that need to be examined to a small finite number (just three in Fig. 4.2) is a tremendous simplification.

The next solution concept defines the flow of the simplex method.

> **Solution concept 2:** The simplex method is an *iterative algorithm* (a systematic solution procedure that keeps repeating a fixed series of steps, called an *iteration,* until a desired result has been obtained) with the following structure.

Initialization:	Set up to start iterations, including finding an initial CPF solution.
↓	
→ **Optimality test:**	Is the current CPF solution optimal?
If no ↓ If yes ⟶	Stop.
Iteration:	Perform an iteration to find a better CPF solution.

When the example was solved, note how this flow diagram was followed through two iterations until an optimal solution was found.

We next focus on how to get started.

> **Solution concept 3:** Whenever possible, the initialization of the simplex method chooses the *origin* (all decision variables equal to zero) to be the initial CPF solution. When there are too many decision variables to find an initial CPF solution graphically, this choice eliminates the need to use algebraic procedures to find and solve for an initial CPF solution.

Choosing the origin commonly is possible when all the decision variables have nonnegativity constraints, because the intersection of these constraint boundaries yields the origin as a corner-point solution. This solution then is a CPF solution *unless* it is *infeasible* because it violates one or more of the functional constraints. If it is infeasible, special procedures described in Sec. 4.6 are needed to find the initial CPF solution.

The next solution concept concerns the choice of a better CPF solution at each iteration.

> **Solution concept 4:** Given a CPF solution, it is much quicker computationally to gather information about its *adjacent* CPF solutions than about other CPF solutions. Therefore, each time the simplex method performs an iteration to move from the current CPF solution to a better one, it *always* chooses a CPF solution that is *adjacent* to the current one. No other CPF solutions are considered. Consequently, the entire path followed to eventually reach an optimal solution is along the *edges* of the feasible region.

The next focus is on which adjacent CPF solution to choose at each iteration.

> **Solution concept 5:** After the current CPF solution is identified, the simplex method examines each of the edges of the feasible region that emanate from this CPF solution. Each of these edges leads to an *adjacent* CPF solution at the other end, but the simplex method does not even take the time to solve for the adjacent CPF solution. Instead, it simply identifies the *rate of improvement in Z* that would be obtained by moving along the edge. Among the edges with a *positive* rate of improvement in Z, it then chooses to move along the one with the *largest* rate of improvement in Z. The iteration is completed by first solving for the adjacent CPF solution at the other end of this one edge and then relabeling this adjacent

CPF solution as the *current* CPF solution for the optimality test and (if needed) the next iteration.

At the first iteration of the example, moving from (0, 0) along the edge on the x_1 axis would give a rate of improvement in Z of 3 (Z increases by 3 per unit increase in x_1), whereas moving along the edge on the x_2 axis would give a rate of improvement in Z of 5 (Z increases by 5 per unit increase in x_2), so the decision is made to move along the latter edge. At the second iteration, the only edge emanating from (0, 6) that would yield a *positive* rate of improvement in Z is the edge leading to (2, 6), so the decision is made to move next along this edge.

The final solution concept clarifies how the optimality test is performed efficiently.

Solution concept 6: Solution concept 5 describes how the simplex method examines each of the edges of the feasible region that emanate from the current CPF solution. This examination of an edge leads to quickly identifying the rate of improvement in Z that would be obtained by moving along the edge toward the adjacent CPF solution at the other end. A *positive* rate of improvement in Z implies that the adjacent CPF solution is *better* than the current CPF solution, whereas a *negative* rate of improvement in Z implies that the adjacent CPF solution is *worse*. Therefore, the optimality test consists simply of checking whether *any* of the edges give a *positive* rate of improvement in Z. If *none* do, then the current CPF solution is optimal.

In the example, moving along *either* edge from (2, 6) decreases Z. Since we want to maximize Z, this fact immediately gives the conclusion that (2, 6) is optimal.

4.2 SETTING UP THE SIMPLEX METHOD

Section 4.1 stressed the geometric concepts that underlie the simplex method. However, this algorithm normally is run on a computer, which can follow only algebraic instructions. Therefore, it is necessary to translate the conceptually geometric procedure just described into a usable algebraic procedure. In this section, we introduce the *algebraic language* of the simplex method and relate it to the concepts of the preceding section.

The algebraic procedure is based on solving systems of equations. Therefore, the first step in setting up the simplex method is to convert the functional *inequality constraints* to equivalent *equality constraints*. (The nonnegativity constraints are left as inequalities because they are treated separately.) This conversion is accomplished by introducing **slack variables.** To illustrate, consider the first functional constraint in the Wyndor Glass Co. example of Sec. 3.1

$$x_1 \leq 4.$$

The slack variable for this constraint is defined to be

$$x_3 = 4 - x_1,$$

which is the amount of slack in the left-hand side of the inequality. Thus,

$$x_1 + x_3 = 4.$$

Given this equation, $x_1 \leq 4$ if and only if $4 - x_1 = x_3 \geq 0$. Therefore, the original constraint $x_1 \leq 4$ is entirely *equivalent* to the pair of constraints

$$x_1 + x_3 = 4 \qquad \text{and} \qquad x_3 \geq 0.$$

Upon the introduction of slack variables for the other functional constraints, the original linear programming model for the example (shown below on the left) can now be replaced by the equivalent model (called the *augmented form* of the model) shown below on the right:

Original Form of the Model

Maximize $Z = 3x_1 + 5x_2$,

subject to

$$
\begin{aligned}
x_1 \quad\quad &\le 4 \\
2x_2 &\le 12 \\
3x_1 + 2x_2 &\le 18
\end{aligned}
$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0.$$

Augmented Form of the Model[4]

Maximize $Z = 3x_1 + 5x_2$,

subject to

$$
\begin{aligned}
&(1) \quad x_1 + x_3 = 4 \\
&(2) \quad 2x_2 + x_4 = 12 \\
&(3) \quad 3x_1 + 2x_2 + x_5 = 18
\end{aligned}
$$

and

$$x_j \ge 0, \qquad \text{for } j = 1, 2, 3, 4, 5.$$

Although both forms of the model represent exactly the same problem, the new form is much more convenient for algebraic manipulation and for identification of CPF solutions. We call this the **augmented form** of the problem because the original form has been *augmented* by some supplementary variables needed to apply the simplex method.

If a slack variable equals 0 in the current solution, then this solution lies on the constraint boundary for the corresponding functional constraint. A value greater than 0 means that the solution lies on the *feasible* side of this constraint boundary, whereas a value less than 0 means that the solution lies on the *infeasible* side of this constraint boundary. A demonstration of these properties is provided by the **demonstration example** in your OR Tutor entitled *Interpretation of the Slack Variables*.

The terminology used in Section 4.1 (corner-point solutions, etc.) applies to the original form of the problem. We now introduce the corresponding terminology for the augmented form.

An **augmented solution** is a solution for the original variables (the *decision variables*) that has been augmented by the corresponding values of the *slack variables*.

For example, augmenting the solution (3, 2) in the example yields the augmented solution (3, 2, 1, 8, 5) because the corresponding values of the slack variables are $x_3 = 1$, $x_4 = 8$, and $x_5 = 5$.

A **basic solution** is an *augmented* corner-point solution.

To illustrate, consider the corner-point infeasible solution (4, 6) in Fig. 4.1. Augmenting it with the resulting values of the slack variables $x_3 = 0$, $x_4 = 0$, and $x_5 = -6$ yields the corresponding basic solution (4, 6, 0, 0, −6).

The fact that corner-point solutions (and so basic solutions) can be either feasible or infeasible implies the following definition:

A **basic feasible (BF) solution** is an *augmented* CPF solution.

Thus, the CPF solution (0, 6) in the example is equivalent to the BF solution (0, 6, 4, 0, 6) for the problem in augmented form.

The only difference between basic solutions and corner-point solutions (or between BF solutions and CPF solutions) is whether the values of the slack variables are included.

[4]The slack variables are not shown in the objective function because the coefficients there are 0.

For any basic solution, the corresponding corner-point solution is obtained simply by deleting the slack variables. Therefore, the geometric and algebraic relationships between these two solutions are very close, as described in Sec. 5.1.

Because the terms *basic solution* and *basic feasible solution* are very important parts of the standard vocabulary of linear programming, we now need to clarify their algebraic properties. For the augmented form of the example, notice that the system of functional constraints has 5 variables and 3 equations, so

$$\text{Number of variables} - \text{number of equations} = 5 - 3 = 2.$$

This fact gives us 2 *degrees of freedom* in solving the system, since any two variables can be chosen to be set equal to any arbitrary value in order to solve the three equations in terms of the remaining three variables.[5] The simplex method uses zero for this arbitrary value. Thus, two of the variables (called the *nonbasic variables*) are set equal to zero, and then the simultaneous solution of the three equations for the other three variables (called the *basic variables*) is a *basic solution.* These properties are described in the following general definitions.

A **basic solution** has the following properties:

1. Each variable is designated as either a nonbasic variable or a basic variable.
2. The *number of basic variables* equals the number of functional constraints (now equations). Therefore, the *number of nonbasic variables* equals the total number of variables *minus* the number of functional constraints.
3. The **nonbasic variables** are set equal to zero.
4. The values of the **basic variables** are obtained as the simultaneous solution of the system of equations (functional constraints in augmented form). (The set of basic variables is often referred to as **the basis.**)
5. If the basic variables satisfy the *nonnegativity constraints,* the basic solution is a **BF solution.**

To illustrate these definitions, consider again the BF solution (0, 6, 4, 0, 6). This solution was obtained before by augmenting the CPF solution (0, 6). However, another way to obtain this same solution is to choose x_1 and x_4 to be the two nonbasic variables, and so the two variables are set equal to zero. The three equations then yield, respectively, $x_3 = 4$, $x_2 = 6$, and $x_5 = 6$ as the solution for the three basic variables, as shown below (with the basic variables in bold type):

$$\begin{array}{llllll} & & & & & x_1 = 0 \text{ and } x_4 = 0 \text{ so} \\ (1) & x_1 & + \mathbf{x_3} & & = 4 & \mathbf{x_3} = 4 \\ (2) & 2\mathbf{x_2} & & + x_4 & = 12 & \mathbf{x_2} = 6 \\ (3) & 3x_1 + 2\mathbf{x_2} & & + \mathbf{x_5} & = 18 & \mathbf{x_5} = 6 \end{array}$$

Because all three of these basic variables are nonnegative, this *basic solution* (0, 6, 4, 0, 6) is indeed a *BF solution*. The Worked Examples section of the book's website includes **another example** of the relationship between CPF solutions and BF solutions.

Just as certain pairs of CPF solutions are *adjacent,* the corresponding pairs of BF solutions also are said to be adjacent. Here is an easy way to tell when two BF solutions are adjacent.

> Two BF solutions are **adjacent** if *all but one* of their *nonbasic variables* are the same. This implies that *all but one* of their *basic variables* also are the same, although perhaps with different numerical values.

[5]This method of determining the number of degrees of freedom for a system of equations is valid as long as the system does not include any redundant equations. This condition always holds for the system of equations formed from the functional constraints in the augmented form of a linear programming model.

Consequently, moving from the current BF solution to an adjacent one involves switching one variable from nonbasic to basic and vice versa for one other variable (and then adjusting the values of the basic variables to continue satisfying the system of equations).

To illustrate *adjacent BF solutions,* consider one pair of adjacent CPF solutions in Fig. 4.1: (0, 0) and (0, 6). Their augmented solutions, (0, 0, 4, 12, 18) and (0, 6, 4, 0, 6), automatically are adjacent BF solutions. However, you do not need to look at Fig. 4.1 to draw this conclusion. Another signpost is that their nonbasic variables, (x_1, x_2) and (x_1, x_4), are the same with just the one exception—x_2 has been replaced by x_4. Consequently, moving from (0, 0, 4, 12, 18) to (0, 6, 4, 0, 6) involves switching x_2 from nonbasic to basic and vice versa for x_4.

When we deal with the problem in augmented form, it is convenient to consider and manipulate the objective function equation at the same time as the new constraint equations. Therefore, before we start the simplex method, the problem needs to be rewritten once again in an equivalent way:

$$\text{Maximize} \quad Z,$$

subject to

$$\begin{array}{llcl}
(0) & Z - 3x_1 - 5x_2 & & = 0 \\
(1) & \quad\;\; x_1 \quad\quad + x_3 & & = 4 \\
(2) & \quad\quad\quad\;\; 2x_2 \quad + x_4 & & = 12 \\
(3) & \quad\;\; 3x_1 + 2x_2 & + x_5 & = 18
\end{array}$$

and

$$x_j \geq 0, \qquad \text{for } j = 1, 2, \ldots, 5.$$

It is just as if Eq. (0) actually were one of the original constraints; but because it already is in equality form, no slack variable is needed. While adding one more equation, we also have added one more unknown (Z) to the system of equations. Therefore, when using Eqs. (1) to (3) to obtain a basic solution as described above, we use Eq. (0) to solve for Z at the same time.

Somewhat fortuitously, the model for the Wyndor Glass Co. problem fits *our standard form,* and all its functional constraints have nonnegative right-hand sides b_i. If this had not been the case, then additional adjustments would have been needed at this point before the simplex method was applied. These details are deferred to Sec. 4.6, and we now focus on the simplex method itself.

4.3 THE ALGEBRA OF THE SIMPLEX METHOD

We continue to use the prototype example of Sec. 3.1, as rewritten at the end of Sec. 4.2, for illustrative purposes. To start connecting the geometric and algebraic concepts of the simplex method, we begin by outlining side by side in Table 4.2 how the simplex method solves this example from both a geometric and an algebraic viewpoint. The geometric viewpoint (first presented in Sec. 4.1) is based on the *original form* of the model (no slack variables), so again refer to Fig. 4.1 for a visualization when you examine the second column of the table. Refer to the *augmented form* of the model presented at the end of Sec. 4.2 when you examine the third column of the table.

We now fill in the details for each step of the third column of Table 4.2.

TABLE 4.2 Geometric and algebraic interpretations of how the simplex method solves the Wyndor Glass Co. problem

Method Sequence	Geometric Interpretation	Algebraic Interpretation
Initialization	Choose (0, 0) to be the initial CPF solution.	Choose x_1 and x_2 to be the nonbasic variables (= 0) for the initial BF solution: (0, 0, 4, 12, 18).
Optimality test	Not optimal, because moving along either edge from (0, 0) increases Z.	Not optimal, because increasing either nonbasic variable (x_1 or x_2) increases Z.
Iteration 1		
Step 1	Move up the edge lying on the x_2 axis.	Increase x_2 while adjusting other variable values to satisfy the system of equations.
Step 2	Stop when the first new constraint boundary ($2x_2 = 12$) is reached.	Stop when the first basic variable (x_3, x_4, or x_5) drops to zero (x_4).
Step 3	Find the intersection of the new pair of constraint boundaries: (0, 6) is the new CPF solution.	With x_2 now a basic variable and x_4 now a nonbasic variable, solve the system of equations: (0, 6, 4, 0, 6) is the new BF solution.
Optimality test	Not optimal, because moving along the edge from (0, 6) to the right increases Z.	Not optimal, because increasing one nonbasic variable (x_1) increases Z.
Iteration 2		
Step 1	Move along this edge to the right.	Increase x_1 while adjusting other variable values to satisfy the system of equations.
Step 2	Stop when the first new constraint boundary ($3x_1 + 2x_2 = 18$) is reached.	Stop when the first basic variable (x_2, x_3, or x_5) drops to zero (x_5).
Step 3	Find the intersection of the new pair of constraint boundaries: (2, 6) is the new CPF solution.	With x_1 now a basic variable and x_5 now a nonbasic variable, solve the system of equations: (2, 6, 2, 0, 0) is the new BF solution.
Optimality test	(2, 6) is optimal, because moving along either edge from (2, 6) decreases Z.	(2, 6, 2, 0, 0) is optimal, because increasing either nonbasic variable (x_4 or x_5) decreases Z.

Initialization

The choice of x_1 and x_2 to be the *nonbasic* variables (the variables set equal to zero) for the initial BF solution is based on solution concept 3 in Sec. 4.1. This choice eliminates the work required to solve for the *basic variables* (x_3, x_4, x_5) from the following system of equations (where the basic variables are shown in bold type):

$$
\begin{array}{llr}
 & & x_1 = 0 \text{ and } x_2 = 0 \text{ so} \\
(1) & x_1 + \boldsymbol{x_3} = 4 & \boldsymbol{x_3} = 4 \\
(2) & 2x_2 + \boldsymbol{x_4} = 12 & \boldsymbol{x_4} = 12 \\
(3) & 3x_1 + 2x_2 + \boldsymbol{x_5} = 18 & \boldsymbol{x_5} = 18
\end{array}
$$

Thus, the **initial BF solution** is (0, 0, 4, 12, 18).

Notice that this solution can be read immediately because each equation has just one basic variable, which has a coefficient of 1, and this basic variable does not appear in any other equation. You will soon see that when the set of basic variables changes, the simplex method uses an algebraic procedure (Gaussian elimination) to convert the equations to this same convenient form for reading every subsequent BF solution as well. This form is called **proper form from Gaussian elimination.**

An Application Vignette

Samsung Electronics Corp., Ltd. (SEC) is a leading merchant of dynamic and static random access memory devices and other advanced digital integrated circuits. Its site at Kiheung, South Korea (probably the largest semiconductor fabrication site in the world) fabricates more than 300,000 silicon wafers per month and employs over 10,000 people.

Cycle time is the industry's term for the elapsed time from the release of a batch of blank silicon wafers into the fabrication process until completion of the devices that are fabricated on those wafers. Reducing cycle times is an ongoing goal since it both decreases costs and enables offering shorter lead times to potential customers, a real key to maintaining or increasing market share in a very competitive industry.

Three factors present particularly major challenges when striving to reduce cycle times. One is that the product mix changes continually. Another is that the company often needs to make substantial changes in the fab-out schedule inside the target cycle time as it revises forecasts of customer demand. The third is that the machines of a general type are not homogenous so only a small number of machines are qualified to perform each device-step.

An OR team developed *a huge linear programming model with tens of thousands of decision variables and functional constraints* to cope with these challenges. The objective function involved minimizing back-orders and finished-goods inventory. Despite the huge size of this model, it was readily solved in minutes whenever needed by using a highly sophisticated implementation of the simplex method (and related techniques) in the CPLEX optimization software. (CPLEX will be discussed further in Sec. 4.8.)

The ongoing implementation of this model enabled the company to reduce manufacturing cycle times to fabricate dynamic random access memory devices from more than 80 days to less than 30 days. This tremendous improvement and the resulting reduction in both manufacturing costs and sale prices enabled Samsung to capture *an additional* **$200 million** *in annual sales revenue*.

Source: R. C. Leachman, J. Kang, and Y. Lin: "SLIM: Short Cycle Time and Low Inventory in Manufacturing at Samsung Electronics," *Interfaces*, **32**(1): 61–77, Jan.–Feb. 2002. (A link to this article is provided on our website, www.mhhe.com/hillier.)

Optimality Test

The objective function is

$$Z = 3x_1 + 5x_2,$$

so $Z = 0$ for the initial BF solution. Because none of the basic variables (x_3, x_4, x_5) have a *nonzero* coefficient in this objective function, the coefficient of each nonbasic variable (x_1, x_2) gives the rate of improvement in Z if that variable were to be increased from zero (while the values of the basic variables are adjusted to continue satisfying the system of equations).[6] These rates of improvement (3 and 5) are *positive*. Therefore, based on solution concept 6 in Sec. 4.1, we conclude that (0, 0, 4, 12, 18) is not optimal.

For each BF solution examined after subsequent iterations, at least one basic variable has a nonzero coefficient in the objective function. Therefore, the optimality test then will use the new Eq. (0) to rewrite the objective function in terms of just the nonbasic variables, as you will see later.

Determining the Direction of Movement (Step 1 of an Iteration)

Increasing one nonbasic variable from zero (while adjusting the values of the basic variables to continue satisfying the system of equations) corresponds to moving along one edge emanating from the current CPF solution. Based on solution concepts 4 and 5 in Sec. 4.1, the choice of which nonbasic variable to increase is made as follows:

[6]Note that this interpretation of the coefficients of the x_j variables is based on these variables being on the right-hand side, $Z = 3x_1 + 5x_2$. When these variables are brought to the left-hand side for Eq. (0), $Z - 3x_1 - 5x_2 = 0$, the nonzero coefficients change their signs.

$Z = 3x_1 + 5x_2$

Increase x_1? Rate of improvement in $Z = 3$.

Increase x_2? Rate of improvement in $Z = 5$.

$5 > 3$, so choose x_2 to increase.

As indicated next, we call x_2 the *entering basic variable* for iteration 1.

> At any iteration of the simplex method, the purpose of step 1 is to choose one *nonbasic variable* to increase from zero (while the values of the basic variables are adjusted to continue satisfying the system of equations). Increasing this nonbasic variable from zero will convert it to a *basic variable* for the next BF solution. Therefore, this variable is called the **entering basic variable** for the current iteration (because it is entering the basis).

Determining Where to Stop (Step 2 of an Iteration)

Step 2 addresses the question of how far to increase the entering basic variable x_2 before stopping. Increasing x_2 increases Z, so we want to go as far as possible without leaving the feasible region. The requirement to satisfy the functional constraints in augmented form (shown below) means that increasing x_2 (while keeping the nonbasic variable $x_1 = 0$) changes the values of some of the basic variables as shown on the right.

$$\begin{array}{llllll} & & & & x_1 = 0, & \text{so} \\ (1) & x_1 & + \boldsymbol{x_3} & = 4 & \boldsymbol{x_3} = 4 & \\ (2) & 2x_2 & + \boldsymbol{x_4} & = 12 & \boldsymbol{x_4} = 12 - 2x_2 & \\ (3) & 3x_1 + 2x_2 & + \boldsymbol{x_5} & = 18 & \boldsymbol{x_5} = 18 - 2x_2. & \end{array}$$

The other requirement for feasibility is that all the variables be *nonnegative*. The nonbasic variables (including the entering basic variable) are nonnegative, but we need to check how far x_2 can be increased without violating the nonnegativity constraints for the basic variables.

$$\boldsymbol{x_3} = 4 \geq 0 \quad \Rightarrow \text{no upper bound on } x_2.$$

$$\boldsymbol{x_4} = 12 - 2x_2 \geq 0 \Rightarrow x_2 \leq \frac{12}{2} = 6 \quad \leftarrow \text{minimum.}$$

$$\boldsymbol{x_5} = 18 - 2x_2 \geq 0 \Rightarrow x_2 \leq \frac{18}{2} = 9.$$

Thus, x_2 can be increased just to 6, at which point $\boldsymbol{x_4}$ has dropped to 0. Increasing x_2 beyond 6 would cause $\boldsymbol{x_4}$ to become negative, which would violate feasibility.

These calculations are referred to as the **minimum ratio test.** The objective of this test is to determine which basic variable drops to zero first as the entering basic variable is increased. We can immediately rule out the basic variable in any equation where the coefficient of the entering basic variable is zero or negative, since such a basic variable would not decrease as the entering basic variable is increased. [This is what happened with x_3 in Eq. (1) of the example.] However, for each equation where the coefficient of the entering basic variable is *strictly positive* (> 0), this test calculates the *ratio* of the right-hand side to the coefficient of the entering basic variable. The basic variable in the equation with the *minimum ratio* is the one that drops to zero first as the entering basic variable is increased.

> At any iteration of the simplex method, step 2 uses the *minimum ratio test* to determine which basic variable drops to zero first as the entering basic variable is increased. Decreasing this basic variable to zero will convert it to a *nonbasic variable* for the next BF solution. Therefore, this variable is called the **leaving basic variable** for the current iteration (because it is leaving the basis).

Thus, $\boldsymbol{x_4}$ is the leaving basic variable for iteration 1 of the example.

Solving for the New BF Solution (Step 3 of an Iteration)

Increasing $x_2 = 0$ to $x_2 = 6$ moves us from the *initial* BF solution on the left to the *new* BF solution on the right.

	Initial BF solution	New BF solution
Nonbasic variables:	$x_1 = 0, \quad x_2 = 0$	$x_1 = 0, \quad x_4 = 0$
Basic variables:	$x_3 = 4, \quad x_4 = 12, \quad x_5 = 18$	$x_3 = ?, \quad x_2 = 6, \quad x_5 = ?$

The purpose of step 3 is to convert the system of equations to a more convenient form (proper form from Gaussian elimination) for conducting the optimality test and (if needed) the next iteration with this new BF solution. In the process, this form also will identify the values of x_3 and x_5 for the new solution.

Here again is the complete original system of equations, where the *new* basic variables are shown in bold type (with Z playing the role of the basic variable in the objective function equation):

$$\begin{array}{llllllll}
(0) & \mathbf{Z} - 3x_1 - 5\boldsymbol{x_2} & & & & = & 0 \\
(1) & \quad\; x_1 & + \boldsymbol{x_3} & & & = & 4 \\
(2) & \qquad\quad 2\boldsymbol{x_2} & & + x_4 & & = & 12 \\
(3) & \quad\; 3x_1 + 2\boldsymbol{x_2} & & & + \boldsymbol{x_5} & = & 18.
\end{array}$$

Thus, x_2 has replaced x_4 as the basic variable in Eq. (2). To solve this system of equations for Z, x_2, x_3, and x_5, we need to perform some **elementary algebraic operations** to reproduce the current pattern of coefficients of x_4 (0, 0, 1, 0) as the new coefficients of x_2. We can use either of two types of elementary algebraic operations:

1. Multiply (or divide) an equation by a nonzero constant.
2. Add (or subtract) a multiple of one equation to (or from) another equation.

To prepare for performing these operations, note that the coefficients of x_2 in the above system of equations are -5, 0, 2, and 2, respectively, whereas we want these coefficients to become 0, 0, 1, and 0, respectively. To turn the coefficient of 2 in Eq. (2) into 1, we use the first type of elementary algebraic operation by dividing Eq. (2) by 2 to obtain

$$(2) \qquad \boldsymbol{x_2} + \frac{1}{2}x_4 = 6.$$

To turn the coefficients of -5 and 2 into zeros, we need to use the second type of elementary algebraic operation. In particular, we add 5 times this new Eq. (2) to Eq. (0), and subtract 2 times this new Eq. (2) from Eq. (3). The resulting complete new system of equations is

$$\begin{array}{llllllll}
(0) & \mathbf{Z} - 3x_1 & & & + \frac{5}{2}x_4 & & = & 30 \\
(1) & \quad\; x_1 & & + \boldsymbol{x_3} & & & = & 4 \\
(2) & & \boldsymbol{x_2} & & + \frac{1}{2}x_4 & & = & 6 \\
(3) & \quad\; 3x_1 & & & - \; x_4 & + \boldsymbol{x_5} & = & 6.
\end{array}$$

Since $x_1 = 0$ and $x_4 = 0$, the equations in this form immediately yield the new BF solution, $(x_1, x_2, x_3, x_4, x_5) = (0, 6, 4, 0, 6)$, which yields $Z = 30$.

This procedure for obtaining the simultaneous solution of a system of linear equations is called the *Gauss-Jordan method of elimination,* or **Gaussian elimination** for short.[7] The key concept for this method is the use of elementary algebraic operations to reduce the original system of equations to proper form from Gaussian elimination, where each basic variable has been eliminated from all but one equation (*its* equation) and has a coefficient of +1 in that equation.

Optimality Test for the New BF Solution

The current Eq. (0) gives the value of the objective function in terms of just the current nonbasic variables

$$Z = 30 + 3x_1 - \frac{5}{2}x_4.$$

Increasing either of these nonbasic variables from zero (while adjusting the values of the basic variables to continue satisfying the system of equations) would result in moving toward one of the two *adjacent* BF solutions. Because x_1 has a *positive* coefficient, increasing x_1 would lead to an adjacent BF solution that is better than the current BF solution, so the current solution is not optimal.

Iteration 2 and the Resulting Optimal Solution

Since $Z = 30 + 3x_1 - \frac{5}{2}x_4$, Z can be increased by increasing x_1, but not x_4. Therefore, step 1 chooses x_1 to be the entering basic variable.

For step 2, the current system of equations yields the following conclusions about how far x_1 can be increased (with $x_4 = 0$):

$$\begin{aligned}
\boldsymbol{x_3} &= 4 - x_1 \geq 0 && \Rightarrow x_1 \leq \frac{4}{1} = 4. \\
\boldsymbol{x_2} &= 6 \geq 0 && \Rightarrow \text{no upper bound on } x_1. \\
\boldsymbol{x_5} &= 6 - 3x_1 \geq 0 && \Rightarrow x_1 \leq \frac{6}{3} = 2 \quad \leftarrow \text{minimum.}
\end{aligned}$$

Therefore, the minimum ratio test indicates that x_5 is the leaving basic variable.

For step 3, with x_1 replacing x_5 as a basic variable, we perform elementary algebraic operations on the current system of equations to reproduce the current pattern of coefficients of x_5 (0, 0, 0, 1) as the new coefficients of x_1. This yields the following new system of equations:

$$\begin{aligned}
&(0) \quad \boldsymbol{Z} && && + \frac{3}{2}x_4 + x_5 = 36 \\
&(1) && && \boldsymbol{x_3} + \frac{1}{3}x_4 - \frac{1}{3}x_5 = 2 \\
&(2) && \boldsymbol{x_2} && + \frac{1}{2}x_4 \qquad\quad = 6 \\
&(3) \quad \boldsymbol{x_1} && && - \frac{1}{3}x_4 + \frac{1}{3}x_5 = 2.
\end{aligned}$$

Therefore, the next BF solution is $(x_1, x_2, x_3, x_4, x_5) = (2, 6, 2, 0, 0)$, yielding $Z = 36$. To apply the *optimality test* to this new BF solution, we use the current Eq. (0) to express Z in terms of just the current nonbasic variables,

[7]Actually, there are some technical differences between the Gauss-Jordan method of elimination and Gaussian elimination, but we shall not make this distinction.

$$Z = 36 - \frac{3}{2}x_4 - x_5.$$

Increasing either x_4 or x_5 would *decrease Z,* so neither adjacent BF solution is as good as the current one. Therefore, based on solution concept 6 in Sec. 4.1, the current BF solution must be optimal.

In terms of the original form of the problem (no slack variables), the optimal solution is $x_1 = 2$, $x_2 = 6$, which yields $Z = 3x_1 + 5x_2 = 36$.

To see **another example** of applying the simplex method, we recommend that you now view the demonstration entitled *Simplex Method—Algebraic Form* in your OR Tutor. This vivid demonstration simultaneously displays both the algebra and the geometry of the simplex method as it dynamically evolves step by step. Like the many other demonstration examples accompanying other sections of the book (including the next section), this computer demonstration highlights concepts that are difficult to convey on the printed page. In addition, the Worked Examples section of the book's website includes **another example** of applying the simplex method.

To further help you learn the simplex method efficiently, the IOR Tutorial in your OR Courseware includes a procedure entitled **Solve Interactively by the Simplex Method.** This routine performs nearly all the calculations while you make the decisions step by step, thereby enabling you to focus on concepts rather than get bogged down in a lot of number crunching. Therefore, you probably will want to use this routine for your homework on this section. The software will help you get started by letting you know whenever you make a mistake on the first iteration of a problem.

After you learn the simplex method, you will want to simply apply an automatic computer implementation of it to obtain optimal solutions of linear programming problems immediately. For your convenience, we also have included an automatic procedure called **Solve Automatically by the Simplex Method** in IOR Tutorial. This procedure is designed for dealing with only textbook-sized problems, including checking the answer you got with the interactive procedure. Section 4.8 will describe more powerful software options for linear programming that also are provided on the book's website.

The next section includes a summary of the simplex method for a more convenient tabular form.

4.4 THE SIMPLEX METHOD IN TABULAR FORM

The algebraic form of the simplex method presented in Sec. 4.3 may be the best one for learning the underlying logic of the algorithm. However, it is not the most convenient form for performing the required calculations. When you need to solve a problem by hand (or interactively with your IOR Tutorial), we recommend the *tabular form* described in this section.[8]

The tabular form of the simplex method records only the essential information, namely, (1) the coefficients of the variables, (2) the constants on the right-hand sides of the equations, and (3) the basic variable appearing in each equation. This saves writing the symbols for the variables in each of the equations, but what is even more important is the fact that it permits highlighting the numbers involved in arithmetic calculations and recording the computations compactly.

Table 4.3 compares the initial system of equations for the Wyndor Glass Co. problem in algebraic form (on the left) and in tabular form (on the right), where the table on the right is called a *simplex tableau.* The basic variable for each equation is shown in bold type

[8]A form more convenient for automatic execution on a computer is presented in Sec. 5.2.

TABLE 4.3 Initial system of equations for the Wyndor Glass Co. problem

(a) Algebraic Form	(b) Tabular Form									
	Basic Variable	Eq.	Coefficient of: Z	x_1	x_2	x_3	x_4	x_5	Right Side
(0) $Z - 3x_1 - 5x_2 = 0$	Z	(0)	1	−3	−5	0	0	0	0
(1) $x_1 + x_3 = 4$	x_3	(1)	0	1	0	1	0	0	4
(2) $2x_2 + x_4 = 12$	x_4	(2)	0	0	2	0	1	0	12
(3) $3x_1 + 2x_2 + x_5 = 18$	x_5	(3)	0	3	2	0	0	1	18

on the left and in the first column of the simplex tableau on the right. [Although only the x_j variables are basic or nonbasic, Z plays the role of the basic variable for Eq. (0).] All variables *not* listed in this *basic variable* column (x_1, x_2) automatically are *nonbasic variables.* After we set $x_1 = 0$, $x_2 = 0$, the *right side* column gives the resulting solution for the basic variables, so that the initial BF solution is $(x_1, x_2, x_3, x_4, x_5) = (0, 0, 4, 12, 18)$ which yields $Z = 0$.

> The *tabular form* of the simplex method uses a **simplex tableau** to compactly display the system of equations yielding the current BF solution. For this solution, each variable in the leftmost column equals the corresponding number in the rightmost column (and variables not listed equal zero). When the optimality test or an iteration is performed, the only relevant numbers are those to the right of the Z column.[9] The term **row** refers to just a row of numbers to the right of the Z column (including the *right side* number), where row i corresponds to Eq. (i).

We summarize the tabular form of the simplex method below and, at the same time, briefly describe its application to the Wyndor Glass Co. problem. Keep in mind that the logic is identical to that for the algebraic form presented in the preceding section. Only the form for displaying both the current system of equations and the subsequent iteration has changed (plus we shall no longer bother to bring variables to the right-hand side of an equation before drawing our conclusions in the optimality test or in steps 1 and 2 of an iteration).

Summary of the Simplex Method (and Iteration 1 for the Example)

Initialization. Introduce slack variables. Select the *decision variables* to be the *initial nonbasic variables* (set equal to zero) and the *slack variables* to be the *initial basic variables.* (See Sec. 4.6 for the necessary adjustments if the model is not in our standard form—maximization, only $\leq$ functional constraints, and all nonnegativity constraints—or if any b_i values are negative.)

For the Example: This selection yields the initial simplex tableau shown in column (b) of Table 4.3, so the initial BF solution is (0, 0, 4, 12, 18).

Optimality Test. The current BF solution is optimal if and only if *every* coefficient in row 0 is nonnegative (≥ 0). If it is, stop; otherwise, go to an iteration to obtain the next BF solution, which involves changing one nonbasic variable to a basic variable (step 1) and vice versa (step 2) and then solving for the new solution (step 3).

For the Example: Just as $Z = 3x_1 + 5x_2$ indicates that increasing either x_1 or x_2 will increase Z, so the current BF solution is not optimal, the same conclusion is drawn from

[9]For this reason, it is permissible to delete the Eq. and Z columns to reduce the size of the simplex tableau. We prefer to retain these columns as a reminder that the simplex tableau is displaying the current system of equations and that Z is one of the variables in Eq. (0).

the equation $Z - 3x_1 - 5x_2 = 0$. These coefficients of -3 and -5 are shown in row 0 in column (*b*) of Table 4.3.

Iteration. *Step 1:* Determine the *entering basic variable* by selecting the variable (automatically a nonbasic variable) with the *negative coefficient* having the largest absolute value (i.e., the "most negative" coefficient) in Eq. (0). Put a box around the column below this coefficient, and call this the **pivot column.**

For the Example: The most negative coefficient is -5 for x_2 ($5 > 3$), so x_2 is to be changed to a basic variable. (This change is indicated in Table 4.4 by the box around the x_2 column below -5.)

Step 2: Determine the *leaving basic variable* by applying the minimum ratio test.

Minimum Ratio Test

1. Pick out each coefficient in the pivot column that is strictly positive (> 0).
2. Divide each of these coefficients into the *right side* entry for the same row.
3. Identify the row that has the *smallest* of these ratios.
4. The basic variable for that row is the leaving basic variable, so replace that variable by the entering basic variable in the basic variable column of the next simplex tableau.

Put a box around this row and call it the **pivot row.** Also call the number that is in *both* boxes the **pivot number.**

For the Example: The calculations for the minimum ratio test are shown to the right of Table 4.4. Thus, row 2 is the pivot row (see the box around this row in the first simplex tableau of Table 4.5), and x_4 is the leaving basic variable. In the next simplex tableau (see the bottom of Table 4.5), x_2 replaces x_4 as the basic variable for row 2.

Step 3: Solve for the *new BF solution* by using **elementary row operations** (multiply or divide a row by a nonzero constant; add or subtract a multiple of one row to another row) to construct a new simplex tableau in proper form from Gaussian elimination below the current one, and then return to the optimality test. The specific elementary row operations that need to be performed are listed below.

1. Divide the pivot row by the pivot number. Use this *new* pivot row in steps 2 and 3.
2. For each other row (including row 0) that has a *negative* coefficient in the pivot column, *add* to this row the *product* of the absolute value of this coefficient and the new pivot row.
3. For each other row that has a *positive* coefficient in the pivot column, *subtract* from this row the *product* of this coefficient and the new pivot row.

TABLE 4.4 Applying the minimum ratio test to determine the first leaving basic variable for the Wyndor Glass Co. problem

Basic Variable	Eq.	Coefficient of: Z	x_1	x_2	x_3	x_4	x_5	Right Side	Ratio
Z	(0)	1	-3	-5	0	0	0	0	
x_3	(1)	0	1	$\boxed{0}$	1	0	0	4	
x_4	(2)	0	0	$\boxed{2}$	0	1	0	$12 \rightarrow$	$\frac{12}{2} = 6 \leftarrow$ minimum
x_5	(3)	0	3	$\boxed{2}$	0	0	1	$18 \rightarrow$	$\frac{18}{2} = 9$

TABLE 4.5 Simplex tableaux for the Wyndor Glass Co. problem after the first pivot row is divided by the first pivot number

Iteration	Basic Variable	Eq.	Coefficient of:						Right Side
			Z	x_1	x_2	x_3	x_4	x_5	
0	Z	(0)	1	−3	−5	0	0	0	0
	x_3	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	2	0	1	0	12
	x_5	(3)	0	3	2	0	0	1	18
1	Z	(0)	1						
	x_3	(1)	0						
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	6
	x_5	(3)	0						

For the Example: Since x_2 is replacing x_4 as a basic variable, we need to reproduce the first tableau's pattern of coefficients in the column of x_4 (0, 0, 1, 0) in the second tableau's column of x_2. To start, divide the pivot row (row 2) by the pivot number (2), which gives the new row 2 shown in Table 4.5. Next, we add to row 0 the product, 5 times the new row 2. Then we subtract from row 3 the product, 2 times the new row 2 (or equivalently, subtract from row 3 the *old* row 2). These calculations yield the new tableau shown in Table 4.6 for iteration 1. Thus, the new BF solution is (0, 6, 4, 0, 6), with $Z = 30$. We next return to the optimality test to check if the new BF solution is optimal. Since the new row 0 still has a negative coefficient (-3 for x_1), the solution is not optimal, and so at least one more iteration is needed.

Iteration 2 for the Example and the Resulting Optimal Solution

The second iteration starts anew from the second tableau of Table 4.6 to find the next BF solution. Following the instructions for steps 1 and 2, we find x_1 as the entering basic variable and x_5 as the leaving basic variable, as shown in Table 4.7.

For step 3, we start by dividing the pivot row (row 3) in Table 4.7 by the pivot number (3). Next, we add to row 0 the product, 3 times the new row 3. Then we subtract the new row 3 from row 1.

We now have the set of tableaux shown in Table 4.8. Therefore, the new BF solution is (2, 6, 2, 0, 0), with $Z = 36$. Going to the optimality test, we find that this solution is

TABLE 4.6 First two simplex tableaux for the Wyndor Glass Co. problem

Iteration	Basic Variable	Eq.	Coefficient of:						Right Side
			Z	x_1	x_2	x_3	x_4	x_5	
0	Z	(0)	1	−3	−5	0	0	0	0
	x_3	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	2	0	1	0	12
	x_5	(3)	0	3	2	0	0	1	18
1	Z	(0)	1	−3	0	0	$\frac{5}{2}$	0	30
	x_3	(1)	0	1	0	1	0	0	4
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	6
	x_5	(3)	0	3	0	0	−1	1	6

TABLE 4.7 Steps 1 and 2 of iteration 2 for the Wyndor Glass Co. problem

Iteration	Basic Variable	Eq.	Coefficient of: Z	x_1	x_2	x_3	x_4	x_5	Right Side	Ratio
1	Z	(0)	1	−3	0	0	$\frac{5}{2}$	0	30	
	x_3	(1)	0	$\boxed{1}$	0	1	0	0	4	$\frac{4}{1} = 4$
	x_2	(2)	0	$\boxed{0}$	1	0	$\frac{1}{2}$	0	6	
	x_5	(3)	0	$\boxed{3}$	0	0	−1	1	$\boxed{6}$	$\frac{6}{3} = 2 \leftarrow$ minimum

optimal because none of the coefficients in row 0 is negative, so the algorithm is finished. Consequently, the optimal solution for the Wyndor Glass Co. problem (before slack variables are introduced) is $x_1 = 2$, $x_2 = 6$.

Now compare Table 4.8 with the work done in Sec. 4.3 to verify that these two forms of the simplex method really are *equivalent*. Then note how the algebraic form is superior for learning the logic behind the simplex method, but the tabular form organizes the work being done in a considerably more convenient and compact form. We generally use the tabular form from now on.

An **additional example** of applying the simplex method in tabular form is available to you in the OR Tutor. See the demonstration entitled *Simplex Method—Tabular Form.* **Another example** also is included in the Worked Examples section of the book's website.

TABLE 4.8 Complete set of simplex tableaux for the Wyndor Glass Co. problem

Iteration	Basic Variable	Eq.	Coefficient of: Z	x_1	x_2	x_3	x_4	x_5	Right Side
0	Z	(0)	1	−3	−5	0	0	0	0
	x_3	(1)	0	1	$\boxed{0}$	1	0	0	4
	x_4	(2)	0	0	$\boxed{2}$	0	1	0	$\boxed{12}$
	x_5	(3)	0	3	$\boxed{2}$	0	0	1	18
1	Z	(0)	1	−3	0	0	$\frac{5}{2}$	0	30
	x_3	(1)	0	$\boxed{1}$	0	1	0	0	4
	x_2	(2)	0	$\boxed{0}$	1	0	$\frac{1}{2}$	0	6
	x_5	(3)	0	$\boxed{3}$	0	0	−1	1	$\boxed{6}$
2	Z	(0)	1	0	0	0	$\frac{3}{2}$	1	36
	x_3	(1)	0	0	0	1	$\frac{1}{3}$	$-\frac{1}{3}$	2
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	6
	x_1	(3)	0	1	0	0	$-\frac{1}{3}$	$\frac{1}{3}$	2

4.5 TIE BREAKING IN THE SIMPLEX METHOD

You may have noticed in the preceding two sections that we never said what to do if the various choice rules of the simplex method do not lead to a clear-cut decision, because of either ties or other similar ambiguities. We discuss these details now.

Tie for the Entering Basic Variable

Step 1 of each iteration chooses the nonbasic variable having the *negative* coefficient with the *largest absolute value* in the current Eq. (0) as the entering basic variable. Now suppose that two or more nonbasic variables are tied for having the largest negative coefficient (in absolute terms). For example, this would occur in the first iteration for the Wyndor Glass Co. problem if its objective function were changed to $Z = 3x_1 + 3x_2$, so that the initial Eq. (0) became $Z - 3x_1 - 3x_2 = 0$. How should this tie be broken?

The answer is that the selection between these contenders may be made *arbitrarily.* The optimal solution will be reached eventually, regardless of the tied variable chosen, and there is no convenient method for predicting in advance which choice will lead there sooner. In this example, the simplex method happens to reach the optimal solution (2, 6) in three iterations with x_1 as the initial entering basic variable, versus two iterations if x_2 is chosen.

Tie for the Leaving Basic Variable—Degeneracy

Now suppose that two or more basic variables tie for being the leaving basic variable in step 2 of an iteration. Does it matter which one is chosen? Theoretically it does, and in a very critical way, because of the following sequence of events that could occur. First, all the tied basic variables reach zero simultaneously as the entering basic variable is increased. Therefore, the one or ones *not* chosen to be the leaving basic variable also will have a value of zero in the new BF solution. (Note that basic variables with a value of *zero* are called **degenerate,** and the same term is applied to the corresponding BF solution.) Second, if one of these degenerate basic variables retains its value of zero until it is chosen at a subsequent iteration to be a leaving basic variable, the corresponding entering basic variable also must remain zero (since it cannot be increased without making the leaving basic variable negative), so the value of Z must remain unchanged. Third, if Z may remain the same rather than increase at each iteration, the simplex method may then go around in a loop, repeating the same sequence of solutions periodically rather than eventually increasing Z toward an optimal solution. In fact, examples have been artificially constructed so that they do become entrapped in just such a perpetual loop.[10]

Fortunately, although a perpetual loop is theoretically possible, it has rarely been known to occur in practical problems. If a loop were to occur, one could always get out of it by changing the choice of the leaving basic variable. Furthermore, special rules[11] have been constructed for breaking ties so that such loops are always avoided. However, these rules frequently are ignored in actual application, and they will not be repeated here. For your purposes, just break this kind of tie arbitrarily and proceed without worrying about the degenerate basic variables that result.

[10]For further information about cycling around a perpetual loop, see J. A. J. Hall and K. I. M. McKinnon: "The Simplest Examples Where the Simplex Method Cycles and Conditions Where EXPAND Fails to Prevent Cycling," *Mathematical Programming,* Series B, **100**(1): 135–150, May 2004.

[11]See R. Bland: "New Finite Pivoting Rules for the Simplex Method," *Mathematics of Operations Research,* **2:** 103–107, 1977.

TABLE 4.9 Initial simplex tableau for the Wyndor Glass Co. problem without the last two functional constraints

Basic Variable	Eq.	Coefficient of: Z	x_1	x_2	x_3	Right Side	Ratio
Z	(0)	1	-3	-5	0	0	
x_3	(1)	0	1	$\boxed{0}$	1	4	None

With $x_1 = 0$ and x_2 increasing, $x_3 = 4 - 1x_1 - 0x_2 = 4 > 0$.

No Leaving Basic Variable—Unbounded Z

In step 2 of an iteration, there is one other possible outcome that we have not yet discussed, namely, that *no* variable qualifies to be the leaving basic variable.[12] This outcome would occur if the entering basic variable could be increased *indefinitely* without giving negative values to *any* of the current basic variables. In tabular form, this means that *every* coefficient in the pivot column (excluding row 0) is either negative or zero.

As illustrated in Table 4.9, this situation arises in the example displayed in Fig. 3.6. In this example, the last two functional constraints of the Wyndor Glass Co. problem have been overlooked and so are not included in the model. Note in Fig. 3.6 how x_2 can be increased indefinitely (thereby increasing Z indefinitely) without ever leaving the feasible region. Then note in Table 4.9 that x_2 is the entering basic variable but the only coefficient in the pivot column is zero. Because the minimum ratio test uses only coefficients that are greater than zero, there is no ratio to provide a leaving basic variable.

The interpretation of a tableau like the one shown in Table 4.9 is that the constraints do not prevent the value of the objective function Z from increasing indefinitely, so the simplex method would stop with the message that Z is *unbounded.* Because even linear programming has not discovered a way of making infinite profits, the real message for practical problems is that a mistake has been made! The model probably has been misformulated, either by omitting relevant constraints or by stating them incorrectly. Alternatively, a computational mistake may have occurred.

Multiple Optimal Solutions

We mentioned in Sec. 3.2 (under the definition of **optimal solution**) that a problem can have more than one optimal solution. This fact was illustrated in Fig. 3.5 by changing the objective function in the Wyndor Glass Co. problem to $Z = 3x_1 + 2x_2$, so that every point on the line segment between (2, 6) and (4, 3) is optimal. Thus, all optimal solutions are a *weighted average* of these two optimal CPF solutions

$$(x_1, x_2) = w_1(2, 6) + w_2(4, 3),$$

where the weights w_1 and w_2 are numbers that satisfy the relationships

$$w_1 + w_2 = 1 \qquad \text{and} \qquad w_1 \geq 0, \qquad w_2 \geq 0.$$

For example, $w_1 = \frac{1}{3}$ and $w_2 = \frac{2}{3}$ give

[12]Note that the analogous case (no *entering* basic variable) cannot occur in step 1 of an iteration, because the optimality test would stop the algorithm first by indicating that an optimal solution had been reached.

$$(x_1, x_2) = \frac{1}{3}(2, 6) + \frac{2}{3}(4, 3) = \left(\frac{2}{3} + \frac{8}{3},\ \frac{6}{3} + \frac{6}{3}\right) = \left(\frac{10}{3},\ 4\right)$$

as one optimal solution.

In general, any weighted average of two or more solutions (vectors) where the weights are nonnegative and sum to 1 is called a **convex combination** of these solutions. Thus, every optimal solution in the example is a convex combination of (2, 6) and (4, 3).

This example is typical of problems with multiple optimal solutions.

> As indicated at the end of Sec. 3.2, *any* linear programming problem with multiple optimal solutions (and a bounded feasible region) has at least two CPF solutions that are optimal. *Every* optimal solution is a convex combination of these optimal CPF solutions. Consequently, in augmented form, every optimal solution is a convex combination of the optimal BF solutions.

(Problems 4.5-5 and 4.5-6 guide you through the reasoning behind this conclusion.)

The simplex method automatically stops after *one* optimal BF solution is found. However, for many applications of linear programming, there are intangible factors not incorporated into the model that can be used to make meaningful choices between alternative optimal solutions. In such cases, these other optimal solutions should be identified as well. As indicated above, this requires finding all the other optimal BF solutions, and then every optimal solution is a convex combination of the optimal BF solutions.

After the simplex method finds one optimal BF solution, you can detect if there are any others and, if so, find them as follows:

> Whenever a problem has more than one optimal BF solution, at least one of the nonbasic variables has a coefficient of zero in the final row 0, so increasing any such variable will not change the value of Z. Therefore, these other optimal BF solutions can be identified (if desired) by performing additional iterations of the simplex method, each time choosing a nonbasic variable with a zero coefficient as the entering basic variable.[13]

To illustrate, consider again the case just mentioned, where the objective function in the Wyndor Glass Co. problem is changed to $Z = 3x_1 + 2x_2$. The simplex method obtains the first three tableaux shown in Table 4.10 and stops with an optimal BF solution. However, because a nonbasic variable (x_3) then has a zero coefficient in row 0, we perform one more iteration in Table 4.10 to identify the other optimal BF solution. Thus, the two optimal BF solutions are (4, 3, 0, 6, 0) and (2, 6, 2, 0, 0), each yielding $Z = 18$. Notice that the last tableau also has a *nonbasic* variable (x_4) with a zero coefficient in row 0. This situation is inevitable because the extra iteration does not change row 0, so this leaving basic variable necessarily retains its zero coefficient. Making x_4 an entering basic variable now would only lead back to the third tableau. (Check this.) Therefore, these two are the only BF solutions that are optimal, and all *other* optimal solutions are a convex combination of these two.

$$(x_1, x_2, x_3, x_4, x_5) = w_1(2, 6, 2, 0, 0) + w_2(4, 3, 0, 6, 0),$$
$$w_1 + w_2 = 1, \qquad w_1 \geq 0, \qquad w_2 \geq 0.$$

[13]If such an iteration has no *leaving* basic variable, this indicates that the feasible region is unbounded and the entering basic variable can be increased indefinitely without changing the value of Z.

■ **TABLE 4.10** Complete set of simplex tableaux to obtain all optimal BF solutions for the Wyndor Glass Co. problem with $c_2 = 2$

Iteration	Basic Variable	Eq.	Z	x_1	x_2	x_3	x_4	x_5	Right Side	Solution Optimal?
0	Z	(0)	1	−3	−2	0	0	0	0	No
	x_3	(1)	0	1	0	1	0	0	4	
	x_4	(2)	0	0	2	0	1	0	12	
	x_5	(3)	0	3	2	0	0	1	18	
1	Z	(0)	1	0	−2	3	0	0	12	No
	x_1	(1)	0	1	0	1	0	0	4	
	x_4	(2)	0	0	2	0	1	0	12	
	x_5	(3)	0	0	2	−3	0	1	6	
2	Z	(0)	1	0	0	**0**	0	1	18	Yes
	x_1	(1)	0	1	0	1	0	0	4	
	x_4	(2)	0	0	0	3	1	−1	6	
	x_2	(3)	0	0	1	$-\frac{3}{2}$	0	$\frac{1}{2}$	3	
Extra	Z	(0)	1	0	0	0	**0**	1	18	Yes
	x_1	(1)	0	1	0	0	$-\frac{1}{3}$	$\frac{1}{3}$	2	
	x_3	(2)	0	0	0	1	$\frac{1}{3}$	$-\frac{1}{3}$	2	
	x_2	(3)	0	0	1	0	$\frac{1}{2}$	0	6	

4.6 ADAPTING TO OTHER MODEL FORMS

Thus far we have presented the details of the simplex method under the assumptions that the problem is in our standard form (maximize Z subject to functional constraints in $\leq$ form and nonnegativity constraints on all variables) and that $b_i \geq 0$ for all $i = 1, 2, \ldots, m$. In this section we point out how to make the adjustments required for other legitimate forms of the linear programming model. You will see that all these adjustments can be made during the initialization, so the rest of the simplex method can then be applied just as you have learned it already.

The only serious problem introduced by the other forms for functional constraints (the $=$ or $\geq$ forms, or having a negative right-hand side) lies in identifying an *initial BF solution.* Before, this initial solution was found very conveniently by letting the slack variables be the initial basic variables, so that each one just equals the *nonnegative* right-hand side of its equation. Now, something else must be done. The standard approach that is used for all these cases is the **artificial-variable technique.** This technique constructs a more convenient *artificial problem* by introducing a dummy variable (called an *artificial variable*) into each constraint that needs one. This new variable is introduced just for the purpose of being the initial basic variable for that equation. The usual nonnegativity constraints are placed on these variables, and the objective function also is modified to impose an exorbitant penalty on their having values larger than zero. The iterations of the simplex method then automatically force the artificial variables to disappear (become zero), one at a time, until they are all gone, after which the *real* problem is solved.

To illustrate the artificial-variable technique, first we consider the case where the only nonstandard form in the problem is the presence of one or more equality constraints.

Equality Constraints

Any equality constraint

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n = b_i$$

actually is equivalent to a pair of inequality constraints:

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \le b_i$$
$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \ge b_i.$$

However, rather than making this substitution and thereby increasing the number of constraints, it is more convenient to use the artificial-variable technique. We shall illustrate this technique with the following example.

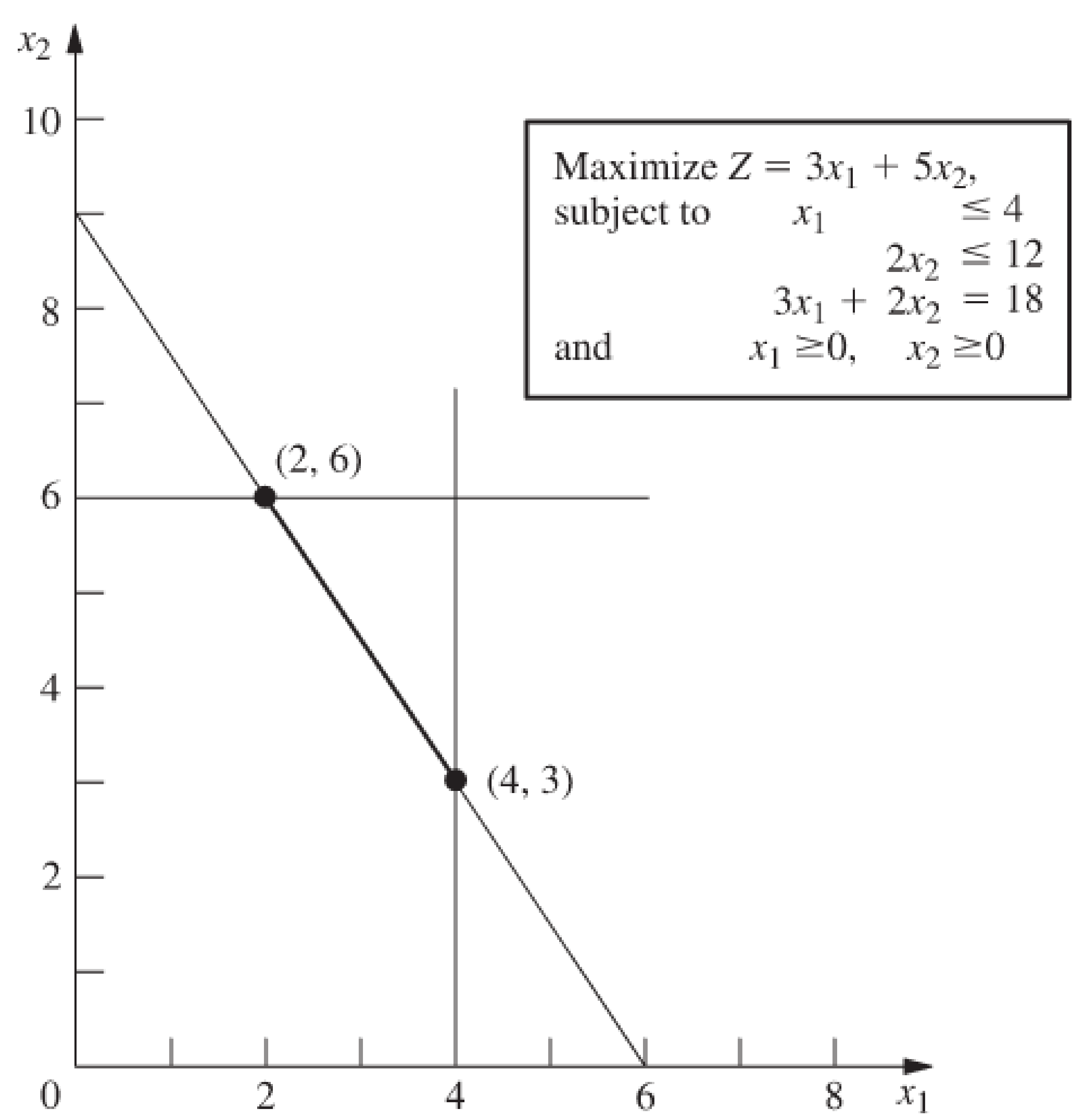
Example. Suppose that the Wyndor Glass Co. problem in Sec. 3.1 is modified to *require* that Plant 3 be used at full capacity. The only resulting change in the linear programming model is that the third constraint, $3x_1 + 2x_2 \le 18$, instead becomes an equality constraint

$$3x_1 + 2x_2 = 18,$$

so that the complete model becomes the one shown in the upper right-hand corner of Fig. 4.3. This figure also shows in darker ink the feasible region which now consists of *just* the line segment connecting (2, 6) and (4, 3).

After the slack variables still needed for the inequality constraints are introduced, the system of equations for the augmented form of the problem becomes

FIGURE 4.3
When the third functional constraint becomes an equality constraint, the feasible region for the Wyndor Glass Co. problem becomes the line segment between (2, 6) and (4, 3).



$$\begin{array}{llr}
(0) & Z - 3x_1 - 5x_2 & = 0 \\
(1) & x_1 \qquad + x_3 & = 4 \\
(2) & 2x_2 \qquad + x_4 & = 12 \\
(3) & 3x_1 + 2x_2 & = 18.
\end{array}$$

Unfortunately, these equations do not have an obvious initial BF solution because there is no longer a slack variable to use as the initial basic variable for Eq. (3). It is necessary to find an initial BF solution to start the simplex method.

This difficulty can be circumvented in the following way.

Obtaining an Initial BF Solution. The procedure is to construct an **artificial problem** that has the same optimal solution as the real problem by making two modifications of the real problem.

1. Apply the **artificial-variable technique** by introducing a *nonnegative* **artificial variable** (call it $\bar{x}_5$)[14] into Eq. (3), just as if it were a slack variable

   (3) $\quad 3x_1 + 2x_2 + \bar{x}_5 = 18.$

2. Assign an *overwhelming penalty* to having $\bar{x}_5 > 0$ by changing the objective function $Z = 3x_1 + 5x_2$ to

   $$Z = 3x_1 + 5x_2 - M\bar{x}_5,$$

   where M symbolically represents a *huge* positive number. (This method of forcing $\bar{x}_5$ to be $\bar{x}_5 = 0$ in the optimal solution is called the **Big *M* method.**)

Now find the optimal solution for the real problem by applying the simplex method to the artificial problem, starting with the following initial BF solution:

Initial BF Solution

Nonbasic variables: $\quad x_1 = 0, \quad x_2 = 0$

Basic variables: $\quad x_3 = 4, \quad x_4 = 12, \quad \bar{x}_5 = 18.$

Because $\bar{x}_5$ plays the role of the slack variable for the third constraint in the artificial problem, this constraint is equivalent to $3x_1 + 2x_2 \le 18$ (just as for the original Wyndor Glass Co. problem in Sec. 3.1). We show below the resulting artificial problem (before augmenting) next to the real problem.

The Real Problem

Maximize $\quad Z = 3x_1 + 5x_2,$

subject to

$$\begin{aligned}
x_1 \qquad &\le 4 \\
2x_2 &\le 12 \\
3x_1 + 2x_2 &= 18
\end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

The Artificial Problem

Define $\quad \bar{x}_5 = 18 - 3x_1 - 2x_2.$

Maximize $\quad Z = 3x_1 + 5x_2 - M\bar{x}_5,$

subject to

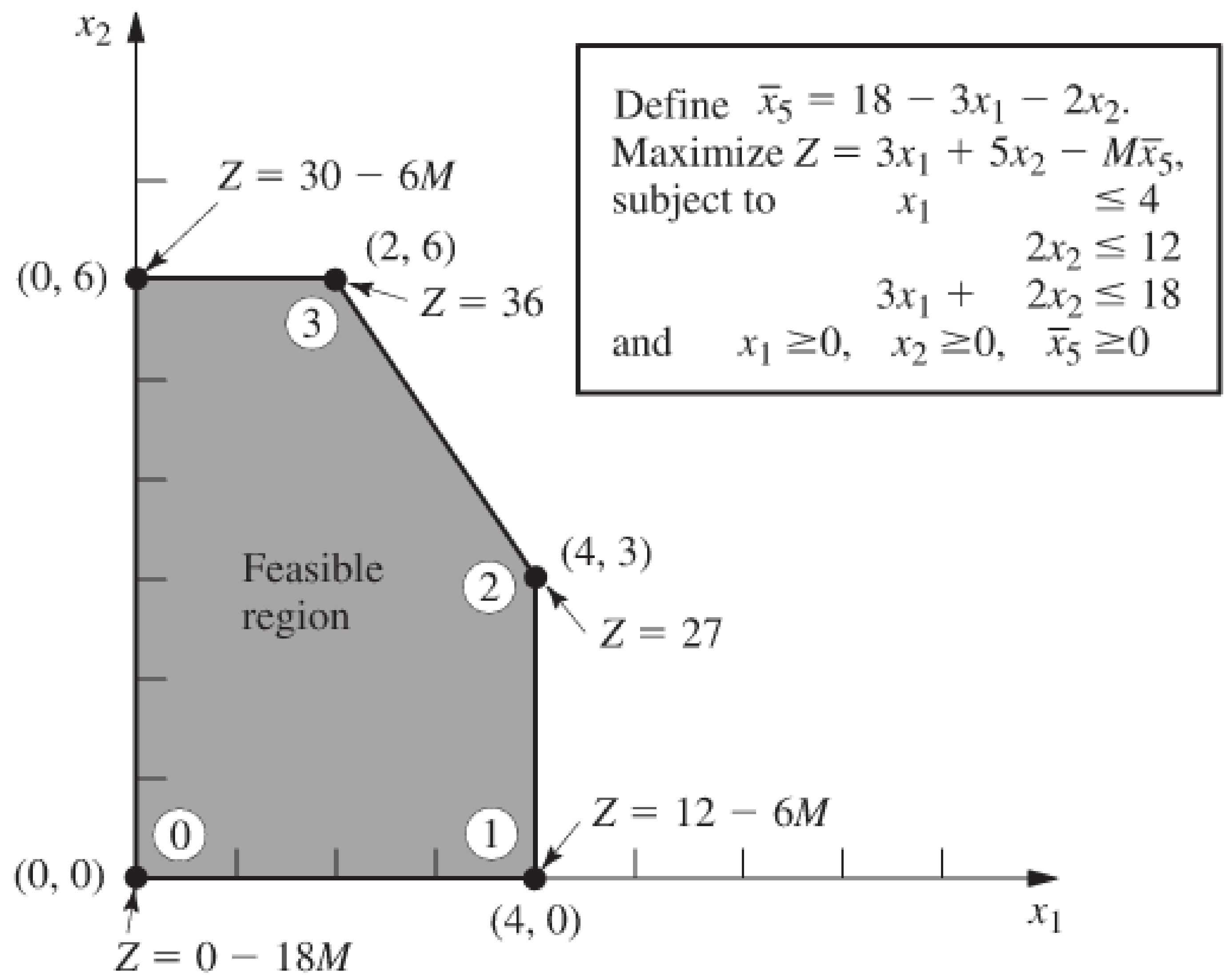
$$\begin{aligned}
x_1 \qquad &\le 4 \\
2x_2 &\le 12 \\
3x_1 + 2x_2 &\le 18 \\
(\text{so} \quad 3x_1 + 2x_2 + \bar{x}_5 &= 18)
\end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad \bar{x}_5 \ge 0.$$

Therefore, just as in Sec. 3.1, the feasible region for (x_1, x_2) for the artificial problem is the one shown in Fig. 4.4. The only portion of this feasible region that coincides with the feasible region for the real problem is where $\bar{x}_5 = 0$ (so $3x_1 + 2x_2 = 18$).

[14]We shall always label the artificial variables by putting a bar over them.



■ **FIGURE 4.4**
This graph shows the feasible region and the sequence of CPF solutions (0, 1, 2, 3) examined by the simplex method for the artificial problem that corresponds to the real problem of Fig. 4.3.

Figure 4.4 also shows the order in which the simplex method examines the CPF solutions (or BF solutions after augmenting), where each circled number identifies which iteration obtained that solution. Note that the simplex method moves counterclockwise here whereas it moved clockwise for the original Wyndor Glass Co. problem (see Fig. 4.2). The reason for this difference is the extra term $-M\bar{x}_5$ in the objective function for the artificial problem.

Before applying the simplex method and demonstrating that it follows the path shown in Fig. 4.4, the following preparatory step is needed.

Converting Equation (0) to Proper Form. The system of equations after the artificial problem is augmented is

$$
\begin{array}{llr}
(0) & Z - 3x_1 - 5x_2 \qquad\qquad + M\bar{\boldsymbol{x}}_5 & = 0 \\
(1) & \qquad x_1 \qquad + \boldsymbol{x}_3 & = 4 \\
(2) & \qquad\qquad 2x_2 \qquad + \boldsymbol{x}_4 & = 12 \\
(3) & \qquad 3x_1 + 2x_2 \qquad\qquad + \bar{\boldsymbol{x}}_5 & = 18
\end{array}
$$

where the initial basic variables (x_3, x_4, $\bar{x}_5$) are shown in bold type. However, this system is not yet in proper form from Gaussian elimination because a basic variable $\bar{x}_5$ has a nonzero coefficient in Eq. (0). Recall that all basic variables must be algebraically eliminated from Eq. (0) before the simplex method can either apply the optimality test or find the entering basic variable. This elimination is necessary so that the negative of the coefficient of each nonbasic variable will give the rate at which Z would increase if that nonbasic variable were to be increased from 0 while adjusting the values of the basic variables accordingly.

To algebraically eliminate $\bar{x}_5$ from Eq. (0), we need to subtract from Eq. (0) the product, M times Eq. (3).

$$
\begin{array}{lr}
 & Z - 3x_1 - 5x_2 + M\bar{x}_5 = \ \ 0 \\
 & -M(3x_1 + 2x_2 + \ \ \bar{x}_5 = 18) \\
\hline
\text{New (0)} & Z - (3M + 3)x_1 - (2M + 5)x_2 = -18M.
\end{array}
$$

Application of the Simplex Method. This new Eq. (0) gives Z in terms of *just* the nonbasic variables (x_1, x_2),

$$Z = -18M + (3M + 3)x_1 + (2M + 5)x_2.$$

Since $3M + 3 > 2M + 5$ (remember that M represents a huge number), increasing x_1 increases Z at a faster rate than increasing x_2 does, so x_1 is chosen as the entering basic variable. This leads to the move from (0, 0) to (4, 0) at iteration 1, shown in Fig. 4.4, thereby increasing Z by $4(3M + 3)$.

The quantities involving M never appear in the system of equations except for Eq. (0), so they need to be taken into account only in the optimality test and when an entering basic variable is determined. One way of dealing with these quantities is to assign some particular (huge) numerical value to M and use the resulting coefficients in Eq. (0) in the usual way. However, this approach may result in significant rounding errors that invalidate the optimality test. Therefore, it is better to do what we have just shown, namely, to express each coefficient in Eq. (0) as a linear function $aM + b$ of the *symbolic* quantity M by separately recording and updating the current numerical value of (1) the *multiplicative* factor a and (2) the *additive* term b. Because M is assumed to be so large that b always is negligible compared with M when $a \neq 0$, the decisions in the optimality test and the choice of the entering basic variable are made by using just the *multiplicative* factors in the usual way, except for breaking ties with the *additive* factors.

Using this approach on the example yields the simplex tableaux shown in Table 4.11. Note that the artificial variable $\bar{x}_5$ is a *basic variable* ($\bar{x}_5 > 0$) in the first two tableaux

TABLE 4.11 Complete set of simplex tableaux for the problem shown in Fig. 4.4

Iteration	Basic Variable	Eq.	Z	x_1	x_2	x_3	x_4	$\bar{x}_5$	Right Side
0	Z	(0)	1	$-3M-3$	$-2M-5$	0	0	0	$-18M$
	x_3	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	2	0	1	0	12
	$\bar{x}_5$	(3)	0	3	2	0	0	1	18
1	Z	(0)	1	0	$-2M-5$	$3M+3$	0	0	$-6M+12$
	x_1	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	2	0	1	0	12
	$\bar{x}_5$	(3)	0	0	2	-3	0	1	6
2	Z	(0)	1	0	0	$-\frac{9}{2}$	0	$M+\frac{5}{2}$	27
	x_1	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	0	3	1	-1	6
	x_2	(3)	0	0	1	$-\frac{3}{2}$	0	$\frac{1}{2}$	3
3	Z	(0)	1	0	0	0	$\frac{3}{2}$	$M+1$	36
	x_1	(1)	0	1	0	0	$-\frac{1}{3}$	$\frac{1}{3}$	2
	x_3	(2)	0	0	0	1	$\frac{1}{3}$	$-\frac{1}{3}$	2
	x_2	(3)	0	0	1	0	$\frac{1}{2}$	0	6

and a *nonbasic variable* ($\bar{x}_5 = 0$) in the last two. Therefore, the first two BF solutions for this artificial problem are *infeasible* for the real problem whereas the last two also are BF solutions for the real problem.

This example involved only one equality constraint. If a linear programming model has more than one, each is handled in just the same way. (If the right-hand side is negative, multiply through both sides by -1 first.)

Negative Right-Hand Sides

The technique mentioned in the preceding sentence for dealing with an equality constraint with a negative right-hand side (namely, multiply through both sides by -1) also works for any inequality constraint with a negative right-hand side. Multiplying through both sides of an inequality by -1 also reverses the direction of the inequality; i.e., $\leq$ changes to $\geq$ or vice versa. For example, doing this to the constraint

$$x_1 - x_2 \leq -1 \qquad (\text{that is, } x_1 \leq x_2 - 1)$$

gives the equivalent constraint

$$-x_1 + x_2 \geq 1 \qquad (\text{that is, } x_2 - 1 \geq x_1)$$

but now the right-hand side is positive. Having nonnegative right-hand sides for all the functional constraints enables the simplex method to begin, because (after augmenting) these right-hand sides become the respective values of the *initial basic variables,* which must satisfy nonnegativity constraints.

We next focus on how to augment $\geq$ constraints, such as $-x_1 + x_2 \geq 1$, with the help of the artificial-variable technique.

Functional Constraints in $\geq$ Form

To illustrate how the artificial-variable technique deals with functional constraints in $\geq$ form, we will use the model for designing Mary's radiation therapy, as presented in Sec. 3.4. For your convenience, this model is repeated below, where we have placed a box around the constraint of special interest here.

Radiation Therapy Example

Minimize $\quad Z = 0.4x_1 + 0.5x_2,$

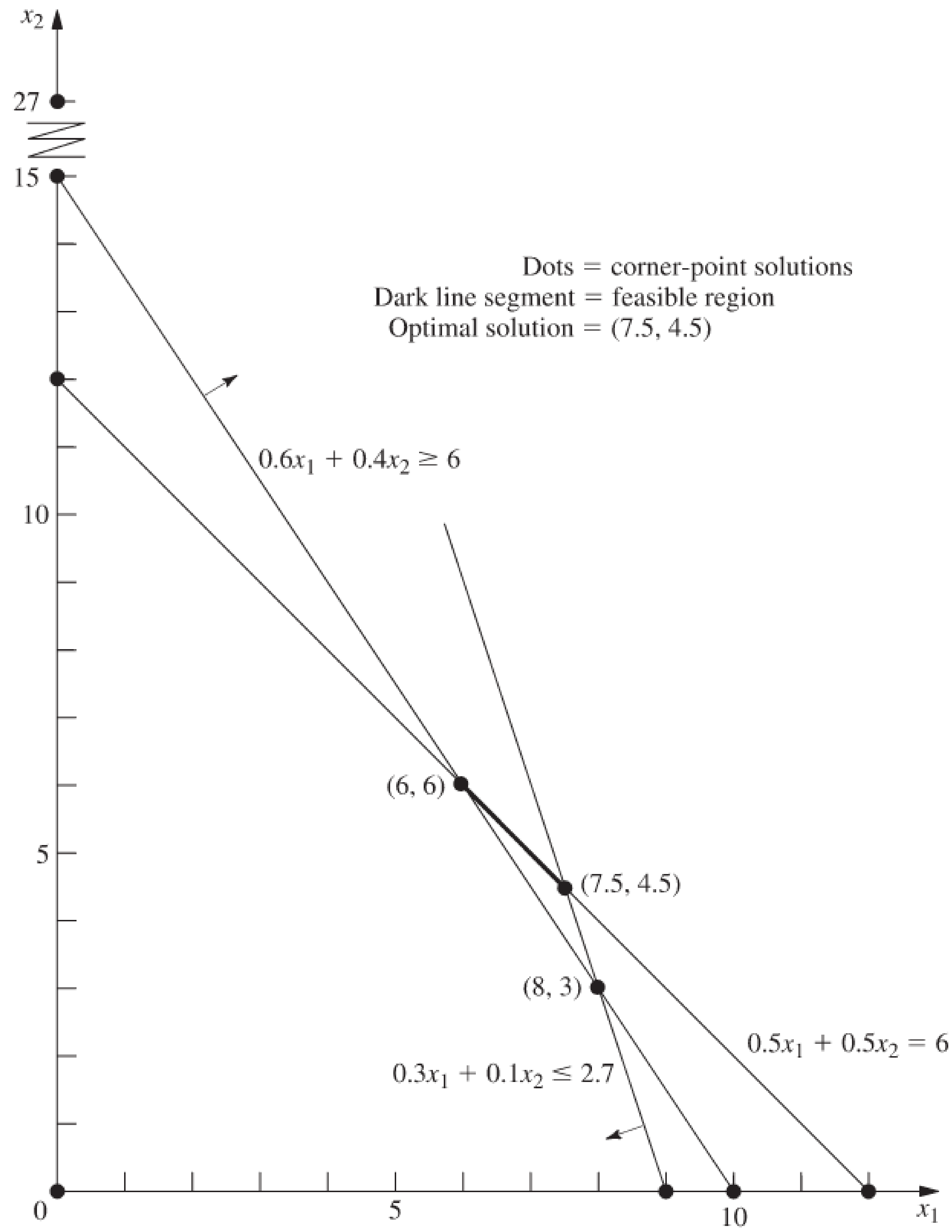
subject to

$$0.3x_1 + 0.1x_2 \leq 2.7$$
$$0.5x_1 + 0.5x_2 = 6$$
$$\boxed{0.6x_1 + 0.4x_2 \geq 6}$$

and

$$x_1 \geq 0, \qquad x_2 \geq 0.$$

The graphical solution for this example (originally presented in Fig. 3.12) is repeated here in a slightly different form in Fig. 4.5. The three lines in the figure, along with the two axes, constitute the five constraint boundaries of the problem. The dots lying at the intersection of a pair of constraint boundaries are the *corner-point solutions.* The only two



■ **FIGURE 4.5**
Graphical display of the radiation therapy example and its corner-point solutions.

corner-point *feasible* solutions are (6, 6) and (7.5, 4.5), and the feasible region is the line segment connecting these two points. The optimal solution is $(x_1, x_2) = (7.5, 4.5)$, with $Z = 5.25$.

We soon will show how the simplex method solves this problem by directly solving the corresponding artificial problem. However, first we must describe how to deal with the third constraint.

Our approach involves introducing *both* a surplus variable x_5 (defined as $x_5 = 0.6x_1 + 0.4x_2 - 6$) and an artificial variable $\bar{x}_6$, as shown next.

$$
\begin{aligned}
& 0.6x_1 + 0.4x_2 && \geq 6 \\
\rightarrow\quad & 0.6x_1 + 0.4x_2 - x_5 && = 6 \qquad (x_5 \geq 0) \\
\rightarrow\quad & 0.6x_1 + 0.4x_2 - x_5 + \bar{x}_6 && = 6 \qquad (x_5 \geq 0,\ \bar{x}_6 \geq 0).
\end{aligned}
$$

Here x_5 is called a **surplus variable** because it subtracts the surplus of the left-hand side over the right-hand side to convert the inequality constraint to an equivalent equality constraint. Once this conversion is accomplished, the artificial variable is introduced just as for any equality constraint.

After a slack variable x_3 is introduced into the first constraint, an artificial variable $\bar{x}_4$ is introduced into the second constraint, and the Big M method is applied, so the complete artificial problem (in augmented form) is

$$\text{Minimize} \quad Z = 0.4x_1 + 0.5x_2 + M\bar{x}_4 + M\bar{x}_6,$$

$$\begin{aligned}\text{subject to} \quad & 0.3x_1 + 0.1x_2 + x_3 && = 2.7 \\ & 0.5x_1 + 0.5x_2 \qquad + \bar{x}_4 && = 6 \\ & 0.6x_1 + 0.4x_2 \qquad\qquad - x_5 + \bar{x}_6 && = 6\end{aligned}$$

$$\text{and} \quad x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0, \quad \bar{x}_4 \ge 0, \quad x_5 \ge 0, \quad \bar{x}_6 \ge 0.$$

Note that the coefficients of the artificial variables in the objective function are $+M$, instead of $-M$, because we now are minimizing Z. Thus, even though $\bar{x}_4 > 0$ and/or $\bar{x}_6 > 0$ is possible for a feasible solution for the artificial problem, the huge unit penalty of $+M$ prevents this from occurring in an optimal solution.

As usual, introducing artificial variables enlarges the feasible region. Compare below the original constraints for the real problem with the corresponding constraints on (x_1, x_2) for the artificial problem.

Constraints on (x_1, x_2) *for the Real Problem*	*Constraints on* (x_1, x_2) *for the Artificial Problem*
$0.3x_1 + 0.1x_2 \le 2.7$	$0.3x_1 + 0.1x_2 \le 2.7$
$0.5x_1 + 0.5x_2 = 6$	$0.5x_1 + 0.5x_2 \le 6$ (= holds when $\bar{x}_4 = 0$)
$0.6x_1 + 0.4x_2 \ge 6$	No such constraint (except when $\bar{x}_6 = 0$)
$x_1 \ge 0, \quad x_2 \ge 0$	$x_1 \ge 0, \quad x_2 \ge 0$

Introducing the artificial variable $\bar{x}_4$ to play the role of a slack variable in the second constraint allows values of (x_1, x_2) *below* the $0.5x_1 + 0.5x_2 = 6$ line in Fig. 4.5. Introducing x_5 and $\bar{x}_6$ into the third constraint of the real problem (and moving these variables to the right-hand side) yields the equation

$$0.6x_1 + 0.4x_2 = 6 + x_5 - \bar{x}_6.$$

Because both x_5 and $\bar{x}_6$ are constrained only to be nonnegative, their difference $x_5 - \bar{x}_6$ can be any positive or negative number. Therefore, $0.6x_1 + 0.4x_2$ can have any value, which has the effect of eliminating the third constraint from the artificial problem and allowing points on either side of the $0.6x_1 + 0.4x_2 = 6$ line in Fig. 4.5. (We keep the third constraint in the system of equations only because it will become relevant again later, after the Big M method forces $\bar{x}_6$ to be zero.) Consequently, the feasible region for the artificial problem is the entire polyhedron in Fig. 4.5 whose vertices are (0, 0), (9, 0), (7.5, 4.5), and (0, 12).

Since the origin now is feasible for the artificial problem, the simplex method starts with (0, 0) as the initial CPF solution, i.e., with $(x_1, x_2, x_3, \bar{x}_4, x_5, \bar{x}_6) = (0, 0, 2.7, 6, 0, 6)$ as the initial BF solution. (Making the origin feasible as a convenient starting point for the simplex method is the whole point of creating the artificial problem.) We soon will trace the entire path followed by the simplex method from the origin to the optimal solution for both the artificial and real problems. But, first, how does the simplex method handle *minimization?*

Minimization

One straightforward way of minimizing Z with the simplex method is to exchange the roles of the positive and negative coefficients in row 0 for both the optimality test and step 1 of an iteration. However, rather than changing our instructions for the simplex method for this case, we present the following simple way of converting any minimization problem to an equivalent maximization problem:

$$\textit{Minimizing} \qquad Z = \sum_{j=1}^{n} c_j x_j$$

is equivalent to

$$\textit{maximizing} \qquad -Z = \sum_{j=1}^{n} (-c_j) x_j;$$

i.e., the two formulations yield the same optimal solution(s).

The two formulations are equivalent because the smaller Z is, the larger $-Z$ is, so the solution that gives the *smallest* value of Z in the entire feasible region must also give the *largest* value of $-Z$ in this region.

Therefore, in the radiation therapy example, we make the following change in the formulation:

$$\begin{array}{lll} & \text{Minimize} & Z = 0.4x_1 + 0.5x_2 \\ \rightarrow & \text{Maximize} & -Z = -0.4x_1 - 0.5x_2. \end{array}$$

After artificial variables $\bar{x}_4$ and $\bar{x}_6$ are introduced and then the Big M method is applied, the corresponding conversion is

$$\begin{array}{lll} & \text{Minimize} & Z = 0.4x_1 + 0.5x_2 + M\bar{x}_4 + M\bar{x}_6 \\ \rightarrow & \text{Maximize} & -Z = -0.4x_1 - 0.5x_2 - M\bar{x}_4 - M\bar{x}_6. \end{array}$$

Solving the Radiation Therapy Example

We now are nearly ready to apply the simplex method to the radiation therapy example. By using the maximization form just obtained, the entire system of equations is now

$$\begin{array}{lllllll} (0) & -Z + 0.4x_1 + 0.5x_2 & & + M\bar{\mathbf{x}}_4 & & + M\bar{\mathbf{x}}_6 & = 0 \\ (1) & 0.3x_1 + 0.1x_2 & + \mathbf{x}_3 & & & & = 2.7 \\ (2) & 0.5x_1 + 0.5x_2 & & + \bar{\mathbf{x}}_4 & & & = 6 \\ (3) & 0.6x_1 + 0.4x_2 & & & - x_5 & + \bar{\mathbf{x}}_6 & = 6. \end{array}$$

The basic variables (x_3, $\bar{x}_4$, $\bar{x}_6$) for the initial BF solution (for this artificial problem) are shown in bold type.

Note that this system of equations is not yet in proper form from Gaussian elimination, as required by the simplex method, since the basic variables $\bar{x}_4$ and $\bar{x}_6$ still need to be algebraically eliminated from Eq. (0). Because $\bar{x}_4$ and $\bar{x}_6$ both have a coefficient of M, Eq. (0) needs to have subtracted from it *both* M times Eq. (2) *and* M times Eq. (3). The calculations for all the coefficients (and the right-hand sides) are summarized below, where the vectors are the relevant rows of the simplex tableau corresponding to the above system of equations.

Row 0:

$$\begin{array}{rrrrrrr} [0.4, & 0.5, & 0, & M, & 0, & M, & 0] \\ -M[0.5, & 0.5, & 0, & 1, & 0, & 0, & 6] \\ -M[0.6, & 0.4, & 0, & 0, & -1, & 1, & 6] \\ \hline \text{New row } 0 = [-1.1M + 0.4, & -0.9M + 0.5, & 0, & 0, & M, & 0, & -12M] \end{array}$$

The resulting initial simplex tableau, ready to begin the simplex method, is shown at the top of Table 4.12. Applying the simplex method in just the usual way then yields the

TABLE 4.12 The Big *M* method for the radiation therapy example

Iteration	Basic Variable	Eq.	Z	x_1	x_2	x_3	$\bar{x}_4$	x_5	$\bar{x}_6$	Right Side
0	Z	(0)	-1	$-1.1M + 0.4$	$-0.9M + 0.5$	0	0	M	0	$-12M$
	x_3	(1)	0	0.3	0.1	1	0	0	0	2.7
	$\bar{x}_4$	(2)	0	0.5	0.5	0	1	0	0	6
	$\bar{x}_6$	(3)	0	0.6	0.4	0	0	-1	1	6
1	Z	(0)	-1	0	$-\frac{16}{30}M + \frac{11}{30}$	$\frac{11}{3}M - \frac{4}{3}$	0	M	0	$-2.1M - 3.6$
	x_1	(1)	0	1	$\frac{1}{3}$	$\frac{10}{3}$	0	0	0	9
	$\bar{x}_4$	(2)	0	0	$\frac{1}{3}$	$-\frac{5}{3}$	1	0	0	1.5
	$\bar{x}_6$	(3)	0	0	0.2	-2	0	-1	1	0.6
2	Z	(0)	-1	0	0	$-\frac{5}{3}M + \frac{7}{3}$	0	$-\frac{5}{3}M + \frac{11}{6}$	$\frac{8}{3}M - \frac{11}{6}$	$-0.5M - 4.7$
	x_1	(1)	0	1	0	$\frac{20}{3}$	0	$\frac{5}{3}$	$-\frac{5}{3}$	8
	$\bar{x}_4$	(2)	0	0	0	$\frac{5}{3}$	1	$\frac{5}{3}$	$-\frac{5}{3}$	0.5
	x_2	(3)	0	0	1	-10	0	-5	5	3
3	Z	(0)	-1	0	0	0.5	$M - 1.1$	0	M	-5.25
	x_1	(1)	0	1	0	5	-1	0	0	7.5
	x_5	(2)	0	0	0	1	0.6	1	-1	0.3
	x_2	(3)	0	0	1	-5	3	0	0	4.5

sequence of simplex tableaux shown in the rest of Table 4.12. For the optimality test and the selection of the entering basic variable at each iteration, the quantities involving M are treated just as discussed in connection with Table 4.11. Specifically, whenever M is present, only its multiplicative factor is used, unless there is a tie, in which case the tie is broken by using the corresponding additive terms. Just such a tie occurs in the last selection of an entering basic variable (see the next-to-last tableau), where the coefficients of x_3 and x_5 in row 0 both have the same multiplicative factor of $-\frac{5}{3}$. Comparing the additive terms, $\frac{11}{6} < \frac{7}{3}$ leads to choosing x_5 as the entering basic variable.

Note in Table 4.12 the progression of values of the artificial variables $\bar{x}_4$ and $\bar{x}_6$ and of Z. We start with large values, $\bar{x}_4 = 6$ and $\bar{x}_6 = 6$, with $Z = 12M$ ($-Z = -12M$). The first iteration greatly reduces these values. The Big M method succeeds in driving $\bar{x}_6$ to zero (as a new nonbasic variable) at the second iteration and then in doing the same to $\bar{x}_4$ at the next iteration. With both $\bar{x}_4 = 0$ and $\bar{x}_6 = 0$, the basic solution given in the last tableau is guaranteed to be feasible for the real problem. Since it passes the optimality test, it also is optimal.

Now see what the Big M method has done graphically in Fig. 4.6. The feasible region for the artificial problem initially has four CPF solutions—(0, 0), (9, 0), (0, 12), and (7.5, 4.5)—and then replaces the first three with two new CPF solutions—(8, 3), (6, 6)—after $\bar{x}_6$ decreases to $\bar{x}_6 = 0$ so that $0.6x_1 + 0.4x_2 \geq 6$ becomes an additional constraint. (Note that the three replaced CPF solutions—(0, 0), (9, 0), and (0, 12)—actually were corner-point *infeasible* solutions for the real problem shown in Fig. 4.5.) Starting with the origin as the convenient initial CPF solution for the artificial problem, we move around

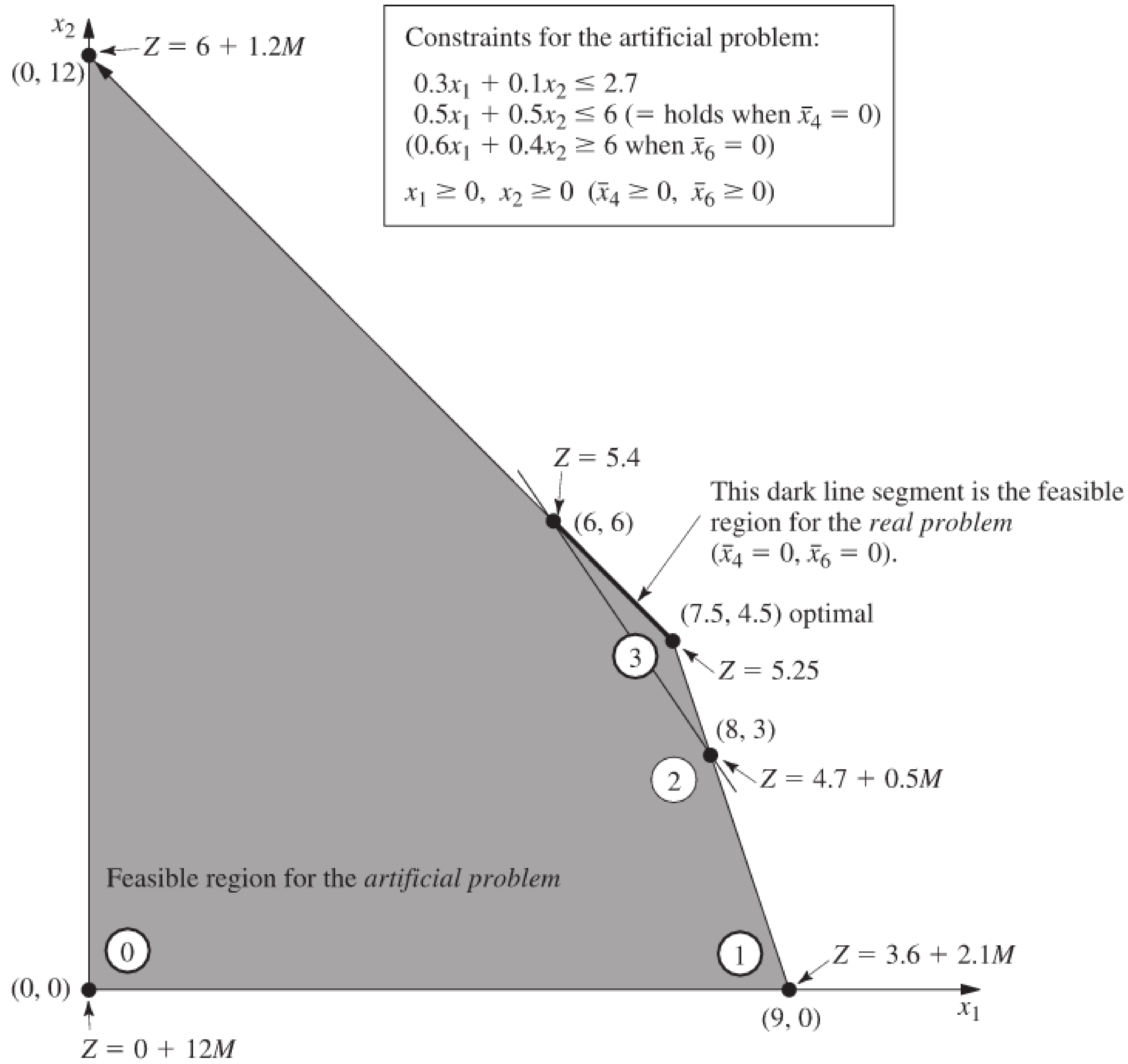


FIGURE 4.6
This graph shows the feasible region and the sequence of CPF solutions (0, 1, 2, 3) examined by the simplex method (with the Big M method) for the artificial problem that corresponds to the real problem of Fig. 4.5.

the boundary to three other CPF solutions—(9, 0), (8, 3), and (7.5, 4.5). The last of these is the first one that also is feasible for the real problem. Fortuitously, this first feasible solution also is optimal, so no additional iterations are needed.

For other problems with artificial variables, it may be necessary to perform additional iterations to reach an optimal solution after the first feasible solution is obtained for the real problem. (This was the case for the example solved in Table 4.11.) Thus, the Big M method can be thought of as having two phases. In the *first phase,* all the artificial variables are driven to zero (because of the penalty of M per unit for being greater than zero) in order to reach an initial BF solution for the *real* problem. In the *second phase,* all the artificial variables are kept at zero (because of this same penalty) while the simplex method generates a sequence of BF solutions for the real problem that leads to an optimal solution. The *two-phase method* described next is a streamlined procedure for performing these two phases directly, without even introducing M explicitly.

The Two-Phase Method

For the radiation therapy example just solved in Table 4.12, recall its real objective function

Real problem: Minimize $Z = 0.4x_1 + 0.5x_2.$

However, the Big M method uses the following objective function (or its equivalent in maximization form) throughout the entire procedure:

Big M method: Minimize $Z = 0.4x_1 + 0.5x_2 + M\bar{x}_4 + M\bar{x}_6.$

Since the first two coefficients are negligible compared to M, the two-phase method is able to drop M by using the following two objective functions with completely different definitions of Z in turn.

Two-phase method:

Phase 1: Minimize $Z = \bar{x}_4 + \bar{x}_6$ (until $\bar{x}_4 = 0$, $\bar{x}_6 = 0$).
Phase 2: Minimize $Z = 0.4x_1 + 0.5x_2$ (with $\bar{x}_4 = 0$, $\bar{x}_6 = 0$).

The phase 1 objective function is obtained by dividing the Big M method objective function by M and then dropping the negligible terms. Since phase 1 concludes by obtaining a BF solution for the real problem (one where $\bar{x}_4 = 0$ and $\bar{x}_6 = 0$), this solution is then used as the *initial* BF solution for applying the simplex method to the real problem (with its real objective function) in phase 2.

Before solving the example in this way, we summarize the general method.

Summary of the Two-Phase Method. *Initialization:* Revise the constraints of the original problem by introducing artificial variables as needed to obtain an obvious initial BF solution for the *artificial problem.*

Phase 1: The objective for this phase is to find a BF solution for the *real problem.* To do this,

Minimize $Z = \Sigma$ artificial variables, subject to revised constraints.

The optimal solution obtained for this problem (with $Z = 0$) will be a BF solution for the real problem.

Phase 2: The objective for this phase is to find an *optimal solution* for the real problem. Since the artificial variables are not part of the real problem, these variables can now be dropped (they are all zero now anyway).[15] Starting from the BF solution obtained at the end of phase 1, use the simplex method to solve the real problem.

For the example, the problems to be solved by the simplex method in the respective phases are summarized below.

Phase 1 Problem (Radiation Therapy Example):

Minimize $Z = \bar{x}_4 + \bar{x}_6,$

subject to

$$\begin{aligned} 0.3x_1 + 0.1x_2 + x_3 \qquad\qquad\qquad &= 2.7 \\ 0.5x_1 + 0.5x_2 \qquad + \bar{x}_4 \qquad\qquad &= 6 \\ 0.6x_1 + 0.4x_2 \qquad\qquad - x_5 + \bar{x}_6 &= 6 \end{aligned}$$

and

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad \bar{x}_4 \geq 0, \quad x_5 \geq 0, \quad \bar{x}_6 \geq 0.$$

Phase 2 Problem (Radiation Therapy Example):

Minimize $Z = 0.4x_1 + 0.5x_2,$

[15]We are skipping over three other possibilities here: (1) artificial variables > 0 (discussed in the next subsection), (2) artificial variables that are degenerate basic variables, and (3) retaining the artificial variables as nonbasic variables in phase 2 (and not allowing them to become basic) as an aid to subsequent postoptimality analysis. Your IOR Tutorial allows you to explore these possibilities.

subject to

$$
\begin{aligned}
0.3x_1 + 0.1x_2 + x_3 \qquad &= 2.7\\
0.5x_1 + 0.5x_2 \qquad\qquad &= 6\\
0.6x_1 + 0.4x_2 \qquad - x_5 &= 6
\end{aligned}
$$

and

$$x_1 \geq 0, \qquad x_2 \geq 0, \qquad x_3 \geq 0, \qquad x_5 \geq 0.$$

The only differences between these two problems are in the objective function and in the inclusion (phase 1) or exclusion (phase 2) of the artificial variables $\bar{x}_4$ and $\bar{x}_6$. Without the artificial variables, the phase 2 problem does not have an obvious *initial BF solution.* The sole purpose of solving the phase 1 problem is to obtain a BF solution with $\bar{x}_4 = 0$ and $\bar{x}_6 = 0$ so that this solution (without the artificial variables) can be used as the initial BF solution for phase 2.

Table 4.13 shows the result of applying the simplex method to this phase 1 problem. [Row 0 in the initial tableau is obtained by converting Minimize $Z = \bar{x}_4 + \bar{x}_6$ to Maximize $(-Z) = -\bar{x}_4 - \bar{x}_6$ and then using *elementary row operations* to eliminate the basic variables $\bar{x}_4$ and $\bar{x}_6$ from $-Z + \bar{x}_4 + \bar{x}_6 = 0$.] In the next-to-last tableau, there is a tie for the *entering basic variable* between x_3 and x_5, which is broken arbitrarily in favor of x_3. The solution obtained at the end of phase 1, then, is $(x_1, x_2, x_3, \bar{x}_4, x_5, \bar{x}_6) = (6, 6, 0.3, 0, 0, 0)$ or, after $\bar{x}_4$ and $\bar{x}_6$ are dropped, $(x_1, x_2, x_3, x_5) = (6, 6, 0.3, 0)$.

TABLE 4.13 Phase 1 of the two-phase method for the radiation therapy example

Iteration	Basic Variable	Eq.	Coefficient of: Z	x_1	x_2	x_3	$\bar{x}_4$	x_5	$\bar{x}_6$	Right Side
0	Z	(0)	-1	-1.1	-0.9	0	0	1	0	-12
	x_3	(1)	0	0.3	0.1	1	0	0	0	2.7
	$\bar{x}_4$	(2)	0	0.5	0.5	0	1	0	0	6
	$\bar{x}_6$	(3)	0	0.6	0.4	0	0	-1	1	6
1	Z	(0)	-1	0	$-\frac{16}{30}$	$\frac{11}{3}$	0	1	0	-2.1
	x_1	(1)	0	1	$\frac{1}{3}$	$\frac{10}{3}$	0	0	0	9
	$\bar{x}_4$	(2)	0	0	$\frac{1}{3}$	$-\frac{5}{3}$	1	0	0	1.5
	$\bar{x}_6$	(3)	0	0	0.2	-2	0	-1	1	0.6
2	Z	(0)	-1	0	0	$-\frac{5}{3}$	0	$-\frac{5}{3}$	$\frac{8}{3}$	-0.5
	x_1	(1)	0	1	0	$\frac{20}{3}$	0	$\frac{5}{3}$	$-\frac{5}{3}$	8
	$\bar{x}_4$	(2)	0	0	0	$\frac{5}{3}$	1	$\frac{5}{3}$	$-\frac{5}{3}$	0.5
	x_2	(3)	0	0	1	-10	0	-5	5	3
3	Z	(0)	-1	0	0	0	1	0	1	0
	x_1	(1)	0	1	0	0	-4	-5	5	6
	x_3	(2)	0	0	0	1	$\frac{3}{5}$	1	-1	0.3
	x_2	(3)	0	0	1	0	6	5	-5	6

As claimed in the summary, this solution from phase 1 is indeed a BF solution for the *real* problem (the phase 2 problem) because it is the solution (after you set $x_5 = 0$) to the system of equations consisting of the three functional constraints for the phase 2 problem. In fact, after deleting the $\bar{x}_4$ and $\bar{x}_6$ columns as well as row 0 for each iteration, Table 4.13 shows one way of using Gaussian elimination to solve this system of equations by reducing the system to the form displayed in the final tableau.

Table 4.14 shows the preparations for beginning phase 2 after phase 1 is completed. Starting from the final tableau in Table 4.13, we drop the artificial variables ($\bar{x}_4$ and $\bar{x}_6$), substitute the phase 2 objective function ($-Z = -0.4x_1 - 0.5x_2$ in maximization form) into row 0, and then restore the proper form from Gaussian elimination (by algebraically eliminating the basic variables x_1 and x_2 from row 0). Thus, row 0 in the last tableau is obtained by performing the following *elementary row operations* in the next-to-last tableau: from row 0 subtract both the product, 0.4 times row 1, and the product, 0.5 times row 3. Except for the deletion of the two columns, note that rows 1 to 3 never change. The only adjustments occur in row 0 in order to replace the phase 1 objective function by the phase 2 objective function.

The last tableau in Table 4.14 is the initial tableau for applying the simplex method to the phase 2 problem, as shown at the top of Table 4.15. Just one iteration then leads to the optimal solution shown in the second tableau: $(x_1, x_2, x_3, x_5) = (7.5, 4.5, 0, 0.3)$. This solution is the desired optimal solution for the real problem of interest rather than the artificial problem constructed for phase 1.

Now we see what the two-phase method has done graphically in Fig. 4.7. Starting at the origin, phase 1 examines a total of four CPF solutions for the artificial problem. The first three actually were corner-point infeasible solutions for the real problem shown in Fig. 4.5. The fourth CPF solution, at (6, 6), is the first one that also is feasible for the real problem, so it becomes the initial CPF solution for phase 2. One iteration in phase 2 leads to the optimal CPF solution at (7.5, 4.5).

TABLE 4.14 Preparing to begin phase 2 for the radiation therapy example

	Basic Variable	Eq.	Coefficient of:							Right Side
			Z	x_1	x_2	x_3	$\bar{x}_4$	x_5	$\bar{x}_6$	
Final Phase 1 tableau	Z	(0)	−1	0	0	0	1	0	1	0
	x_1	(1)	0	1	0	0	−4	−5	5	6
	x_3	(2)	0	0	0	1	$\frac{3}{5}$	1	−1	0.3
	x_2	(3)	0	0	1	0	6	5	−5	6
Drop $\bar{x}_4$ and $\bar{x}_6$	Z	(0)	−1	0	0	0		0		0
	x_1	(1)	0	1	0	0		−5		6
	x_3	(2)	0	0	0	1		1		0.3
	x_2	(3)	0	0	1	0		5		6
Substitute phase 2 objective function	Z	(0)	−1	0.4	0.5	0		0		0
	x_1	(1)	0	1	0	0		−5		6
	x_3	(2)	0	0	0	1		1		0.3
	x_2	(3)	0	0	1	0		5		6
Restore proper form from Gaussian elimination	Z	(0)	−1	0	0	0		−0.5		−5.4
	x_1	(1)	0	1	0	0		−5		6
	x_3	(2)	0	0	0	1		1		0.3
	x_2	(3)	0	0	1	0		5		6

TABLE 4.15 Phase 2 of the two-phase method for the radiation therapy example

Iteration	Basic Variable	Eq.	Coefficient of: Z	x_1	x_2	x_3	x_5	Right Side
0	Z	(0)	−1	0	0	0	−0.5	−5.4
	x_1	(1)	0	1	0	0	−5	6
	x_3	(2)	0	0	0	1	1	0.3
	x_2	(3)	0	0	1	0	5	6
1	Z	(0)	−1	0	0	0.5	0	−5.25
	x_1	(1)	0	1	0	5	0	7.5
	x_5	(2)	0	0	0	1	1	0.3
	x_2	(3)	0	0	1	−5	0	4.5

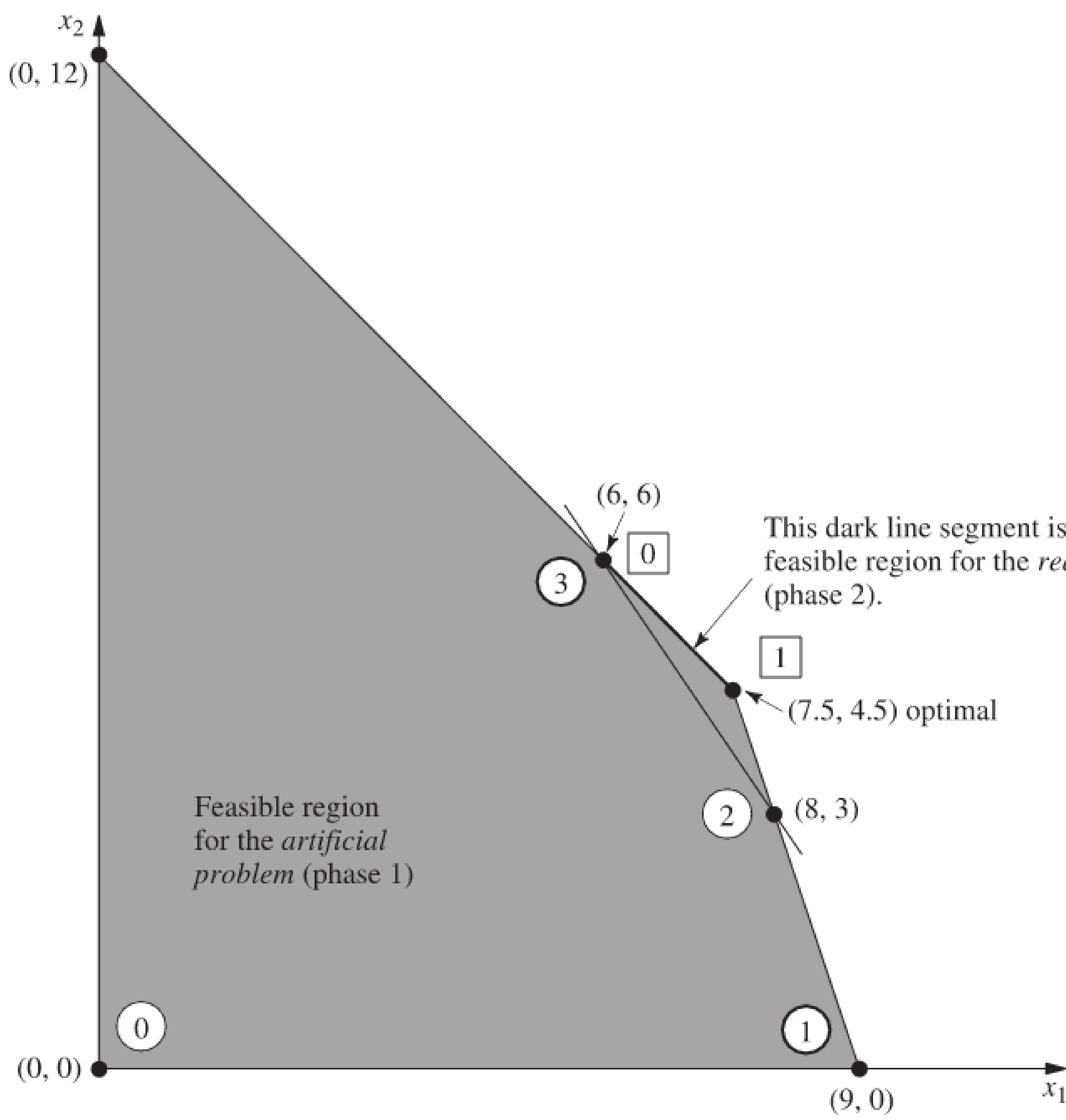


FIGURE 4.7
This graph shows the sequence of CPF solutions for phase 1 (0, 1, 2, 3) and then for phase 2 (0, 1) when the two-phase method is applied to the radiation therapy example.

If the tie for the entering basic variable in the next-to-last tableau of Table 4.13 had been broken in the other way, then phase 1 would have gone directly from (8, 3) to (7.5, 4.5). After (7.5, 4.5) was used to set up the initial simplex tableau for phase 2, the *optimality test* would have revealed that this solution was optimal, so no iterations would be done.

It is interesting to compare the Big M and two-phase methods. Begin with their objective functions.

Big M Method:

Minimize $\quad Z = 0.4x_1 + 0.5x_2 + M\bar{x}_4 + M\bar{x}_6.$

Two-Phase Method:

Phase 1: Minimize $\quad Z = \bar{x}_4 + \bar{x}_6.$
Phase 2: Minimize $\quad Z = 0.4x_1 + 0.5x_2.$

Because the $M\bar{x}_4$ and $M\bar{x}_6$ terms dominate the $0.4x_1$ and $0.5x_2$ terms in the objective function for the Big M method, this objective function is essentially equivalent to the phase 1 objective function as long as $\bar{x}_4$ and/or $\bar{x}_6$ is greater than zero. Then, when both $\bar{x}_4 = 0$ and $\bar{x}_6 = 0$, the objective function for the Big M method becomes completely equivalent to the phase 2 objective function.

Because of these virtual equivalencies in objective functions, the Big M and two-phase methods generally have the same sequence of BF solutions. The one possible exception occurs when there is a tie for the entering basic variable in phase 1 of the two-phase method, as happened in the third tableau of Table 4.13. Notice that the first three tableaux of Tables 4.12 and 4.13 are almost identical, with the only difference being that the multiplicative factors of M in Table 4.12 become the sole quantities in the corresponding spots in Table 4.13. Consequently, the additive terms that broke the tie for the entering basic variable in the third tableau of Table 4.12 were not present to break this same tie in Table 4.13. The result for this example was an extra iteration for the two-phase method. Generally, however, the advantage of having the additive factors is minimal.

The two-phase method streamlines the Big M method by using only the multiplicative factors in phase 1 and by dropping the artificial variables in phase 2. (The Big M method could combine the multiplicative and additive factors by assigning an actual huge number to M, but this might create numerical instability problems.) For these reasons, the two-phase method is commonly used in computer codes.

The Worked Examples section on the book's website provides **another example** of applying both the Big M method and the two-phase method to the same problem.

No Feasible Solutions

So far in this section we have been concerned primarily with the fundamental problem of identifying an initial BF solution when an obvious one is not available. You have seen how the artificial-variable technique can be used to construct an artificial problem and obtain an initial BF solution for this artificial problem instead. Use of either the Big M method or the two-phase method then enables the simplex method to begin its pilgrimage toward the BF solutions, and ultimately toward the optimal solution, for the *real* problem.

However, you should be wary of a certain pitfall with this approach. There may be no obvious choice for the initial BF solution for the very good reason that there are no feasible solutions at all! Nevertheless, by constructing an artificial feasible solution, there is nothing to prevent the simplex method from proceeding as usual and ultimately reporting a supposedly optimal solution.

Fortunately, the artificial-variable technique provides the following signpost to indicate when this has happened:

> If the original problem has *no feasible solutions,* then either the Big M method or phase 1 of the two-phase method yields a final solution that has at least one artificial variable *greater* than zero. Otherwise, they *all* equal zero.

■ **TABLE 4.16** The Big M method for the revision of the radiation therapy example that has no feasible solutions

Iteration	Basic Variable	Eq.	Z	x_1	x_2	x_3	$\bar{x}_4$	x_5	$\bar{x}_6$	Right Side
				Coefficient of:						
0	Z	(0)	-1	$-1.1M + 0.4$	$-0.9M + 0.5$	0	0	M	0	$-12M$
	x_3	(1)	0	0.3	0.1	1	0	0	0	1.8
	$\bar{x}_4$	(2)	0	0.5	0.5	0	1	0	0	6
	$\bar{x}_6$	(3)	0	0.6	0.4	0	0	-1	1	6
1	Z	(0)	-1	0	$-\frac{16}{30}M + \frac{11}{30}$	$\frac{11}{3}M - \frac{4}{3}$	0	M	0	$-5.4M - 2.4$
	x_1	(1)	0	1	$\frac{1}{3}$	$\frac{10}{3}$	0	0	0	6
	$\bar{x}_4$	(2)	0	0	$\frac{1}{3}$	$-\frac{5}{3}$	1	0	0	3
	$\bar{x}_6$	(3)	0	0	0.2	-2	0	-1	1	2.4
2	Z	(0)	-1	0	0	$M + 0.5$	$1.6M - 1.1$	M	0	$-0.6M - 5.7$
	x_1	(1)	0	1	0	5	-1	0	0	3
	x_2	(2)	0	0	1	-5	3	0	0	9
	$\bar{x}_6$	(3)	0	0	0	-1	-0.6	-1	1	0.6

To illustrate, let us change the first constraint in the radiation therapy example (see Fig. 4.5) as follows:

$$0.3x_1 + 0.1x_2 \le 2.7 \qquad \rightarrow \qquad 0.3x_1 + 0.1x_2 \le 1.8,$$

so that the problem no longer has any feasible solutions. Applying the Big M method just as before (see Table 4.12) yields the tableaux shown in Table 4.16. (Phase 1 of the two-phase method yields the same tableaux except that each expression involving M is replaced by just the multiplicative factor.) Hence, the Big M method normally would be indicating that the optimal solution is (3, 9, 0, 0, 0, 0.6). However, since an artificial variable $\bar{x}_6 = 0.6 > 0$, the real message here is that the problem has no feasible solutions.[16]

Variables Allowed to Be Negative

In most practical problems, negative values for the decision variables would have no physical meaning, so it is necessary to include nonnegativity constraints in the formulations of their linear programming models. However, this is not always the case. To illustrate, suppose that the Wyndor Glass Co. problem is changed so that product 1 already is in production, and the first decision variable x_1 represents the *increase* in its production rate. Therefore, a negative value of x_1 would indicate that product 1 is to be cut back by that amount. Such reductions might be desirable to allow a larger production rate for the new, more profitable product 2, so negative values should be allowed for x_1 in the model.

Since the procedure for determining the *leaving basic variable* requires that all the variables have nonnegativity constraints, any problem containing variables allowed to be negative must be converted to an *equivalent* problem involving only nonnegative variables before the simplex method is applied. Fortunately, this conversion can be done. The

[16]Techniques have been developed (and incorporated into linear programming software) to analyze what causes a large linear programming problem to have no feasible solutions so that any errors in the formulation can be corrected. For example, see J. W. Chinneck: *Feasibility and Infeasibility in Optimization: Algorithms and Computational Methods*, Springer Science + Business Media, New York, 2008.

modification required for each variable depends upon whether it has a (negative) lower bound on the values allowed. Each of these two cases is now discussed.

Variables with a Bound on the Negative Values Allowed. Consider any decision variable x_j that is allowed to have negative values which satisfy a constraint of the form

$$x_j \geq L_j,$$

where L_j is some negative constant. This constraint can be converted to a nonnegativity constraint by making the change of variables

$$x_j' = x_j - L_j, \qquad \text{so} \qquad x_j' \geq 0.$$

Thus, $x_j' + L_j$ would be substituted for x_j throughout the model, so that the redefined decision variable x_j' cannot be negative. (This same technique can be used when L_j is *positive* to convert a functional constraint $x_j \geq L_j$ to a nonnegativity constraint $x_j' \geq 0$.)

To illustrate, suppose that the current production rate for product 1 in the Wyndor Glass Co. problem is 10. With the definition of x_1 just given, the complete model at this point is the same as that given in Sec. 3.1 except that the nonnegativity constraint $x_1 \geq 0$ is replaced by

$$x_1 \geq -10.$$

To obtain the equivalent model needed for the simplex method, this decision variable would be redefined as the *total* production rate of product 1

$$x_j' = x_1 + 10,$$

which yields the changes in the objective function and constraints as shown:

$$\boxed{\begin{array}{l} Z = 3x_1 + 5x_2 \\ x_1 \qquad\quad \leq 4 \\ \qquad 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 \\ x_1 \geq -10, \quad x_2 \geq 0 \end{array}} \rightarrow \boxed{\begin{array}{l} Z = 3(x_1' - 10) + 5x_2 \\ x_1' - 10 \qquad\quad \leq 4 \\ \qquad\qquad 2x_2 \leq 12 \\ 3(x_1' - 10) + 2x_2 \leq 18 \\ x_1' - 10 \geq -10, \quad x_2 \geq 0 \end{array}} \rightarrow \boxed{\begin{array}{l} Z = -30 + 3x_1' + 5x_2 \\ x_1' \qquad\quad \leq 14 \\ \qquad 2x_2 \leq 12 \\ 3x_1' + 2x_2 \leq 48 \\ x_1' \geq 0, \quad x_2 \geq 0 \end{array}}$$

Variables with No Bound on the Negative Values Allowed. In the case where x_j does *not* have a lower-bound constraint in the model formulated, another approach is required: x_j is replaced throughout the model by the *difference* of two new *nonnegative* variables

$$x_j = x_j^+ - x_j^-, \qquad \text{where } x_j^+ \geq 0, x_j^- \geq 0.$$

Since x_j^+ and x_j^- can have any nonnegative values, this difference $x_j^+ - x_j^-$ can have *any* value (positive or negative), so it is a legitimate substitute for x_j in the model. But after such substitutions, the simplex method can proceed with just nonnegative variables.

The new variables x_j^+ and x_j^- have a simple interpretation. As explained in the next paragraph, each BF solution for the new form of the model necessarily has the property that *either* $x_j^+ = 0$ or $x_j^- = 0$ (or both). Therefore, at the optimal solution obtained by the simplex method (a BF solution),

$$x_j^+ = \begin{cases} x_j & \text{if } x_j \geq 0, \\ 0 & \text{otherwise;} \end{cases}$$

$$x_j^- = \begin{cases} |x_j| & \text{if } x_j \leq 0, \\ 0 & \text{otherwise;} \end{cases}$$

so that x_j^+ represents the positive part of the decision variable x_j and x_j^- its negative part (as suggested by the superscripts).

For example, if $x_j = 10$, the above expressions give $x_j^+ = 10$ and $x_j^- = 0$. This same value of $x_j = x_j^+ - x_j^- = 10$ also would occur with larger values of x_j^+ and x_j^- such that $x_j^+ = x_j^- + 10$. Plotting these values of x_j^+ and x_j^- on a two-dimensional graph gives a line with an endpoint at $x_j^+ = 10$, $x_j^- = 0$ to avoid violating the nonnegativity constraints. This endpoint is the only corner-point solution on the line. Therefore, only this endpoint can be part of an overall CPF solution or BF solution involving all the variables of the model. This illustrates why each BF solution necessarily has either $x_j^+ = 0$ or $x_j^- = 0$ (or both).

To illustrate the use of the x_j^+ and x_j^-, let us return to the example on the preceding page where x_1 is redefined as the increase over the current production rate of 10 for product 1 in the Wyndor Glass Co. problem.

However, now suppose that the $x_1 \geq -10$ constraint was not included in the original model because it clearly would not change the optimal solution. (In some problems, certain variables do not need explicit lower-bound constraints because the functional constraints already prevent lower values.) Therefore, before the simplex method is applied, x_1 would be replaced by the difference

$$x_1 = x_1^+ - x_1^-, \qquad \text{where } x_1^+ \geq 0,\ x_1^- \geq 0,$$

as shown:

$$\boxed{\begin{array}{ll} \text{Maximize} & Z = 3x_1 + 5x_2, \\ \text{subject to} & \begin{array}{r} x_1 \qquad \leq \ \ 4 \\ 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 \\ x_2 \geq 0 \text{ (only)} \end{array} \end{array}} \longrightarrow \boxed{\begin{array}{ll} \text{Maximize} & Z = 3x_1^+ - 3x_1^- + 5x_2, \\ \text{subject to} & \begin{array}{r} x_1^+ - x_1^- \qquad \leq \ \ 4 \\ 2x_2 \leq 12 \\ 3x_1^+ - 3x_1^- + 2x_2 \leq 18 \\ x_1^+ \geq 0, \quad x_1^- \geq 0, \quad x_2 \geq 0 \end{array} \end{array}}$$

From a computational viewpoint, this approach has the disadvantage that the new equivalent model to be used has more variables than the original model. In fact, if *all* the original variables lack lower-bound constraints, the new model will have *twice* as many variables. Fortunately, the approach can be modified slightly so that the number of variables is increased by only one, regardless of how many original variables need to be replaced. This modification is done by replacing each such variable x_j by

$$x_j = x_j' - x'', \qquad \text{where } x_j' \geq 0,\ x'' \geq 0,$$

instead, where x'' is the *same* variable for all relevant j. The interpretation of x'' in this case is that $-x''$ is the current value of the *largest* (in absolute terms) negative original variable, so that x_j' is the amount by which x_j exceeds this value. Thus, the simplex method now can make some of the x_j' variables larger than zero even when $x'' > 0$.

4.7 POSTOPTIMALITY ANALYSIS

We stressed in Secs. 2.3, 2.4, and 2.5 that *postoptimality analysis*—the analysis done *after* an optimal solution is obtained for the initial version of the model—constitutes a very major and very important part of most operations research studies. The fact that postoptimality analysis is very important is particularly true for typical linear programming applications. In this section, we focus on the role of the simplex method in performing this analysis.

Table 4.17 summarizes the typical steps in postoptimality analysis for linear programming studies. The rightmost column identifies some algorithmic techniques that

TABLE 4.17 Postoptimality analysis for linear programming

Task	Purpose	Technique
Model debugging	Find errors and weaknesses in model	Reoptimization
Model validation	Demonstrate validity of final model	See Sec. 2.4
Final managerial decisions on resource allocations (the b_i values)	Make appropriate division of organizational resources between activities under study and other important activities	Shadow prices
Evaluate estimates of model parameters	Determine crucial estimates that may affect optimal solution for further study	Sensitivity analysis
Evaluate trade-offs between model parameters	Determine best trade-off	Parametric linear programming

involve the simplex method. These techniques are introduced briefly here with the technical details deferred to later chapters.

Reoptimization

As discussed in Sec. 3.6, linear programming models that arise in practice commonly are very large, with hundreds, thousands, or even millions of functional constraints and decision variables. In such cases, many variations of the basic model may be of interest for considering different scenarios. Therefore, after having found an optimal solution for one version of a linear programming model, we frequently must solve again (often many times) for the solution of a slightly different version of the model. We nearly always have to solve again several times during the model debugging stage (described in Secs. 2.3 and 2.4), and we usually have to do so a large number of times during the later stages of postoptimality analysis as well.

One approach is simply to reapply the simplex method from scratch for each new version of the model, even though each run may require hundreds or even thousands of iterations for large problems. However, a *much more efficient* approach is to *reoptimize.* Reoptimization involves deducing how changes in the model get carried along to the *final* simplex tableau (as described in Secs. 5.3 and 6.6). This revised tableau and the optimal solution for the prior model are then used as the *initial tableau* and the *initial basic solution* for solving the new model. If this solution is feasible for the new model, then the simplex method is applied in the usual way, starting from this initial BF solution. If the solution is not feasible, a related algorithm called the *dual simplex method* (described in Sec. 7.1) probably can be applied to find the new optimal solution,[17] starting from this initial basic solution.

The big advantage of this **reoptimization technique** over re-solving from scratch is that an optimal solution for the revised model probably is going to be *much* closer to the prior optimal solution than to an initial BF solution constructed in the usual way for the simplex method. Therefore, assuming that the model revisions were modest, only a few iterations should be required to reoptimize instead of the hundreds or thousands that may be required when you start from scratch. In fact, the optimal solutions for the prior and revised models are frequently the same, in which case the reoptimization technique requires only one application of the optimality test and *no* iterations.

[17]The one requirement for using the dual simplex method here is that the *optimality test* is still passed when applied to row 0 of the *revised* final tableau. If not, then still another algorithm called the *primal-dual method* can be used instead.

Shadow Prices

Recall that linear programming problems often can be interpreted as allocating resources to activities. In particular, when the functional constraints are in $\leq$ form, we interpreted the b_i (the right-hand sides) as the amounts of the respective resources being made available for the activities under consideration. In many cases, there may be some latitude in the amounts that will be made available. If so, the b_i values used in the initial (validated) model actually may represent management's *tentative initial decision* on how much of the organization's resources will be provided to the activities considered in the model instead of to other important activities under the purview of management. From this broader perspective, some of the b_i values can be increased in a revised model, but only if a sufficiently strong case can be made to management that this revision would be beneficial.

Consequently, information on the economic contribution of the resources to the measure of performance (Z) for the current study often would be extremely useful. The simplex method provides this information in the form of *shadow prices* for the respective resources.

> The **shadow price** for resource i (denoted by y_i^*) measures the *marginal value* of this resource, i.e., the rate at which Z could be increased by (slightly) increasing the amount of this resource (b_i) being made available.[18,19] The simplex method identifies this shadow price by y_i^* = coefficient of the ith slack variable in row 0 of the final simplex tableau.

To illustrate, for the Wyndor Glass Co. problem,

Resource i = production capacity of Plant i ($i = 1, 2, 3$) being made available to the two new products under consideration,

b_i = hours of production time per week being made available in Plant i for these new products.

Providing a substantial amount of production time for the new products would require adjusting production times for the current products, so choosing the b_i value is a difficult managerial decision. The tentative initial decision has been

$$b_1 = 4, \qquad b_2 = 12, \qquad b_3 = 18,$$

as reflected in the basic model considered in Sec. 3.1 and in this chapter. However, management now wishes to evaluate the effect of changing any of the b_i values.

The shadow prices for these three resources provide just the information that management needs. The final tableau in Table 4.8 yields

$$y_1^* = 0 = \text{shadow price for resource 1},$$

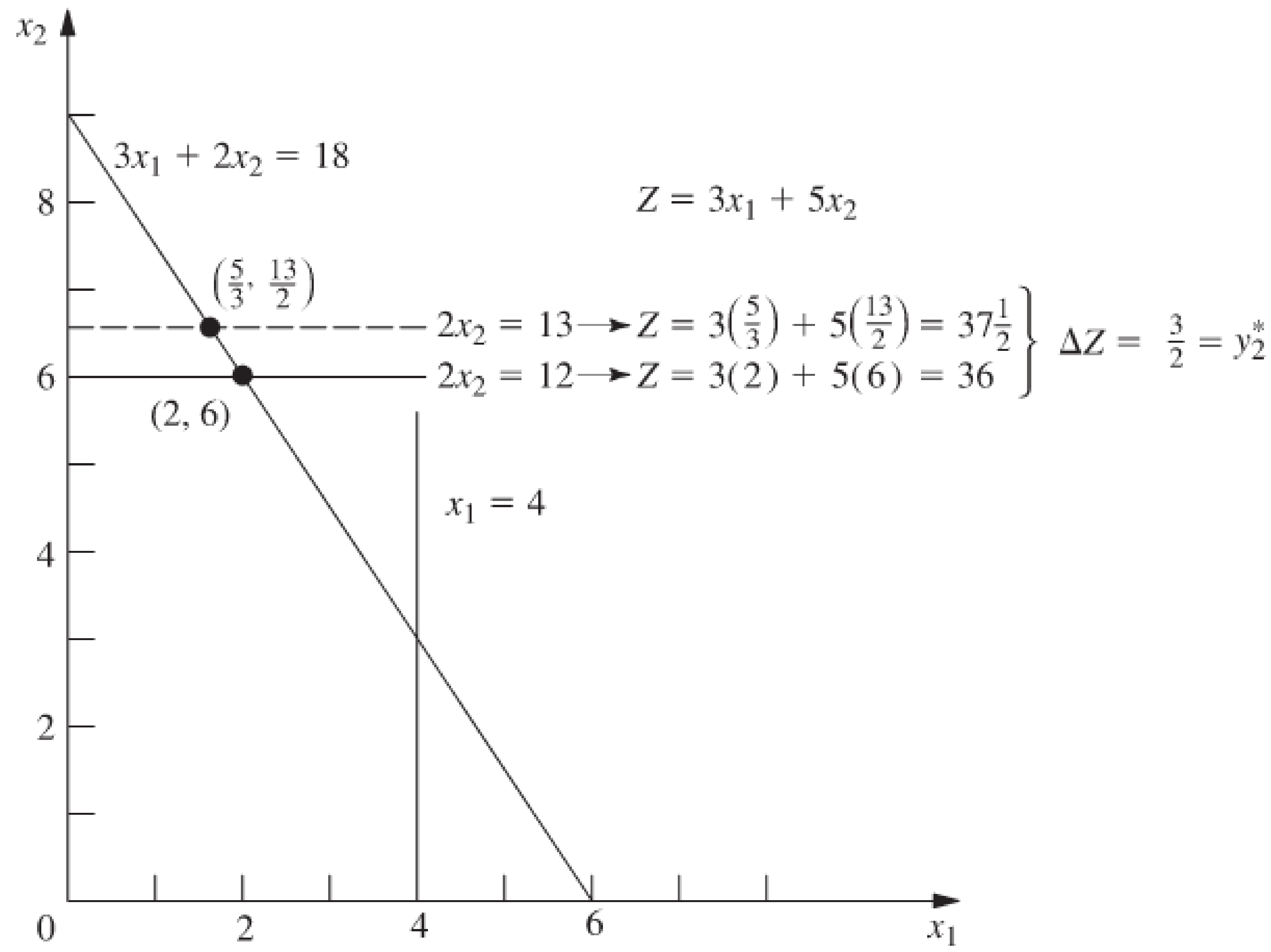
$$y_2^* = \frac{3}{2} = \text{shadow price for resource 2},$$

$$y_3^* = 1 = \text{shadow price for resource 3}.$$

With just two decision variables, these numbers can be verified by checking graphically that individually increasing any b_i by 1 indeed would increase the optimal value of Z by y_i^*. For example, Fig. 4.8 demonstrates this increase for resource 2 by reapplying the

[18]The increase in b_i must be sufficiently small that the current set of basic variables remains optimal since this rate (marginal value) changes if the set of basic variables changes.

[19]In the case of a functional constraint in $\geq$ or $=$ form, its shadow price is again defined as the rate at which Z could be increased by (slightly) increasing the value of b_i, although the interpretation of b_i now would normally be something other than the amount of a resource being made available.



■ **FIGURE 4.8**
This graph shows that the shadow price is $y_2^* = \frac{3}{2}$ for resource 2 for the Wyndor Glass Co. problem. The two dots are the optimal solutions for $b_2 = 12$ or $b_2 = 13$, and plugging these solutions into the objective function reveals that increasing b_2 by 1 increases Z by $y_2^* = \frac{3}{2}$.

graphical method presented in Sec. 3.1. The optimal solution, (2, 6) with $Z = 36$, changes to $(\frac{5}{3}, \frac{13}{2})$ with $Z = 37\frac{1}{2}$ when b_2 is increased by 1 (from 12 to 13), so that

$$y_2^* = \Delta Z = 37\frac{1}{2} - 36 = \frac{3}{2}.$$

Since Z is expressed in thousands of dollars of profit per week, $y_2^* = \frac{3}{2}$ indicates that adding 1 more hour of production time per week in Plant 2 for these two new products would increase their total profit by \$1,500 per week. Should this actually be done? It depends on the marginal profitability of other products currently using this production time. If there is a current product that contributes less than \$1,500 of weekly profit per hour of weekly production time in Plant 2, then some shift of production time to the new products would be worthwhile.

We shall continue this story in Sec. 6.7, where the Wyndor OR team uses shadow prices as part of its *sensitivity analysis* of the model.

Figure 4.8 demonstrates that $y_2^* = \frac{3}{2}$ is the rate at which Z could be increased by increasing b_2 slightly. However, it also demonstrates the common phenomenon that this interpretation holds only for a small increase in b_2. Once b_2 is increased beyond 18, the optimal solution stays at (0, 9) with no further increase in Z. (At that point, the set of basic variables in the optimal solution has changed, so a new final simplex tableau will be obtained with new shadow prices, including $y_2^* = 0$.)

Now note in Fig. 4.8 why $y_1^* = 0$. Because the constraint on resource 1, $x_1 \leq 4$, is *not binding* on the optimal solution (2, 6), there is a *surplus* of this resource. Therefore, increasing b_1 beyond 4 cannot yield a new optimal solution with a larger value of Z.

By contrast, the constraints on resources 2 and 3, $2x_2 \leq 12$ and $3x_1 + 2x_2 \leq 18$, are **binding constraints** (constraints that hold with equality at the optimal solution). Because the limited supply of these resources ($b_2 = 12$, $b_3 = 18$) *binds* Z from being increased further, they have *positive* shadow prices. Economists refer to such resources as *scarce goods,* whereas resources available in surplus (such as resource 1) are *free goods* (resources with a zero shadow price).

The kind of information provided by shadow prices clearly is valuable to management when it considers reallocations of resources within the organization. It also is very helpful when an increase in b_i can be achieved only by going outside the organization to purchase more of the resource in the marketplace. For example, suppose that Z represents *profit* and that the unit profits of the activities (the c_j values) include the costs (at regular prices) of all the resources consumed. Then a *positive* shadow price of y_i^* for resource i means that the total profit Z can be increased by y_i^* by purchasing 1 more unit of this resource at its regular price. Alternatively, if a *premium* price must be paid for the resource in the marketplace, then y_i^* represents the *maximum* premium (excess over the regular price) that would be worth paying.[20]

The theoretical foundation for shadow prices is provided by the duality theory described in Chap. 6.

Sensitivity Analysis

When discussing the *certainty assumption* for linear programming at the end of Sec. 3.3, we pointed out that the values used for the model parameters (the a_{ij}, b_i, and c_j identified in Table 3.3) generally are just *estimates* of quantities whose true values will not become known until the linear programming study is implemented at some time in the future. A main purpose of sensitivity analysis is to identify the **sensitive parameters** (i.e., those that cannot be changed without changing the optimal solution). The sensitive parameters are the parameters that need to be estimated with special care to minimize the risk of obtaining an erroneous optimal solution. They also will need to be monitored particularly closely as the study is implemented. If it is discovered that the true value of a sensitive parameter differs from its estimated value in the model, this immediately signals a need to change the solution.

How are the sensitive parameters identified? In the case of the b_i, you have just seen that this information is given by the shadow prices provided by the simplex method. In particular, if $y_i^* > 0$, then the optimal solution changes if b_i is changed, so b_i is a sensitive parameter. However, $y_i^* = 0$ implies that the optimal solution is not sensitive to at least small changes in b_i. Consequently, if the value used for b_i is an estimate of the amount of the resource that will be available (rather than a managerial decision), then the b_i values that need to be monitored more closely are those with *positive* shadow prices—especially those with *large* shadow prices.

When there are just two variables, the sensitivity of the various parameters can be analyzed graphically. For example, in Fig. 4.9, $c_1 = 3$ can be changed to any other value from 0 to 7.5 without the optimal solution changing from (2, 6). (The reason is that any value of c_1 within this range keeps the slope of $Z = c_1x_1 + 5x_2$ between the slopes of the lines $2x_2 = 12$ and $3x_1 + 2x_2 = 18$.) Similarly, if $c_2 = 5$ is the only parameter changed, it can have any value greater than 2 without affecting the optimal solution. Hence, neither c_1 nor c_2 is a sensitive parameter. (The procedure called **Graphical Method and Sensitivity Analysis** in IOR Tutorial enables you to perform this kind of graphical analysis very efficiently.)

The easiest way to analyze the sensitivity of each of the a_{ij} parameters graphically is to check whether the corresponding constraint is *binding* at the optimal solution. Because $x_1 \leq 4$ is *not* a binding constraint, any sufficiently small change in its coefficients ($a_{11} = 1$, $a_{12} = 0$) is not going to change the optimal solution, so these are *not* sensitive parameters. On the other hand, both $2x_2 \leq 12$ and $3x_1 + 2x_2 \leq 18$ are *binding constraints,*

[20]If the unit profits do *not* include the costs of the resources consumed, then y_i^* represents the maximum *total* unit price that would be worth paying to increase b_i.

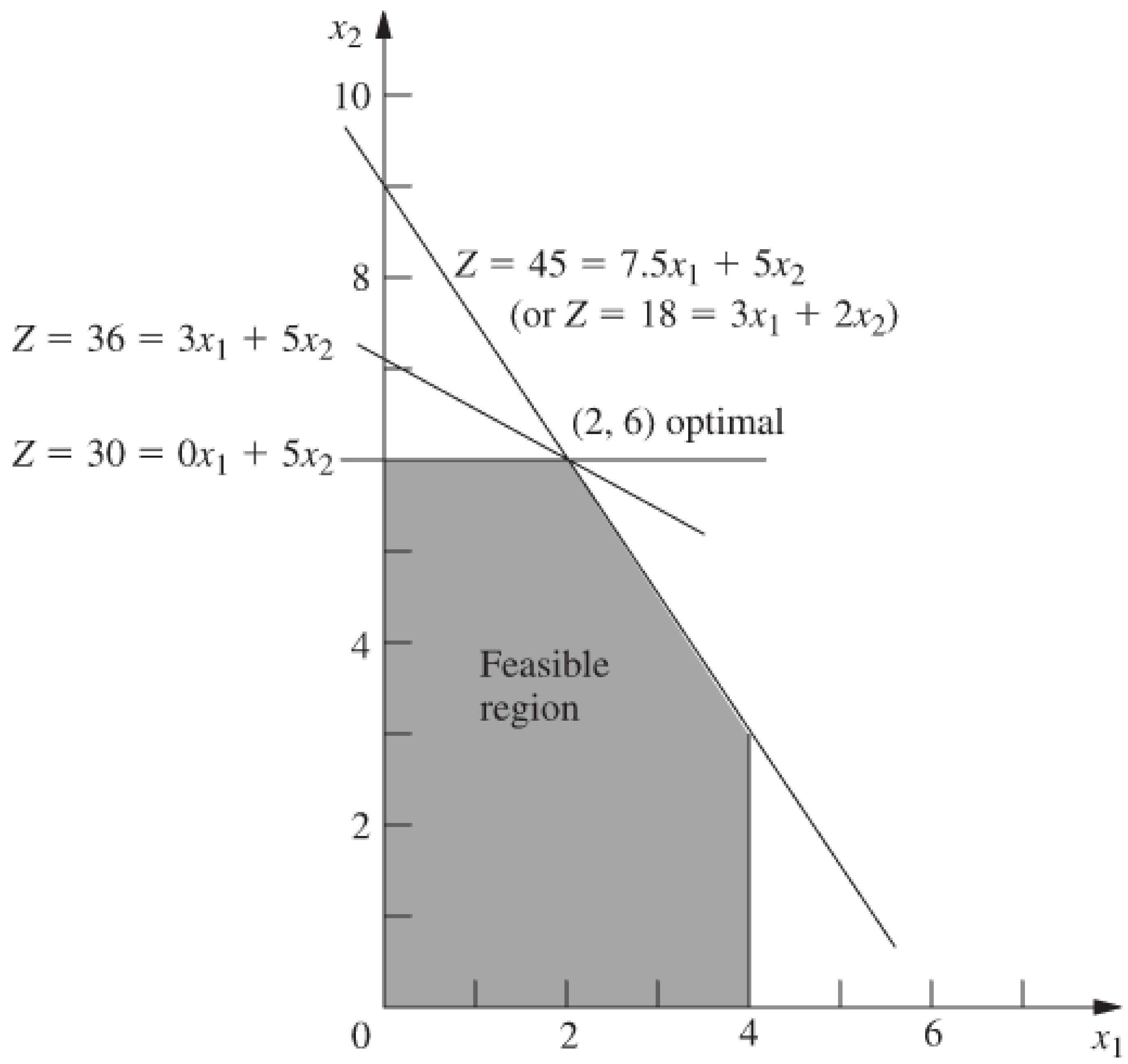


FIGURE 4.9
This graph demonstrates the sensitivity analysis of c_1 and c_2 for the Wyndor Glass Co. problem. Starting with the original objective function line [where $c_1 = 3$, $c_2 = 5$, and the optimal solution is (2, 6)], the other two lines show the extremes of how much the slope of the objective function line can change and still retain (2, 6) as an optimal solution. Thus, with $c_2 = 5$, the allowable range for c_1 is $0 \leq c_1 \leq 7.5$. With $c_1 = 3$, the allowable range for c_2 is $c_2 \geq 2$.

so changing *any* one of their coefficients ($a_{21} = 0$, $a_{22} = 2$, $a_{31} = 3$, $a_{32} = 2$) is going to change the optimal solution, and therefore these are sensitive parameters.

Typically, greater attention is given to performing sensitivity analysis on the b_i and c_j parameters than on the a_{ij} parameters. On real problems with hundreds or thousands of constraints and variables, the effect of changing one a_{ij} value is usually negligible, but changing one b_i or c_j value can have real impact. Furthermore, in many cases, the a_{ij} values are determined by the technology being used (the a_{ij} values are sometimes called *technological coefficients*), so there may be relatively little (or no) uncertainty about their final values. This is fortunate, because there are far more a_{ij} parameters than b_i and c_j parameters for large problems.

For problems with more than two (or possibly three) decision variables, you cannot analyze the sensitivity of the parameters graphically as was just done for the Wyndor Glass Co. problem. However, you can extract the same kind of information from the simplex method. Getting this information requires using the *fundamental insight* described in Sec. 5.3 to deduce the changes that get carried along to the final simplex tableau as a result of changing the value of a parameter in the original model. The rest of the procedure is described and illustrated in Secs. 6.6 and 6.7.

Using Excel to Generate Sensitivity Analysis Information

Sensitivity analysis normally is incorporated into software packages based on the simplex method. For example, the Excel Solver will generate sensitivity analysis information upon request. As was shown in Fig. 3.21, when the Solver gives the message that it has found a solution, it also gives on the right a list of three reports that can be provided. By selecting the second one (labeled "Sensitivity") after solving the Wyndor Glass Co. problem, you will obtain the *sensitivity report* shown in Fig. 4.10. The upper table in this report provides sensitivity analysis information about the decision variables and their coefficients in the objective function. The lower table does the same for the functional constraints and their right-hand sides.

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$C$12	Batches Produced Doors	2	0	3,000	4,500	3,000
$D$12	Batches Produced Windows	6	0	5,000	1E+30	3,000

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$E$7	Plant 1 Used	2	0	4	1E+30	2
$E$8	Plant 2 Used	12	1,500	12	6	6
$E$9	Plant 3 Used	18	1,000	18	6	6

■ **FIGURE 4.10**
The sensitivity report provided by the Excel Solver for the Wyndor Glass Co. problem.

Look first at the upper table in this figure. The "Final Value" column indicates the optimal solution. The next column gives the *reduced costs.* (We will not discuss these reduced costs now because the information they provide can also be gleaned from the rest of the upper table.) The next three columns provide the information needed to identify the *allowable range* for each coefficient c_j in the objective function.

> For any c_j, its **allowable range** is the range of values for this coefficient over which the current optimal solution remains optimal, assuming no change in the other coefficients.

The "Objective Coefficient" column gives the current value of each coefficient, and then the next two columns give the *allowable increase* and the *allowable decrease* from this value to remain within the allowable range. The spreadsheet model (Fig. 3.22) expresses the profits per batch in units of *dollars,* whereas the c_j in the algebraic version of the linear programming model uses units of *thousands of dollars,* so the quantities in all three of these columns need to be divided by 1000 to use the same units as the c_j. Therefore,

$$\frac{3{,}000 - 3{,}000}{1{,}000} \le c_1 \le \frac{3{,}000 + 4{,}500}{1{,}000}, \qquad \text{so} \qquad 0 \le c_1 \le 7.5$$

is the allowable range for c_1 over which the current optimal solution will stay optimal (assuming $c_2 = 5$), just as was found graphically in Fig. 4.9. Similarly, since Excel uses $1E + 30$ (10^{30}) to represent infinity,

$$\frac{5{,}000 - 3{,}000}{1{,}000} \le c_2 \le \frac{5{,}000 + \infty}{1{,}000}, \qquad \text{so} \qquad 2 \le c_2$$

is the allowable range for c_2.

The fact that both the allowable increase and the allowable decrease are greater than zero for the coefficient of both decision variables provides another useful piece of information, as described below.

> When the upper table in the sensitivity report generated by the Excel Solver indicates that both the allowable increase and the allowable decrease are greater than zero for every objective coefficient, this is a signpost that the optimal solution in the "Final Value" column is the only optimal solution. Conversely, having any allowable increase or allowable decrease equal to zero is a signpost that there are multiple optimal solutions. Changing the corresponding coefficient a tiny amount beyond the zero allowed and re-solving provides another optimal CPF solution for the original model.

Now consider the lower table in Fig. 4.10 that focuses on sensitivity analysis for the three functional constraints. The "Final Value" column gives the value of each constraint's left-hand side for the optimal solution. The next two columns give the shadow price and the current value of the right-hand side (b_i) for each constraint. (These shadow prices from the spreadsheet model use units of *dollars,* so they need to be divided by 1000 to use the same units of *thousands of dollars* as Z in the algebraic version of the linear programming model.) When just one b_i value is then changed, the last two columns give the *allowable increase* or *allowable decrease* in order to remain within its *allowable range.*

> For any b_i, its **allowable range** is the range of values for this right-hand side over which the current optimal BF solution (with adjusted values[21] for the basic variables) remains feasible, assuming no change in the other right-hand sides. A key property of this range of values is that the current *shadow price* for b_i remains valid for evaluating the effect on Z of changing b_i only as long as b_i remains within this allowable range.

Thus, using the lower table in Fig. 4.10, combining the last two columns with the current values of the right-hand sides gives the following allowable ranges:

$$2 \le b_1$$
$$6 \le b_2 \le 18$$
$$12 \le b_3 \le 24.$$

This sensitivity report generated by the Excel Solver is typical of the sensitivity analysis information provided by linear programming software packages. You will see in Appendix 4.1 that LINDO and LINGO provide essentially the same report. MPL/CPLEX does also when it is requested with the Solution File dialogue box. Once again, this information obtained algebraically also can be derived from graphical analysis for this two-variable problem. (See Prob. 4.7-1.) For example, when b_2 is increased from 12 in Fig. 4.8, the originally optimal CPF solution at the intersection of two constraint boundaries $2x_2 = b_2$ and $3x_1 + 2x_2 = 18$ will remain feasible (including $x_1 \ge 0$) only for $b_2 \le 18$.

The Worked Examples section of the book's website includes **another example** of applying sensitivity analysis (using both graphical analysis and the sensitivity report). The latter part of Chap. 6 also will delve into this type of analysis more deeply.

Parametric Linear Programming

Sensitivity analysis involves changing one parameter at a time in the original model to check its effect on the optimal solution. By contrast, **parametric linear programming** (or **parametric programming** for short) involves the systematic study of how the optimal solution changes as *many* of the parameters change *simultaneously* over some range. This study can provide a very useful extension of sensitivity analysis, e.g., to check the effect of "correlated" parameters that change together due to exogenous factors such as the state of the economy. However, a more important application is the investigation of *trade-offs* in parameter values. For example, if the c_j values represent the unit profits of the respective activities, it may be possible to increase some of the c_j values at the expense of decreasing others by an appropriate shifting of personnel and equipment among activities. Similarly, if the

[21]Since the values of the basic variables are obtained as the simultaneous solution of a system of equations (the functional constraints in augmented form), at least some of these values change if one of the right-hand sides changes. However, the adjusted values of the current set of basic variables still will satisfy the nonnegativity constraints, and so still will be feasible, as long as the new value of this right-hand side remains within its allowable range. If the adjusted basic solution is still feasible, it also will still be optimal. We shall elaborate further in Sec. 6.7.

b_i values represent the amounts of the respective resources being made available, it may be possible to increase some of the b_i values by agreeing to accept decreases in some of the others. The analysis of such possibilities is discussed and illustrated at the end of Sec. 6.7.

In some applications, the main purpose of the study is to determine the most appropriate trade-off between two basic factors, such as *costs* and *benefits*. The usual approach is to express one of these factors in the objective function (e.g., minimize total cost) and incorporate the other into the constraints (e.g., benefits $\geq$ minimum acceptable level), as was done for the Nori & Leets Co. air pollution problem in Sec. 3.4. Parametric linear programming then enables systematic investigation of what happens when the initial tentative decision on the trade-off (e.g., the minimum acceptable level for the benefits) is changed by improving one factor at the expense of the other.

The algorithmic technique for parametric linear programming is a natural extension of that for sensitivity analysis, so it, too, is based on the simplex method. The procedure is described in Sec. 7.2.

4.8 COMPUTER IMPLEMENTATION

If the electronic computer had never been invented, you probably would have never heard of linear programming and the simplex method. Even though it is possible to apply the simplex method by hand (perhaps with the aid of a calculator) to solve tiny linear programming problems, the calculations involved are just too tedious to do this on a routine basis. However, the simplex method is ideally suited for execution on a computer. It is the computer revolution that has made possible the widespread application of linear programming in recent decades.

Implementation of the Simplex Method

Computer codes for the simplex method now are widely available for essentially all modern computer systems. These codes commonly are part of a sophisticated software package for mathematical programming that includes many of the procedures described in subsequent chapters (including those used for postoptimality analysis).

These production computer codes do not closely follow either the algebraic form or the tabular form of the simplex method presented in Secs. 4.3 and 4.4. These forms can be streamlined considerably for computer implementation. Therefore, the codes use instead a *matrix form* (usually called the *revised simplex method*) that is especially well suited for the computer. This form accomplishes exactly the same things as the algebraic or tabular form, but it does this while computing and storing only the numbers that are actually needed for the current iteration; and then it carries along the essential data in a more compact form. The revised simplex method is described in Secs. 5.2 and 5.4.

The simplex method is used routinely to solve surprisingly large linear programming problems. For example, powerful desktop computers (including workstations) commonly are used to solve problems with hundreds of thousands, or even millions, of functional constraints and a larger number of decision variables. Occasionally, successfully solved problems have even tens of millions of functional constraints and decision variables.[22] For certain *special types* of linear programming problems (such as the transportation,

[22]Do not try this at home. Attacking such a massive problem requires an especially sophisticated linear programming system that uses the latest techniques for exploiting sparcity in the coefficient matrix as well as other special techniques (e.g., *crashing techniques* for quickly finding an advanced initial BF solution). When problems are re-solved periodically after minor updating of the data, much time often is saved by using (or modifying) the last optimal solution to provide the initial BF solution for the new run.

assignment, and minimum cost flow problems to be described later in the book), even larger problems now can be solved by *specialized* versions of the simplex method.

Several factors affect how long it will take to solve a linear programming problem by the general simplex method. The most important one is the *number of ordinary functional constraints*. In fact, computation time tends to be roughly proportional to the cube of this number, so that doubling this number may multiply the computation time by a factor of approximately 8. By contrast, the number of variables is a relatively minor factor.[23] Thus, doubling the number of variables probably will not even double the computation time. A third factor of some importance is the *density* of the table of constraint coefficients (i.e., the *proportion* of the coefficients that are *not* zero), because this affects the computation time *per iteration*. (For large problems encountered in practice, it is common for the density to be under 5 percent, or even under 1 percent, and this much "sparcity" tends to greatly accelerate the simplex method.) One common rule of thumb for the *number of iterations* is that it tends to be roughly twice the number of functional constraints.

With large linear programming problems, it is inevitable that some mistakes and faulty decisions will be made initially in formulating the model and inputting it into the computer. Therefore, as discussed in Sec. 2.4, a thorough process of testing and refining the model (*model validation*) is needed. The usual end product is not a single static model that is solved once by the simplex method. Instead, the OR team and management typically consider a long series of variations on a basic model (sometimes even thousands of variations) to examine different scenarios as part of postoptimality analysis. This entire process is greatly accelerated when it can be carried out *interactively* on a *desktop computer.* And, with the help of both mathematical programming modeling languages and improving computer technology, this now is becoming common practice.

Until the mid-1980s, linear programming problems were solved almost exclusively on *mainframe computers.* Since then, there has been an explosion in the capability of doing linear programming on desktop computers, including personal computers as well as workstations. Workstations, including some with parallel processing capabilities, now are commonly used instead of mainframe computers to solve massive linear programming models. The fastest personal computers are not lagging far behind, although solving huge models usually requires additional memory.

Linear Programming Software Featured in This Book

A considerable number of excellent software packages for linear programming and its extensions now are available to fill a variety of needs. One leading package of this type is **Express-MP**, a product of Dash Optimization (which now has joined Fair Isaac). Another that is widely regarded to be a particularly powerful package for solving massive problems is **CPLEX,** a product of ILOG, Inc., located in Silicon Valley. Since 1988, CPLEX has helped to lead the way in solving larger and larger linear programming problems. An extensive research and development effort has enabled a series of upgrades with dramatic increases in efficiency. CPLEX 11 released in 2007 provided another major improvement. This software package frequently is capable of solving real linear programming problems arising in industry with tens of millions of functional constraints and decision variables! CPLEX often uses the simplex method and its variants (such as the dual simplex method presented in Sec. 7.1) to solve these massive problems. In addition to the simplex method, CPLEX also features some other powerful weapons for attacking linear programming problems. One is a lightning-fast algorithm (referred to as the *barrier algorithm*) that uses the *interior-point approach* introduced in Section 4.9. This algorithm can solve some huge

[23]This statement assumes that the revised simplex method described in Secs. 5.2 and 5.4 is being used.

general linear programming problems that the simplex method cannot (and vice versa). Another feature is the *network simplex method* (described in Sec. 9.7) that can solve even larger special types of linear programming problems. CPLEX 11 also extends beyond linear programming by including state-of-the-art algorithms for *integer programming* (Chap. 11) and *quadratic programming* (Sec. 12.7), as well as *integer quadratic programming*.

We anticipate that these major improvements in the state-of-the-art optimization software packages such as CPLEX will continue in the future as well. Continuing rapid improvements in the speed of computers also will further accelerate the speedup of these future software packages.

Because it often is used to solve really large problems, CPLEX normally is used in conjunction with a mathematical programming *modeling language.* As described in Sec. 3.7, modeling languages are designed for efficiently formulating large linear programming models (and related models) in a compact way, after which a solver is called upon to solve the model. Several of the prominent modeling languages support CPLEX as a solver. ILOG also has introduced its own modeling language, called the *Optimization Programming Language (OPL),* that can be used with CPLEX to form the *OPL-CPLEX Development System*. (A trial version of that product is available at ILOG's website, www.ilog.com.)

As we mentioned in Sec. 3.7, the student version of CPLEX is included in your OR Courseware as the main solver for the MPL modeling language. This version features the simplex method for solving linear programming problems.

The student version of MPL in your OR Courseware also includes two other solvers that are an alternative to CPLEX for solving both linear programming problems and integer programming problems (discussed in Chap. 11). One is **CoinMP,** an open source solver that can solve larger problems than the student version of CPLEX (which is limited to 300 constraints and variables). The other is LINDO.

LINDO (short for Linear, Interactive, and Discrete Optimizer) has an even longer history than CPLEX in the realm of applications of linear programming and its extensions. The easy-to-use LINDO interface is available as a subset of the **LINGO** optimization modeling package from LINDO Systems, www.lindo.com. The long-time popularity of LINDO is partially due to its ease of use. For "textbook-sized" problems, the model can be entered and solved in an intuitive, straightforward manner, so the LINDO interface provides a convenient tool for students to use. Although easy to use for small models, LINDO/LINGO can also solve large models, e.g., the largest version has solved real problems with 4 million variables and 2 million constraints.

The OR Courseware provided on this book's website contains a student version of LINDO/LINGO, accompanied by an extensive tutorial. Appendix 4.1 provides a quick introduction. Additionally, the software contains extensive online help. The OR Courseware also contains LINGO/LINDO formulations for the major examples used in the book.

Spreadsheet-based solvers are becoming increasingly popular for linear programming and its extensions. Leading the way are the solvers produced by Frontline Systems for Microsoft Excel and other spreadsheet packages. In addition to the basic solver shipped with these packages, more powerful *Premium Solver* products also are available. Because of the widespread use of spreadsheet packages such as Microsoft Excel today, these solvers are introducing large numbers of people to the potential of linear programming for the first time. For textbook-sized linear programming problems (and considerably larger problems as well), spreadsheets provide a convenient way to formulate and solve the model, as described in Sec. 3.5. The more powerful spreadsheet solvers can solve fairly large models with many thousand decision variables. However, when the spreadsheet grows to an unwieldy size, a good modeling language and its solver may provide a more efficient approach to formulating and solving the model.

Spreadsheets provide an excellent communication tool, especially when dealing with typical managers who are very comfortable with this format but not with the algebraic formulations of OR models. Therefore, optimization software packages and modeling languages now can commonly import and export data and results in a spreadsheet format. For example, the MPL modeling language now includes an enhancement (called the *OptiMax 2000 Component Library*) that enables the modeler to create the feel of a spreadsheet model for the user of the model while still using MPL to formulate the model very efficiently. (The student version of OptiMax 2000 is included in your OR Courseware.)

Premium Solver for Education is one of the Excel add-ins included on the book's website. You can install this add-in to obtain more functionality than with the standard Excel Solver.

Consequently, all the software, tutorials, and examples packed on the book's website are providing you with several attractive software options for linear programming.

Available Software Options for Linear Programming

1. Demonstration examples (in OR Tutor) and both interactive and automatic procedures in IOR Tutorial for efficiently learning the simplex method.
2. Excel and its Premium Solver for formulating and solving linear programming models in a spreadsheet format.
3. MPL/CPLEX for efficiently formulating and solving large linear programming models.
4. LINGO and its solver (shared with LINDO) for an alternative way of efficiently formulating and solving large linear programming models.

Your instructor may specify which software to use. Whatever the choice, you will be gaining experience with the kind of state-of-the-art software that is used by OR professionals.

4.9 THE INTERIOR-POINT APPROACH TO SOLVING LINEAR PROGRAMMING PROBLEMS

The most dramatic new development in operations research during the 1980s was the discovery of the interior-point approach to solving linear programming problems. This discovery was made in 1984 by a young mathematician at AT&T Bell Laboratories, Narendra Karmarkar, when he successfully developed a new algorithm for linear programming with this kind of approach. Although this particular algorithm experienced only mixed success in competing with the simplex method, the key solution concept described below appeared to have great potential for solving *huge* linear programming problems beyond the reach of the simplex method. Many top researchers subsequently worked on modifying Karmarkar's algorithm to fully tap this potential. Much progress has been made (and continues to be made), and a number of powerful algorithms using the interior-point approach have been developed. Today, the more powerful software packages that are designed for solving really large linear programming problems (such as CPLEX) include at least one algorithm using the interior-point approach along with the simplex method and its variants. As research continues on these algorithms, their computer implementations continue to improve. This has spurred renewed research on the simplex method, and its computer implementations continue to improve as well. The competition between the two approaches for supremacy in solving huge problems is continuing.

Now let us look at the key idea behind Karmarkar's algorithm and its subsequent variants that use the interior-point approach.

The Key Solution Concept

Although radically different from the simplex method, Karmarkar's algorithm does share a few of the same characteristics. It is an *iterative* algorithm. It gets started by identifying a feasible *trial solution*. At each iteration, it moves from the current trial solution to a better trial solution in the feasible region. It then continues this process until it reaches a trial solution that is (essentially) optimal.

The big difference lies in the nature of these trial solutions. For the simplex method, the trial solutions are *CPF solutions* (or BF solutions after augmenting), so all movement is along edges on the *boundary* of the feasible region. For Karmarkar's algorithm, the trial solutions are **interior points,** i.e., points *inside* the boundary of the feasible region. For this reason, Karmarkar's algorithm and its variants are referred to as **interior-point algorithms.**

However, because of an early patent obtained on an early version of an interior-point algorithm, such an algorithm also is commonly referred to as a **barrier algorithm** (or *barrier method*). The term *barrier* is used because, from the perspective of a search whose trial solutions are *interior points,* each constraint boundary is treated as a barrier. Most optimization software packages now use the barrier terminology when referring to their solver option that is based on the interior-point approach. Both CPLEX and LINDO API include a "barrier algorithm" that can be used to solve either linear programming problems or quadratic programming problems (discussed in Sec. 12.7).

To illustrate the interior-point approach, Fig. 4.11 shows the path followed by the interior-point algorithm in your OR Courseware when it is applied to the Wyndor Glass Co. problem, starting from the initial trial solution (1, 2). Note how all the trial solutions (dots)

FIGURE 4.11
The curve from (1, 2) to (2, 6) shows a typical path followed by an interior-point algorithm, right through the *interior* of the feasible region for the Wyndor Glass Co. problem.

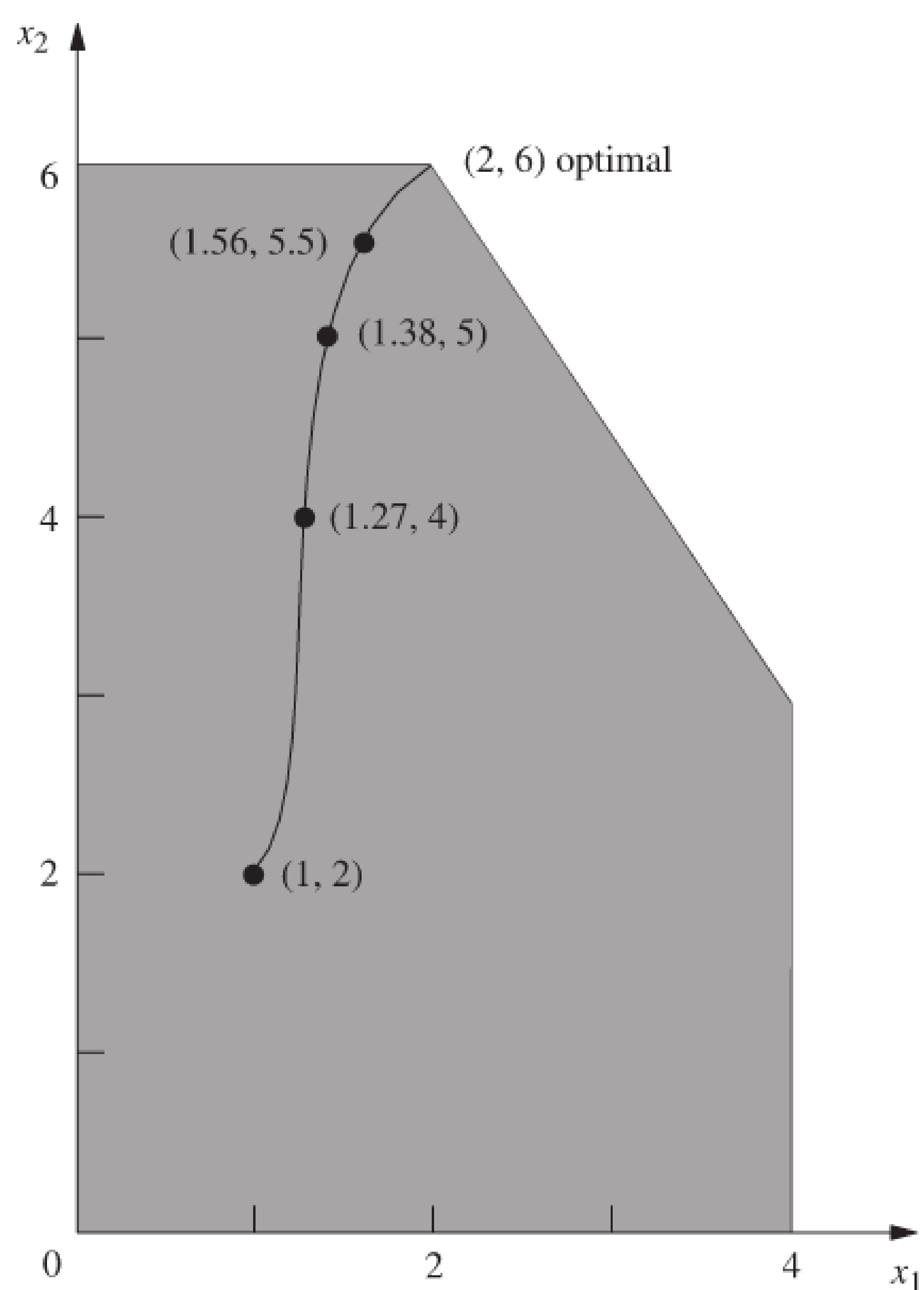


TABLE 4.18 Output of interior-point algorithm in OR Courseware for Wyndor Glass Co. problem

Iteration	x_1	x_2	Z
0	1	2	13
1	1.27298	4	23.8189
2	1.37744	5	29.1323
3	1.56291	5.5	32.1887
4	1.80268	5.71816	33.9989
5	1.92134	5.82908	34.9094
6	1.96639	5.90595	35.429
7	1.98385	5.95199	35.7115
8	1.99197	5.97594	35.8556
9	1.99599	5.98796	35.9278
10	1.99799	5.99398	35.9639
11	1.999	5.99699	35.9819
12	1.9995	5.9985	35.991
13	1.99975	5.99925	35.9955
14	1.99987	5.99962	35.9977
15	1.99994	5.99981	35.9989

shown on this path are inside the boundary of the feasible region as the path approaches the optimal solution (2, 6). (All the subsequent trial solutions not shown also are inside the boundary of the feasible region.) Contrast this path with the path followed by the simplex method around the boundary of the feasible region from (0, 0) to (0, 6) to (2, 6).

Table 4.18 shows the actual output from IOR Tutorial for this problem.[24] (Try it yourself.) Note how the successive trial solutions keep getting closer and closer to the optimal solution, but never literally get there. However, the deviation becomes so infinitesimally small that the final trial solution can be taken to be the optimal solution for all practical purposes. (The Worked Examples section on the book's website shows the output from IOR Tutorial for **another example** as well.)

Section 7.4 presents the details of the specific interior-point algorithm that is implemented in IOR Tutorial.

Comparison with the Simplex Method

One meaningful way of comparing interior-point algorithms with the simplex method is to examine their theoretical properties regarding computational complexity. Karmarkar has proved that the original version of his algorithm is a **polynomial time algorithm;** i.e., the time required to solve *any* linear programming problem can be bounded above by a polynomial function of the size of the problem. Pathological counterexamples have been constructed to demonstrate that the simplex method does not possess this property, so it is an **exponential time algorithm** (i.e., the required time can be bounded above only by an exponential function of the problem size). This difference in *worst-case performance* is noteworthy. However, it tells us nothing about their comparison in average performance on real problems, which is the more crucial issue.

The two basic factors that determine the performance of an algorithm on a real problem are the *average computer time per iteration* and the *number of iterations*. Our next comparisons concern these factors.

[24]The procedure is called *Solve Automatically by the Interior-Point Algorithm*. The option menu provides two choices for a certain parameter of the algorithm α (defined in Sec. 7.4). The choice used here is the default value of $\alpha = 0.5$.

Interior-point algorithms are far more complicated than the simplex method. Considerably more extensive computations are required for each iteration to find the next trial solution. Therefore, the computer time per iteration for an interior-point algorithm is many times longer than that for the simplex method.

For fairly small problems, the numbers of iterations needed by an interior-point algorithm and by the simplex method tend to be somewhat comparable. For example, on a problem with 10 functional constraints, roughly 20 iterations would be typical for either kind of algorithm. Consequently, on problems of similar size, the total computer time for an interior-point algorithm will tend to be many times longer than that for the simplex method.

On the other hand, a key advantage of interior-point algorithms is that large problems do not require many more iterations than small problems. For example, a problem with 10,000 functional constraints probably will require well under 100 iterations. Even considering the very substantial computer time per iteration needed for a problem of this size, such a small number of iterations makes the problem quite tractable. By contrast, the simplex method might need 20,000 iterations and so might not finish within a reasonable amount of computer time. Therefore, interior-point algorithms often are faster than the simplex method for such huge problems.

The reason for this very large difference in the number of iterations on huge problems is the difference in the paths followed. At each iteration, the simplex method moves from the current CPF solution to an adjacent CPF solution along an edge on the boundary of the feasible region. Huge problems have an astronomical number of CPF solutions. The path from the initial CPF solution to an optimal solution may be a very circuitous one around the boundary, taking only a small step each time to the next adjacent CPF solution, so a huge number of steps may be required to reach an optimal solution. By contrast, an interior-point algorithm bypasses all this by shooting through the interior of the feasible region toward an optimal solution. Adding more functional constraints adds more constraint boundaries to the feasible region, but has little effect on the number of trial solutions needed on this path through the interior. This makes it possible for interior-point algorithms to solve problems with a huge number of functional constraints.

A final key comparison concerns the ability to perform the various kinds of postoptimality analysis described in Sec. 4.7. The simplex method and its extensions are very well suited to and are widely used for this kind of analysis. For example, an ILOG product called *Optimization Decision Manager* makes full use of the simplex method in CPLEX to perform a wide variety of postoptimality analysis tasks in convenient ways. Unfortunately, the interior-point approach currently has limited capability in this area.[25] Given the great importance of postoptimality analysis, this is a crucial drawback of interior-point algorithms. However, we point out next how the simplex method can be combined with the interior-point approach to overcome this drawback.

The Complementary Roles of the Simplex Method and the Interior-Point Approach

Ongoing research is continuing to provide substantial improvements in computer implementations of both the simplex method (including its variants) and interior-point algorithms. Therefore, any predictions about their future roles are risky. However, we do summarize below our current assessment of their complementary roles.

[25]However, research aimed at increasing this capability continues to make progress. For example, see E. A. Yildirim and M. J. Todd: "Sensitivity Analysis in Linear Programming and Semidefinite Programming Using Interior-Point Methods," *Mathematical Programming,* Series A, **90**(2): 229–261, April 2001.

The simplex method (and its variants) continues to be the standard algorithm for the routine use of linear programming. It continues to be the most efficient algorithm for problems with less than, say, 10,000 functional constraints. It also is the most efficient for some (but not all) problems with up to, say, 100,000 functional constraints and nearly an unlimited number of decision variables, so most users are continuing to use the simplex method for such problems. However, as the number of functional constraints increases even further, it becomes increasingly likely that an interior-point approach will be the most efficient, so it often is now used instead. As the size grows into the hundreds of thousands, or even millions, of functional constraints, the interior-point approach may be the only one capable of solving the problem. However, this certainly is not always the case. As mentioned in the preceding section, the latest state-of-the-art software is successfully using the simplex method and its variants to solve some truly massive problems with millions, or even tens of millions of functional constraints and decision variables.

These generalizations about how the interior-point approach and the simplex method should compare for various problem sizes will not hold across the board. The specific software packages and computer equipment being used have a major impact. The comparison also is affected considerably by the *specific type* of linear programming problem being solved. As time goes on, we should learn much more about how to identify specific types which are better suited for one kind of algorithm.

One of the by-products of the emergence of the interior-point approach has been a major renewal of efforts to improve the efficiency of computer implementations of the simplex method. As we indicated, impressive progress has been made in recent years, and more lies ahead. At the same time, ongoing research and development of the interior-point approach will further increase its power, and perhaps at a faster rate than for the simplex method.

Improving computer technology, such as massive parallel processing (a huge number of computer units operating in parallel on different parts of the same problem), also will substantially increase the size of problem that either kind of algorithm can solve. However, it now appears that the interior-point approach has much greater potential to take advantage of parallel processing than the simplex method does.

As discussed earlier, a key disadvantage of the interior-point approach is its limited capability for performing postoptimality analysis. To overcome this drawback, researchers have been developing procedures for switching over to the simplex method after an interior-point algorithm has finished. Recall that the trial solutions obtained by an interior-point algorithm keep getting closer and closer to an optimal solution (the best CPF solution), but never quite get there. Therefore, a switching procedure requires identifying a CPF solution (or BF solution after augmenting) that is very close to the final trial solution.

For example, by looking at Fig. 4.11, it is easy to see that the final trial solution in Table 4.18 is very near the CPF solution (2, 6). Unfortunately, on problems with thousands of decision variables (so no graph is available), identifying a nearby CPF (or BF) solution is a very challenging and time-consuming task. However, good progress has been made in developing procedures to do this. For example, the full-fledged professional version of CPLEX includes a *crossover algorithm* which converts the solutions obtained by its "barrier algorithm" into a BF solution.

Once this nearby BF solution has been found, the optimality test for the simplex method is applied to check whether this actually is the optimal BF solution. If it is not optimal, some iterations of the simplex method are conducted to move from this BF solution to an optimal solution. Generally, only a very few iterations (perhaps one) are needed because the interior-point algorithm has brought us so close to an optimal solution. Therefore, these iterations should be done quite quickly, even on problems that are too huge to be solved from scratch. After an optimal solution is actually reached, the simplex method and its variants are applied to help perform postoptimality analysis.

Because of the difficulties involved in applying a switching procedure (including the extra computer time), some practitioners prefer to just use the simplex method from the outset. This makes good sense when you only occasionally encounter problems that are large enough for an interior-point algorithm to be modestly faster (before switching) than the simplex method. This modest speed-up would not justify both the extra computer time for a switching procedure and the high cost of acquiring (and learning to use) a software package based on the interior-point approach. However, for organizations which frequently must deal with extremely large linear programming problems, acquiring a state-of-the-art software package of this kind (including a switching procedure) probably is worthwhile. For sufficiently huge problems, the only available way of solving them may be with such a package.

Applications of huge linear programming models sometimes lead to savings of millions of dollars. Just one such application can pay many times over for a state-of-the-art software package based on the interior-point approach plus switching over to the simplex method at the end.

4.10 CONCLUSIONS

The simplex method is an efficient and reliable algorithm for solving linear programming problems. It also provides the basis for performing the various parts of postoptimality analysis very efficiently.

Although it has a useful geometric interpretation, the simplex method is an algebraic procedure. At each iteration, it moves from the current BF solution to a better, adjacent BF solution by choosing both an entering basic variable and a leaving basic variable and then using Gaussian elimination to solve a system of linear equations. When the current solution has no adjacent BF solution that is better, the current solution is optimal and the algorithm stops.

We presented the full algebraic form of the simplex method to convey its logic, and then we streamlined the method to a more convenient tabular form. To set up for starting the simplex method, it is sometimes necessary to use artificial variables to obtain an initial BF solution for an artificial problem. If so, either the Big M method or the two-phase method is used to ensure that the simplex method obtains an optimal solution for the real problem.

Computer implementations of the simplex method and its variants have become so powerful that they now are frequently used to solve linear programming problems with many hundreds of thousands of functional constraints and decision variables, and occasionally vastly larger problems. Interior-point algorithms also provide a powerful tool for solving very large problems.

APPENDIX 4.1 AN INTRODUCTION TO USING LINDO AND LINGO

The LINGO software can accept optimization models in either of two styles or syntax: (a) LINDO syntax or (b) LINGO syntax. We will first describe LINDO syntax. The relative advantages of LINDO syntax are that it is very easy and natural for simple linear and integer programming problems. It has been in wide use since 1981.

The LINDO syntax allows you to enter a model in a natural form, essentially as presented in a textbook. For example, here is how the Wyndor Glass Co. example introduced in Sec. 3.1. is entered. Presuming you have installed LINGO, you click on the LINGO icon to start up LINGO and then immediately type the following:

```
! Wyndor Glass Co. Problem. LINDO model
! X1 = batches of product 1 per week
! X2 = batches of product 2 per week
! Profit, in 1000 of dollars,
MAX  Profit) 3 X1 + 5 X2
Subject to
! Production time
Plant1) X1 <= 4
Plant2) 2 X2 <= 12
Plant3) 3 X1 + 2 X2 <= 18
END
```

The first four lines, each starting with an exclamation point at the beginning, are simply comments. The comment on the fourth line further clarifies that the objective function is expressed in units of thousands of dollars. The number 1000 in this comment does not have the usual comma in front of the last three digits because LINDO/LINGO does not accept commas. (LINDO syntax also does not accept parentheses in algebraic expressions.) Lines five onward specify the model. The decision variables can be either lowercase or uppercase. Uppercase usually is used so the variables won't be dwarfed by the following "subscripts." Instead of X1 or X2, you may use more suggestive names, such as the name of the product being produced; e.g., DOORS and WINDOWS, to represent the decision variable throughout the model.

The fifth line of the LINDO formulation indicates that the objective of the model is to maximize the objective function, $3x_1 + 5x_2$. The word Profit followed by a parenthesis is optional. It clarifies that the quantity being maximized is to be called Profit on the solution report.

The comment on the seventh line points out that the following constraints are on the production times being used. The next three lines start by giving a name (again, optional, followed by a parenthesis) for each of the functional constraints. These constraints are written in the usual way except for the inequality signs. Because most keyboards do not include $\leq$ and $\geq$ signs, LINDO interprets either $<$ or $<=$ as $\leq$ and either $>$ or $>=$ as $\geq$. (On keyboards that include $\leq$ and $\geq$ signs, LINDO will not recognize them.)

The end of the constraints is signified by the word END. No nonnegativity constraints are stated because LINDO automatically assumes that all variables are ≥ 0. If, say, x_1 had not had a nonnegativity constraint, this would be indicated by typing FREE X1 on the next line below END.

To solve this model in LINGO/LINDO, click on the red Bull's Eye solve button at the top of the LINGO window. Figure A4.1 shows the resulting "solution report." The top lines indicate that the best overall, or "global," solution has been found, with an objective function value of 36, in two iterations. Next come the values for x_1 and x_2 for the optimal solution.

FIGURE A4.1
The solution report provided by LINDO syntax for the Wyndor Glass Co. problem.

```
Global optimal solution found.
Objective value:                              36.00000
Total solver iterations:                             2

Variable     Value      Reduced Cost
   X1      2.000000      0.000000
   X2      6.000000      0.000000

  Row      Slack or Surplus    Dual  Price
PROFIT        36.00000          1.000000
PLANT1        2.000000          0.000000
PLANT2        0.000000          1.500000
PLANT3        0.000000          1.000000
```

The column to the right of the Values column gives the **reduced costs.** We have not discussed reduced costs in this chapter because the information they provide can also be gleaned from the *allowable range* for the coefficients in the objective function. These allowable ranges are readily available (as you will see in the next figure). When the variable is a *basic variable* in the optimal solution (as for both variables in the Wyndor problem), its reduced cost automatically is 0. When the variable is a *nonbasic variable*, its reduced cost provides some interesting information. A variable whose objective coefficient is "too small" in a maximizing model or "too large" in a minimizing model will have a value of 0 in an optimal solution. The reduced cost indicates how much this coefficient needs to be *increased* (when maximizing) or *decreased* (when minimizing) before the optimal solution would change and this variable would become a basic variable. However, recall that this same information already is available from the allowable range for the coefficient of this variable in the objective function. The reduced cost (for a nonbasic variable) is just the *allowable increase* (when maximizing) from the current value of this coefficient to remain within its allowable range or the *allowable decrease* (when minimizing).

The bottom portion of Fig. A.4.1 provides information about the three functional constraints. The Slack or Surplus column gives the difference between the two sides of each constraint. The Dual Price column gives, by another name, the *shadow prices* discussed in Sec. 4.7 for these constraints. (This alternate name comes from the fact found in Sec. 6.1 that these shadow prices are just the optimal values of the dual variables introduced in Chap. 6.) Be aware, however, that LINDO uses a different sign convention from the common one adopted elsewhere in this text (see footnote 19 regarding the definition of shadow price in Sec. 4.7). In particular, for minimization problems, LINGO/LINDO shadow prices (dual prices) are the negative of ours.

After LINDO provides you with the solution report, you also have the option to do range (sensitivity) analysis. Fig. A4.2 shows the range report, which is generated by clicking on: LINGO | Range.

Except for using units of thousand of dollars instead of dollars for the coefficients in the objective function, this report is identical to the last three columns of the table in the sensitivity report generated by the Excel Solver, as shown earlier in Fig. 4.10. Thus, as already discussed in Sec. 4.7, the first two rows of numbers in this range report indicate that the allowable range for each coefficient in the objective function (assuming no other change in the model) is

$$0 \leq c_1 \leq 7.5$$
$$2 \leq c_2$$

Similarly, the last three rows indicate that the allowable range for each right-hand side (assuming no other change in the model) is

$$2 \leq b_1$$
$$6 \leq b_2 \leq 18$$
$$12 \leq b_3 \leq 24$$

You can print the results in standard Windows fashion by clicking on Files | Print.

■ **FIGURE A4.2**
Range report provided by LINDO for the Wyndor Glass Co. problem.

```
Ranges in which the basis is unchanged:
              Objective Coefficient Ranges
               Current    Allowable   Allowable
Variable    Coefficient   Increase     Decrease
   X1        3.000000     4.500000     3.000000
   X2        5.000000     INFINITY     3.000000

              Righthand Side Ranges
  Row        Current     Allowable    Allowable
               RHS        Increase     Decrease
PLANT1      4.000000     INFINITY     2.000000
PLANT2     12.000000     6.000000     6.000000
PLANT3     18.000000     6.000000     6.000000
```

These are the basics for getting started with LINGO/LINDO. You can turn on or turn off the generation of reports. For example, if the automatic generation of the standard solution report has been turned off (Terse mode), you can turn it back on by clicking on: LINGO | Options | Interface | Output level | Verbose | Apply. The ability to generate range reports can be turned on or off by clicking on: LINGO | Options | General solver | Dual computations | Prices & Ranges | Apply.

The second input style that LINGO supports is LINGO syntax. LINGO syntax is dramatically more powerful than LINDO syntax. The advantages to using LINGO syntax are: (a) it allows arbitrary mathematical expressions, including parentheses and all familiar mathematical operators such as division, multiplication, log, sin, etc., (b) the ability to solve not just linear programming problems but also nonlinear programming problems, (c) scalability to large applications using subscripted variables and sets, (d) the ability to read input data from a spreadsheet or database and send solution information back into a spreadsheet or database, (e) the ability to naturally represent sparse relationships, (f) programming ability so that you can solve a series of models automatically as when doing parametric analysis. A formulation of the Wyndor problem in LINGO, using the subscript/sets feature is:

```
! Wyndor Glass Co. Problem;

SETS:
 PRODUCT: PPB, X;             ! Each product has a profit/batch
and amount;
 RESOURCE: HOURSAVAILABLE;    ! Each resource has a capacity;
! Each resource product combination has an hours/batch;
 RXP(RESOURCE,PRODUCT): HPB;
ENDSETS
DATA:
 PRODUCT  = DOORS  WINDOWS;      ! The products;
     PPB =    3       5;         ! Profit per batch;

       RESOURCE = PLANT1 PLANT2 PLANT3;
 HOURSAVAILABLE =    4      12     18;

  HPB =    1  0    ! Hours per batch;
           0  2
           3  2;
ENDDATA
  ! Sum over all products j the profit per batch times batches
produced;
 MAX = @SUM( PRODUCT(j): PPB(j)*X(j));

 @FOR( RESOURCE(i)):  ! For each resource i...;
    ! Sum over all products j of hours per batch time batches
produced...;
   @SUM(RXP(i,j): HPB(i,j)*X(j)) <= HOURSAVAILABLE(i);
     );
```

The original Wyndor problem has two products and three resources. If Wyndor expands to having four products and five resources, it is a trivial change to insert the appropriate new data into the DATA section. The formulation of the model adjusts automatically. The subscript/sets capability also allows one to naturally represent three dimensional or higher models. The large problem described in Sec. 3.6 has five dimensions: plants, machines, products, regions/customers, and time periods. This would be hard to fit into a two-dimensional spreadsheet but is easy to represent in a modeling language with sets and subscripts. In practice, for problems like that in Sec. 3.6, many of the 10(10)(10)(10)(10) = 100,000 possible combinations of relationships do not exist; e.g., not all plants can make all products, and not all customers demand all products. The subscript/sets capability in modeling languages make it easy to represent such sparse relationships.

For most models that you enter, LINGO will be able to detect automatically whether you are using LINDO syntax or LINGO syntax. You may choose your default syntax by clicking on: LINGO | Options | Interface | File format | lng (for LINGO) or ltx (for LINDO).

LINGO includes an extensive online Help menu to give more details and examples. Supplements 1 and 2 to Chapter 3 (shown on the book's website) provide a relatively complete introduction to LINGO. The LINGO tutorial on the website also provides additional details. The LINGO/LINDO files on the website for various chapters show LINDO/LINGO formulations for numerous examples from most of the chapters.

SELECTED REFERENCES

1. Bixby, R. E.: "Solving Real-World Linear Programs: A Decade and More of Progress," *Operations Research,* **50**(1): 3–15, January–February 2002.
2. Dantzig, G. B., and M. N. Thapa: *Linear Programming 1: Introduction,* Springer, New York, 1997.
3. Fourer, R.: "Software Survey: Linear Programming," *OR/MS Today,* June 2007, pp. 42–51.
4. Luenberger, D., and Y. Ye: *Linear and Nonlinear Programming*, 3rd ed., Springer, New York, 2008.
5. Maros, I.: *Computational Techniques of the Simplex Method,* Kluwer Academic Publishers (now Springer), Boston, MA, 2003.
6. Schrage, L.: *Optimization Modeling with LINGO*, LINDO Systems, Chicago, 2008.
7. Tretkoff, C., and I. Lustig: "New Age of Optimization Applications," *OR/MS Today*, December 2006, pp. 46–49.
8. Vanderbei, R. J.: *Linear Programming: Foundations and Extensions,* 3rd ed., Springer, New York, 2008.

LEARNING AIDS FOR THIS CHAPTER ON OUR WEBSITE (www.mhhe.com/hillier)

Worked Examples:

Examples for Chapter 4

Demonstration Examples in OR Tutor:

Interpretation of the Slack Variables
Simplex Method—Algebraic Form
Simplex Method—Tabular Form

Interactive Procedures in IOR Tutorial:

Enter or Revise a General Linear Programming Model
Set Up for the Simplex Method—Interactive Only
Solve Interactively by the Simplex Method
Interactive Graphical Method

Automatic Procedures in IOR Tutorial:

Solve Automatically by the Simplex Method
Solve Automatically by the Interior-Point Algorithm
Graphical Method and Sensitivity Analysis

An Excel Add-In:

Premium Solver for Education

Files (Chapter 3) for Solving the Wyndor and Radiation Therapy Examples:

Excel Files
LINGO/LINDO File
MPL/CPLEX File

Glossary for Chapter 4

See Appendix 1 for documentation of the software.

PROBLEMS

The symbols to the left of some of the problems (or their parts) have the following meaning:

D: The corresponding demonstration example listed on the preceding page may be helpful.
I: We suggest that you use the corresponding interactive procedure listed on the preceding page (the printout records your work).
C: Use the computer with any of the software options available to you (or as instructed by your instructor) to solve the problem automatically. (See Sec. 4.8 for a listing of the options featured in this book and on the book's website.)

An asterisk on the problem number indicates that at least a partial answer is given in the back of the book.

4.1-1. Consider the following problem.

Maximize $\quad Z = x_1 + 2x_2,$

subject to

$$\begin{aligned} x_1 \quad\quad &\le 5 \\ x_2 &\le 6 \\ x_1 + x_2 &\le 8 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0.$$

(a) Plot the feasible region and circle all the CPF solutions.
(b) For each CPF solution, identify the pair of constraint boundary equations that it satisfies.
(c) For each CPF solution, use this pair of constraint boundary equations to solve algebraically for the values of x_1 and x_2 at the corner point.
(d) For each CPF solution, identify its adjacent CPF solutions.
(e) For each pair of adjacent CPF solutions, identify the constraint boundary they share by giving its equation.

4.1-2. Consider the following problem.

Maximize $\quad Z = 3x_1 + 2x_2,$

subject to

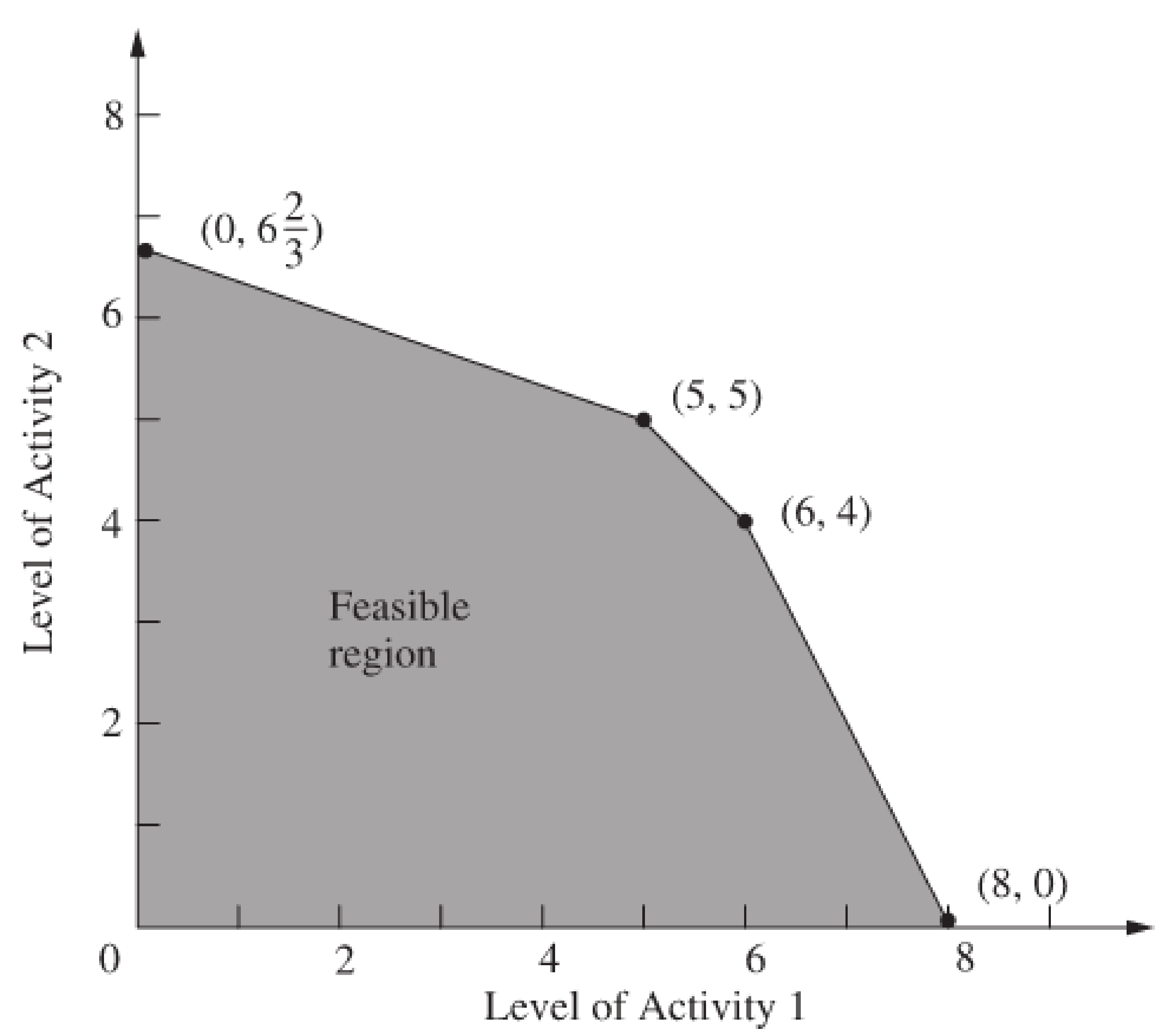
$$\begin{aligned} 2x_1 + x_2 &\le 6 \\ x_1 + 2x_2 &\le 6 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0.$$

D,I **(a)** Use the graphical method to solve this problem. Circle all the corner points on the graph.
(b) For each CPF solution, identify the pair of constraint boundary equations it satisfies.
(c) For each CPF solution, identify its adjacent CPF solutions.
(d) Calculate Z for each CPF solution. Use this information to identify an optimal solution.
(e) Describe graphically what the simplex method does step by step to solve the problem.

4.1-3. A certain linear programming model involving two activities has the feasible region shown below.



The objective is to maximize the total profit from the two activities. The unit profit for activity 1 is \$1,000 and the unit profit for activity 2 is \$2,000.

(a) Calculate the total profit for each CPF solution. Use this information to find an optimal solution.
(b) Use the solution concepts of the simplex method given in Sec. 4.1 to identify the sequence of CPF solutions that would be examined by the simplex method to reach an optimal solution.

4.1-4.* Consider the linear programming model (given in the back of the book) that was formulated for Prob. 3.2-3.
(a) Use graphical analysis to identify all the *corner-point solutions* for this model. Label each as either feasible or infeasible.
(b) Calculate the value of the objective function for each of the CPF solutions. Use this information to identify an optimal solution.
(c) Use the solution concepts of the simplex method given in Sec. 4.1 to identify which sequence of CPF solutions might be examined by the simplex method to reach an optimal solution. (*Hint*: There are *two* alternative sequences to be identified for this particular model.)

4.1-5. Repeat Prob. 4.1-4 for the following problem.

$$\text{Maximize} \quad Z = x_1 + 2x_2,$$

subject to

$$\begin{aligned} x_1 + 3x_2 &\le 8 \\ x_1 + x_2 &\le 4 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0.$$

4.1-6. Describe graphically what the simplex method does step by step to solve the following problem.

$$\text{Maximize} \quad Z = 2x_1 + 3x_2,$$

subject to

$$\begin{aligned} -3x_1 + x_2 &\le 1 \\ 4x_1 + 2x_2 &\le 20 \\ 4x_1 - x_2 &\le 10 \\ -x_1 + 2x_2 &\le 5 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0.$$

4.1-7. Describe graphically what the simplex method does step by step to solve the following problem.

$$\text{Minimize} \quad Z = 5x_1 + 7x_2,$$

subject to

$$\begin{aligned} 2x_1 + 3x_2 &\ge 147 \\ 3x_1 + 4x_2 &\ge 210 \\ x_1 + x_2 &\ge 63 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0.$$

4.1-8. Label each of the following statements about linear programming problems as true or false, and then justify your answer.
(a) For minimization problems, if the objective function evaluated at a CPF solution is no larger than its value at every adjacent CPF solution, then that solution is optimal.
(b) Only CPF solutions can be optimal, so the number of optimal solutions cannot exceed the number of CPF solutions.
(c) If multiple optimal solutions exist, then an optimal CPF solution may have an adjacent CPF solution that also is optimal (the same value of Z).

4.1-9. The following statements give inaccurate paraphrases of the six solution concepts presented in Sec. 4.1. In each case, explain what is wrong with the statement.
(a) The best CPF solution always is an optimal solution.
(b) An iteration of the simplex method checks whether the current CPF solution is optimal and, if not, moves to a new CPF solution.
(c) Although any CPF solution can be chosen to be the initial CPF solution, the simplex method always chooses the origin.
(d) When the simplex method is ready to choose a new CPF solution to move to from the current CPF solution, it only considers adjacent CPF solutions because one of them is likely to be an optimal solution.
(e) To choose the new CPF solution to move to from the current CPF solution, the simplex method identifies all the adjacent CPF solutions and determines which one gives the largest rate of improvement in the value of the objective function.

4.2-1. Reconsider the model in Prob. 4.1-4.
(a) Introduce slack variables in order to write the functional constraints in augmented form.
(b) For each CPF solution, identify the corresponding BF solution by calculating the values of the slack variables. For each BF solution, use the values of the variables to identify the nonbasic variables and the basic variables.
(c) For each BF solution, demonstrate (by plugging in the solution) that, after the nonbasic variables are set equal to zero, this BF solution also is the simultaneous solution of the system of equations obtained in part (*a*).

4.2-2. Reconsider the model in Prob. 4.1-5. Follow the instructions of Prob. 4.2-1 for parts (*a*), (*b*), and (*c*).
(d) Repeat part (*b*) for the corner-point infeasible solutions and the corresponding basic infeasible solutions.
(e) Repeat part (*c*) for the basic infeasible solutions.

4.3-1. Read the referenced article that fully describes the OR study summarized in the application vignette presented in Sec. 4.3. Briefly describe the application of the simplex method in this study. Then list the various financial and nonfinancial benefits that resulted from this study.

D,I **4.3-2.** Work through the simplex method (in algebraic form) step by step to solve the model in Prob. 4.1-4.

4.3-3. Reconsider the model in Prob. 4.1-5.
(a) Work through the simplex method (in algebraic form) *by hand* to solve this model.

D,I **(b)** Repeat part (*a*) with the corresponding interactive routine in your IOR Tutorial.

C **(c)** Verify the optimal solution you obtained by using a software package based on the simplex method.

D,I **4.3-4.*** Work through the simplex method (in algebraic form) step by step to solve the following problem.

$$\text{Maximize} \quad Z = 4x_1 + 3x_2 + 6x_3,$$

subject to

$$\begin{aligned} 3x_1 + x_2 + 3x_3 &\le 30 \\ 2x_1 + 2x_2 + 3x_3 &\le 40 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

D,I **4.3-5.** Work through the simplex method (in algebraic form) step by step to solve the following problem.

$$\text{Maximize} \quad Z = 3x_1 + 4x_2 + 5x_3,$$

subject to

$$\begin{aligned} 3x_1 + x_2 + 5x_3 &\le 150 \\ x_1 + 4x_2 + x_3 &\le 120 \\ 2x_1 + 2x_3 &\le 105 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

4.3-6. Consider the following problem.

$$\text{Maximize} \quad Z = 5x_1 + 3x_2 + 4x_3,$$

subject to

$$\begin{aligned} 2x_1 + x_2 + x_3 &\le 20 \\ 3x_1 + x_2 + 2x_3 &\le 30 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

You are given the information that the *nonzero* variables in the optimal solution are x_2 and x_3.

(a) Describe how you can use this information to adapt the simplex method to solve this problem in the minimum possible number of iterations (when you start from the usual initial BF solution). Do *not* actually perform any iterations.

(b) Use the procedure developed in part (*a*) to solve this problem by hand. (Do *not* use your OR Courseware.)

4.3-7. Consider the following problem.

$$\text{Maximize} \quad Z = 2x_1 + 4x_2 + 3x_3,$$

subject to

$$\begin{aligned} x_1 + 3x_2 + 2x_3 &\le 30 \\ x_1 + x_2 + x_3 &\le 24 \\ 3x_1 + 5x_2 + 3x_3 &\le 60 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

You are given the information that $x_1 > 0$, $x_2 = 0$, and $x_3 > 0$ in the optimal solution.

(a) Describe how you can use this information to adapt the simplex method to solve this problem in the minimum possible number of iterations (when you start from the usual initial BF solution). Do *not* actually perform any iterations.

(b) Use the procedure developed in part (*a*) to solve this problem by hand. (Do *not* use your OR Courseware.)

4.3-8. Label each of the following statements as true or false, and then justify your answer by referring to specific statements in the chapter.

(a) The simplex method's rule for choosing the entering basic variable is used because it always leads to the *best* adjacent BF solution (largest Z).

(b) The simplex method's minimum ratio rule for choosing the leaving basic variable is used because making another choice with a larger ratio would yield a basic solution that is not feasible.

(c) When the simplex method solves for the next BF solution, elementary algebraic operations are used to eliminate each nonbasic variable from all but one equation (*its* equation) and to give it a coefficient of $+1$ in that one equation.

D,I **4.4-1.** Repeat Prob. 4.3-2, using the tabular form of the simplex method.

D,I,C **4.4-2.** Repeat Prob. 4.3-3, using the tabular form of the simplex method.

4.4-3. Consider the following problem.

$$\text{Maximize} \quad Z = 2x_1 + x_2,$$

subject to

$$\begin{aligned} x_1 + x_2 &\le 40 \\ 4x_1 + x_2 &\le 100 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0.$$

(a) Solve this problem graphically in a freehand manner. Also identify all the CPF solutions.

D,I **(b)** Now use IOR Tutorial to solve the problem graphically.

D **(c)** Use hand calculations to solve this problem by the simplex method in algebraic form.

D,I **(d)** Now use IOR Tutorial to solve this problem interactively by the simplex method in algebraic form.

D **(e)** Use hand calculations to solve this problem by the simplex method in tabular form.

D,I **(f)** Now use IOR Tutorial to solve this problem interactively by the simplex method in tabular form.

C **(g)** Use a software package based on the simplex method to solve the problem.

4.4-4. Repeat Prob. 4.4-3 for the following problem.

Maximize $Z = 2x_1 + 3x_2$,

subject to

$$x_1 + 2x_2 \le 30$$
$$x_1 + x_2 \le 20$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

4.4-5. Consider the following problem.

Maximize $Z = 5x_1 + 9x_2 + 7x_3$,

subject to

$$x_1 + 3x_2 + 2x_3 \le 10$$
$$3x_1 + 4x_2 + 2x_3 \le 12$$
$$2x_1 + x_2 + 2x_3 \le 8$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0.$$

D,I **(a)** Work through the simplex method step by step in algebraic form.
D,I **(b)** Work through the simplex method step by step in tabular form.
C **(c)** Use a software package based on the simplex method to solve the problem.

4.4-6. Consider the following problem.

Maximize $Z = 3x_1 + 5x_2 + 6x_3$,

subject to

$$2x_1 + x_2 + x_3 \le 4$$
$$x_1 + 2x_2 + x_3 \le 4$$
$$x_1 + x_2 + 2x_3 \le 4$$
$$x_1 + x_2 + x_3 \le 3$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0.$$

D,I **(a)** Work through the simplex method step by step in algebraic form.
D,I **(b)** Work through the simplex method in tabular form.
C **(c)** Use a computer package based on the simplex method to solve the problem.

D,I **4.4-7.** Work through the simplex method step by step (in tabular form) to solve the following problem.

Maximize $Z = 2x_1 - x_2 + x_3$,

subject to

$$3x_1 + x_2 + x_3 \le 6$$
$$x_1 - x_2 + 2x_3 \le 1$$
$$x_1 + x_2 - x_3 \le 2$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0.$$

D,I **4.4-8.** Work through the simplex method step by step to solve the following problem.

Maximize $Z = -x_1 + x_2 + 2x_3$,

subject to

$$x_1 + 2x_2 - x_3 \le 20$$
$$-2x_1 + 4x_2 + 2x_3 \le 60$$
$$2x_1 + 3x_2 + x_3 \le 50$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0.$$

4.5-1. Consider the following statements about linear programming and the simplex method. Label each statement as true or false, and then justify your answer.

(a) In a particular iteration of the simplex method, if there is a tie for which variable should be the leaving basic variable, then the next BF solution must have at least one basic variable equal to zero.
(b) If there is no leaving basic variable at some iteration, then the problem has no feasible solutions.
(c) If at least one of the basic variables has a coefficient of zero in row 0 of the final tableau, then the problem has multiple optimal solutions.
(d) If the problem has multiple optimal solutions, then the problem must have a bounded feasible region.

4.5-2. Suppose that the following constraints have been provided for a linear programming model with decision variables x_1 and x_2.

$$-2x_1 + 3x_2 \le 12$$
$$-3x_1 + 2x_2 \le 12$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

(a) Demonstrate graphically that the feasible region is unbounded.
(b) If the objective is to maximize $Z = -x_1 + x_2$, does the model have an optimal solution? If so, find it. If not, explain why not.
(c) Repeat part (*b*) when the objective is to maximize $Z = x_1 - x_2$.
(d) For objective functions where this model has no optimal solution, does this mean that there are no good solutions according to the model? Explain. What probably went wrong when formulating the model?
D,I **(e)** Select an objective function for which this model has no optimal solution. Then work through the simplex method step by step to demonstrate that Z is unbounded.
C **(f)** For the objective function selected in part (*e*), use a software package based on the simplex method to determine that Z is unbounded.

4.5-3. Follow the instructions of Prob. 4.5-2 when the constraints are the following:

$$2x_1 - x_2 \le 20$$
$$x_1 - 2x_2 \le 20$$

and

$$x_1 \geq 0, \qquad x_2 \geq 0.$$

D,I **4.5-4.** Consider the following problem.

$$\text{Maximize} \qquad Z = 5x_1 + x_2 + 3x_3 + 4x_4,$$

subject to

$$\begin{aligned} x_1 - 2x_2 + 4x_3 + 3x_4 &\leq 20 \\ -4x_1 + 6x_2 + 5x_3 - 4x_4 &\leq 40 \\ 2x_1 - 3x_2 + 3x_3 + 8x_4 &\leq 50 \end{aligned}$$

and

$$x_1 \geq 0, \qquad x_2 \geq 0, \qquad x_3 \geq 0, \qquad x_4 \geq 0.$$

Work through the simplex method step by step to demonstrate that Z is unbounded.

4.5-5. A basic property of any linear programming problem with a bounded feasible region is that every feasible solution can be expressed as a convex combination of the CPF solutions (perhaps in more than one way). Similarly, for the augmented form of the problem, every feasible solution can be expressed as a convex combination of the BF solutions.

(a) Show that *any* convex combination of *any* set of feasible solutions must be a feasible solution (so that any convex combination of CPF solutions must be feasible).

(b) Use the result quoted in part (*a*) to show that any convex combination of BF solutions must be a feasible solution.

4.5-6. Using the facts given in Prob. 4.5-5, show that the following statements must be true for any linear programming problem that has a bounded feasible region and multiple optimal solutions:

(a) Every convex combination of the optimal BF solutions must be optimal.

(b) No other feasible solution can be optimal.

4.5-7. Consider a two-variable linear programming problem whose CPF solutions are (0, 0), (6, 0), (6, 3), (3, 3), and (0, 2). (See Prob. 3.2-2 for a graph of the feasible region.)

(a) Use the graph of the feasible region to identify all the constraints for the model.

(b) For each pair of adjacent CPF solutions, give an example of an objective function such that all the points on the line segment between these two corner points are multiple optimal solutions.

(c) Now suppose that the objective function is $Z = -x_1 + 2x_2$. Use the graphical method to find all the optimal solutions.

D,I **(d)** For the objective function in part (*c*), work through the simplex method step by step to find all the optimal BF solutions. Then write an algebraic expression that identifies all the optimal solutions.

D,I **4.5-8.** Consider the following problem.

$$\text{Maximize} \qquad Z = 50x_1 + 25x_2 + 20x_3 + 40x_4,$$

subject to

$$\begin{aligned} 2x_1 + x_2 &\leq 30 \\ x_3 + 2x_4 &\leq 20 \end{aligned}$$

and

$$x_j \geq 0, \qquad \text{for } j = 1, 2, 3, 4.$$

Work through the simplex method step by step to find *all* the optimal BF solutions.

4.6-1.* Consider the following problem.

$$\text{Maximize} \qquad Z = 2x_1 + 3x_2,$$

subject to

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ x_1 + x_2 &= 3 \end{aligned}$$

and

$$x_1 \geq 0, \qquad x_2 \geq 0.$$

D,I **(a)** Solve this problem graphically.

(b) Using the Big M method, construct the complete first simplex tableau for the simplex method and identify the corresponding initial (artificial) BF solution. Also identify the initial entering basic variable and the leaving basic variable.

I **(c)** Continue from part (*b*) to work through the simplex method step by step to solve the problem.

4.6-2. Consider the following problem.

$$\text{Maximize} \qquad Z = 4x_1 + 2x_2 + 3x_3 + 5x_4,$$

subject to

$$\begin{aligned} 2x_1 + 3x_2 + 4x_3 + 2x_4 &= 300 \\ 8x_1 + x_2 + x_3 + 5x_4 &= 300 \end{aligned}$$

and

$$x_j \geq 0, \qquad \text{for } j = 1, 2, 3, 4.$$

(a) Using the Big M method, construct the complete first simplex tableau for the simplex method and identify the corresponding initial (artificial) BF solution. Also identify the initial entering basic variable and the leaving basic variable.

I **(b)** Work through the simplex method step by step to solve the problem.

(c) Using the two-phase method, construct the complete first simplex tableau for phase 1 and identify the corresponding initial (artificial) BF solution. Also identify the initial entering basic variable and the leaving basic variable.

I **(d)** Work through phase 1 step by step.

(e) Construct the complete first simplex tableau for phase 2.

I **(f)** Work through phase 2 step by step to solve the problem.

(g) Compare the sequence of BF solutions obtained in part (*b*) with that in parts (*d*) and (*f*). Which of these solutions are feasible only for the artificial problem obtained by introducing artificial variables and which are actually feasible for the real problem?

C **(h)** Use a software package based on the simplex method to solve the problem.

4.6-3.* Consider the following problem.

$$\text{Minimize} \qquad Z = 2x_1 + 3x_2 + x_3,$$

subject to

$$\begin{aligned} x_1 + 4x_2 + 2x_3 &\ge 8 \\ 3x_1 + 2x_2 &\ge 6 \end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0.$$

(a) Reformulate this problem to fit our standard form for a linear programming model presented in Sec. 3.2.

I **(b)** Using the Big M method, work through the simplex method step by step to solve the problem.

I **(c)** Using the two-phase method, work through the simplex method step by step to solve the problem.

(d) Compare the sequence of BF solutions obtained in parts (b) and (c). Which of these solutions are feasible only for the artificial problem obtained by introducing artificial variables and which are actually feasible for the real problem?

C **(e)** Use a software package based on the simplex method to solve the problem.

4.6-4. For the Big M method, explain why the simplex method never would choose an artificial variable to be an entering basic variable once all the artificial variables are nonbasic.

4.6-5. Consider the following problem.

Maximize $\quad Z = 5x_1 + 4x_2,$

subject to

$$\begin{aligned} 3x_1 + 2x_2 &\le 6 \\ 2x_1 - x_2 &\ge 6 \end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

(a) Demonstrate graphically that this problem has no feasible solutions.

C **(b)** Use a computer package based on the simplex method to determine that the problem has no feasible solutions.

I **(c)** Using the Big M method, work through the simplex method step by step to demonstrate that the problem has no feasible solutions.

I **(d)** Repeat part (c) when using phase 1 of the two-phase method.

4.6-6. Follow the instructions of Prob. 4.6-5 for the following problem.

Minimize $\quad Z = 5{,}000x_1 + 7{,}000x_2,$

subject to

$$\begin{aligned} -2x_1 + x_2 &\ge 1 \\ x_1 - 2x_2 &\ge 1 \end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0.$$

4.6-7. Consider the following problem.

Maximize $\quad Z = 2x_1 + 5x_2 + 3x_3,$

subject to

$$\begin{aligned} x_1 - 2x_2 + x_3 &\ge 20 \\ 2x_1 + 4x_2 + x_3 &= 50 \end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0.$$

(a) Using the Big M method, construct the complete first simplex tableau for the simplex method and identify the corresponding initial (artificial) BF solution. Also identify the initial entering basic variable and the leaving basic variable.

I **(b)** Work through the simplex method step by step to solve the problem.

I **(c)** Using the two-phase method, construct the complete first simplex tableau for phase 1 and identify the corresponding initial (artificial) BF solution. Also identify the initial entering basic variable and the leaving basic variable.

I **(d)** Work through phase 1 step by step.

(e) Construct the complete first simplex tableau for phase 2.

I **(f)** Work through phase 2 step by step to solve the problem.

(g) Compare the sequence of BF solutions obtained in part (b) with that in parts (d) and (f). Which of these solutions are feasible only for the artificial problem obtained by introducing artificial variables and which are actually feasible for the real problem?

C **(h)** Use a software package based on the simplex method to solve the problem.

4.6-8. Consider the following problem.

Minimize $\quad Z = 2x_1 + x_2 + 3x_3,$

subject to

$$\begin{aligned} 5x_1 + 2x_2 + 7x_3 &= 420 \\ 3x_1 + 2x_2 + 5x_3 &\ge 280 \end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0.$$

I **(a)** Using the two-phase method, work through phase 1 step by step.

C **(b)** Use a software package based on the simplex method to formulate and solve the phase 1 problem.

I **(c)** Work through phase 2 step by step to solve the original problem.

C **(d)** Use a computer code based on the simplex method to solve the original problem.

4.6-9.* Consider the following problem.

Minimize $\quad Z = 3x_1 + 2x_2 + 4x_3,$

subject to

$$\begin{aligned} 2x_1 + x_2 + 3x_3 &= 60 \\ 3x_1 + 3x_2 + 5x_3 &\ge 120 \end{aligned}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0.$$

I **(a)** Using the Big *M* method, work through the simplex method step by step to solve the problem.
I **(b)** Using the two-phase method, work through the simplex method step by step to solve the problem.
(c) Compare the sequence of BF solutions obtained in parts (*a*) and (*b*). Which of these solutions are feasible only for the artificial problem obtained by introducing artificial variables and which are actually feasible for the real problem?
C **(d)** Use a software package based on the simplex method to solve the problem.

4.6-10. Follow the instructions of Prob. 4.6-9 for the following problem.

Minimize $Z = 3x_1 + 2x_2 + 7x_3$,

subject to

$$\begin{aligned} -x_1 + x_2 &= 10 \\ 2x_1 - x_2 + x_3 &\geq 10 \end{aligned}$$

and

$$x_1 \geq 0, \qquad x_2 \geq 0, \qquad x_3 \geq 0.$$

4.6-11. Label each of the following statements as true or false, and then justify your answer.
(a) When a linear programming model has an equality constraint, an artificial variable is introduced into this constraint in order to start the simplex method with an obvious initial basic solution that is feasible for the original model.
(b) When an artificial problem is created by introducing artificial variables and using the Big *M* method, if all artificial variables in an optimal solution for the artificial problem are equal to zero, then the real problem has no feasible solutions.
(c) The two-phase method is commonly used in practice because it usually requires fewer iterations to reach an optimal solution than the Big *M* method does.

4.6-12. Consider the following problem.

Maximize $Z = 3x_1 + 7x_2 + 5x_3$,

subject to

$$\begin{aligned} 3x_1 + x_2 + 2x_3 &\leq 9 \\ -2x_1 + x_2 + 3x_3 &\leq 12 \end{aligned}$$

and

$$x_2 \geq 0, \qquad x_3 \geq 0$$

(no nonnegativity constraint for x_1).
(a) Reformulate this problem so all variables have nonnegativity constraints.
D,I **(b)** Work through the simplex method step by step to solve the problem.
C **(c)** Use a software package based on the simplex method to solve the problem.

4.6-13.* Consider the following problem.

Maximize $Z = -x_1 + 4x_2$,

subject to

$$\begin{aligned} -3x_1 + x_2 &\leq 6 \\ x_1 + 2x_2 &\leq 4 \\ x_2 &\geq -3 \end{aligned}$$

(no lower bound constraint for x_1).
D,I **(a)** Solve this problem graphically.
(b) Reformulate this problem so that it has only two functional constraints and all variables have nonnegativity constraints.
D,I **(c)** Work through the simplex method step by step to solve the problem.

4.6-14. Consider the following problem.

Maximize $Z = -x_1 + 2x_2 + x_3$,

subject to

$$\begin{aligned} 3x_2 + x_3 &\leq 120 \\ x_1 - x_2 - 4x_3 &\leq 80 \\ -3x_1 + x_2 + 2x_3 &\leq 100 \end{aligned}$$

(no nonnegativity constraints).
(a) Reformulate this problem so that all variables have nonnegativity constraints.
D,I **(b)** Work through the simplex method step by step to solve the problem.
C **(c)** Use a computer package based on the simplex method to solve the problem.

4.6-15. This chapter has described the simplex method as applied to linear programming problems where the objective function is to be maximized. Section 4.6 then described how to convert a minimization problem to an equivalent maximization problem for applying the simplex method. Another option with minimization problems is to make a few modifications in the instructions for the simplex method given in the chapter in order to apply the algorithm directly.
(a) Describe what these modifications would need to be.
(b) Using the Big *M* method, apply the modified algorithm developed in part (*a*) to solve the following problem directly by hand. (Do not use your OR Courseware.)

Minimize $Z = 3x_1 + 8x_2 + 5x_3$,

subject to

$$\begin{aligned} 3x_2 + 4x_3 &\geq 70 \\ 3x_1 + 5x_2 + 2x_3 &\geq 70 \end{aligned}$$

and

$$x_1 \geq 0, \qquad x_2 \geq 0, \qquad x_3 \geq 0.$$

4.6-16. Consider the following problem.

Maximize $Z = -2x_1 + x_2 - 4x_3 + 3x_4$,

subject to

$$\begin{aligned} x_1 + x_2 + 3x_3 + 2x_4 &\leq 4 \\ x_1 - x_3 + x_4 &\geq -1 \\ 2x_1 + x_2 &\leq 2 \\ x_1 + 2x_2 + x_3 + 2x_4 &= 2 \end{aligned}$$

and

$$x_2 \geq 0, \qquad x_3 \geq 0, \qquad x_4 \geq 0$$

(no nonnegativity constraint for x_1).

(a) Reformulate this problem to fit our standard form for a linear programming model presented in Sec. 3.2.
(b) Using the Big M method, construct the complete first simplex tableau for the simplex method and identify the corresponding initial (artificial) BF solution. Also identify the initial entering basic variable and the leaving basic variable.
(c) Using the two-phase method, construct row 0 of the first simplex tableau for phase 1.
C **(d)** Use a computer package based on the simplex method to solve the problem.

I **4.6-17.** Consider the following problem.

$$\text{Maximize} \quad Z = 4x_1 + 5x_2 + 3x_3,$$

subject to

$$\begin{aligned} x_1 + x_2 + 2x_3 &\geq 20 \\ 15x_1 + 6x_2 - 5x_3 &\leq 50 \\ x_1 + 3x_2 + 5x_3 &\leq 30 \end{aligned}$$

and

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Work through the simplex method step by step to demonstrate that this problem does not possess any feasible solutions.

4.7-1. Refer to Fig. 4.10 and the resulting *allowable range* for the respective right-hand sides of the Wyndor Glass Co. problem given in Sec. 3.1. Use graphical analysis to demonstrate that each given allowable range is correct.

4.7-2. Reconsider the model in Prob. 4.1-5. Interpret the right-hand side of the respective functional constraints as the amount available of the respective resources.
I **(a)** Use graphical analysis as in Fig. 4.8 to determine the shadow prices for the respective resources.
I **(b)** Use graphical analysis to perform sensitivity analysis on this model. In particular, check each parameter of the model to determine whether it is a *sensitive* parameter (a parameter whose value cannot be changed without changing the optimal solution) by examining the graph that identifies the optimal solution.
I **(c)** Use graphical analysis as in Fig. 4.9 to determine the allowable range for each c_j value (coefficient of x_j in the objective function) over which the current optimal solution will remain optimal.
I **(d)** Changing just one b_i value (the right-hand side of functional constraint i) will shift the corresponding constraint boundary. If the current optimal CPF solution lies on this constraint boundary, this CPF solution also will shift. Use graphical analysis to determine the allowable range for each b_i value over which this CPF solution will remain feasible.
C **(e)** Verify your answers in parts (*a*), (*c*), and (*d*) by using a computer package based on the simplex method to solve the problem and then to generate sensitivity analysis information.

4.7-3. You are given the following linear programming problem.

$$\text{Maximize} \quad Z = 3x_1 + 2x_2,$$

subject to

$$\begin{aligned} 3x_1 \qquad &\leq 60 \qquad \text{(resource 1)} \\ 2x_1 + 3x_2 &\leq 75 \qquad \text{(resource 2)} \\ 2x_2 &\leq 40 \qquad \text{(resource 3)} \end{aligned}$$

and

$$x_1 \geq 0, \quad x_2 \geq 0.$$

D,I **(a)** Solve this problem graphically.
(b) Use graphical analysis to find the shadow prices for the resources.
(c) Determine how many additional units of resource 1 would be needed to increase the optimal value of Z by 15.

4.7-4. Consider the following problem.

$$\text{Maximize} \quad Z = x_1 - 7x_2 + 3x_3,$$

subject to

$$\begin{aligned} 2x_1 + x_2 - x_3 &\leq 4 \qquad \text{(resource 1)} \\ 4x_1 - 3x_2 \qquad &\leq 2 \qquad \text{(resource 2)} \\ -3x_1 + 2x_2 + x_3 &\leq 3 \qquad \text{(resource 3)} \end{aligned}$$

and

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

D,I **(a)** Work through the simplex method step by step to solve the problem.
(b) Identify the shadow prices for the three resources and describe their significance.
C **(c)** Use a software package based on the simplex method to solve the problem and then to generate sensitivity information. Use this information to identify the shadow price for each resource, the allowable range for each objective function coefficient, and the allowable range for each right-hand side.

4.7-5.* Consider the following problem.

$$\text{Maximize} \quad Z = 2x_1 - 2x_2 + 3x_3,$$

subject to

$$\begin{aligned} -x_1 + x_2 + x_3 &\leq 4 \qquad \text{(resource 1)} \\ 2x_1 - x_2 + x_3 &\leq 2 \qquad \text{(resource 2)} \\ x_1 + x_2 + 3x_3 &\leq 12 \qquad \text{(resource 3)} \end{aligned}$$

and

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

D,I **(a)** Work through the simplex method step by step to solve the problem.
(b) Identify the shadow prices for the three resources and describe their significance.
C **(c)** Use a software package based on the simplex method to solve the problem and then to generate sensitivity information. Use this information to identify the shadow price for each resource, the allowable range for each objective function coefficient and the allowable range for each right-hand side.

4.7-6. Consider the following problem.

$$\text{Maximize} \quad Z = 5x_1 + 4x_2 - x_3 + 3x_4,$$

subject to

$$3x_1 + 2x_2 - 3x_3 + x_4 \le 24 \quad \text{(resource 1)}$$
$$3x_1 + 3x_2 + x_3 + 3x_4 \le 36 \quad \text{(resource 2)}$$

and

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0, \quad x_4 \ge 0.$$

D,I **(a)** Work through the simplex method step by step to solve the problem.

(b) Identify the shadow prices for the two resources and describe their significance.

C **(c)** Use a software package based on the simplex method to solve the problem and then to generate sensitivity information. Use this information to identify the shadow price for each resource, the allowable range for each objective function coefficient, and the allowable range for each right-hand side.

4.9.1. Use the interior-point algorithm in your IOR Tutorial to solve the model in Prob. 4.1-4. Choose $\alpha = 0.5$ from the Option menu, use $(x_1, x_2) = (0.1, 0.4)$ as the initial trial solution, and run 15 iterations. Draw a graph of the feasible region, and then plot the trajectory of the trial solutions through this feasible region.

4.9-2. Repeat Prob. 4.9-1 for the model in Prob. 4.1-5.

CASES

CASE 4.1 Fabrics and Fall Fashions

From the tenth floor of her office building, Katherine Rally watches the swarms of New Yorkers fight their way through the streets infested with yellow cabs and the sidewalks littered with hot dog stands. On this sweltering July day, she pays particular attention to the fashions worn by the various women and wonders what they will choose to wear in the fall. Her thoughts are not simply random musings; they are critical to her work since she owns and manages TrendLines, an elite women's clothing company.

Today is an especially important day because she must meet with Ted Lawson, the production manager, to decide upon next month's production plan for the fall line. Specifically, she must determine the quantity of each clothing item she should produce given the plant's production capacity, limited resources, and demand forecasts. Accurate planning for next month's production is critical to fall sales since the items produced next month will appear in stores during September, and women generally buy the majority of the fall fashions when they first appear in September.

She turns back to her sprawling glass desk and looks at the numerous papers covering it. Her eyes roam across the clothing patterns designed almost six months ago, the lists of materials requirements for each pattern, and the lists of demand forecasts for each pattern determined by customer surveys at fashion shows. She remembers the hectic and sometimes nightmarish days of designing the fall line and presenting it at fashion shows in New York, Milan, and Paris. Ultimately, she paid her team of six designers a total of \$860,000 for their work on her fall line. With the cost of hiring runway models, hair stylists, and makeup artists, sewing and fitting clothes, building the set, choreographing and rehearsing the show, and renting the conference hall, each of the three fashion shows cost her an additional \$2,700,000.

She studies the clothing patterns and material requirements. Her fall line consists of both professional and casual fashions. She determined the prices for each clothing item by taking into account the quality and cost of material, the cost of labor and machining, the demand for the item, and the prestige of the TrendLines brand name.

The fall professional fashions include:

Clothing Item	Materials Requirements	Price	Labor and Machine Cost
Tailored wool slacks	3 yards of wool 2 yards of acetate for lining	\$300	\$160
Cashmere sweater	1.5 yards of cashmere	\$450	\$150
Silk blouse	1.5 yards of silk	\$180	\$100
Silk camisole	0.5 yard of silk	\$120	\$ 60
Tailored skirt	2 yards of rayon 1.5 yards of acetate for lining	\$270	\$120
Wool blazer	2.5 yards of wool 1.5 yards of acetate for lining	\$320	\$140

The fall casual fashions include:

Clothing Item	Materials Requirements	Price	Labor and Machine Cost
Velvet pants	3 yards of velvet 2 yards of acetate for lining	$350	$175
Cotton sweater	1.5 yards of cotton	$130	$ 60
Cotton miniskirt	0.5 yard of cotton	$ 75	$ 40
Velvet shirt	1.5 yards of velvet	$200	$160
Button-down blouse	1.5 yards of rayon	$120	$ 90

She knows that for the next month, she has ordered 45,000 yards of wool, 28,000 yards of acetate, 9,000 yards of cashmere, 18,000 yards of silk, 30,000 yards of rayon, 20,000 yards of velvet, and 30,000 yards of cotton for production. The prices of the materials are as follows:

Material	Price per yard
Wool	$ 9.00
Acetate	$ 1.50
Cashmere	$60.00
Silk	$13.00
Rayon	$ 2.25
Velvet	$12.00
Cotton	$ 2.50

Any material that is not used in production can be sent back to the textile wholesaler for a full refund, although scrap material cannot be sent back to the wholesaler.

She knows that the production of both the silk blouse and cotton sweater leaves leftover scraps of material. Specifically, for the production of one silk blouse or one cotton sweater, 2 yards of silk and cotton, respectively, are needed. From these 2 yards, 1.5 yards are used for the silk blouse or the cotton sweater and 0.5 yard is left as scrap material. She does not want to waste the material, so she plans to use the rectangular scrap of silk or cotton to produce a silk camisole or cotton miniskirt, respectively. Therefore, whenever a silk blouse is produced, a silk camisole is also produced. Likewise, whenever a cotton sweater is produced, a cotton miniskirt is also produced. Note that it is possible to produce a silk camisole without producing a silk blouse and a cotton miniskirt without producing a cotton sweater.

The demand forecasts indicate that some items have limited demand. Specifically, because the velvet pants and velvet shirts are fashion fads, TrendLines has forecasted that it can sell only 5,500 pairs of velvet pants and 6,000 velvet shirts. TrendLines does not want to produce more than the forecasted demand because once the pants and shirts go out of style, the company cannot sell them. TrendLines can produce less than the forecasted demand, however, since the company is not required to meet the demand. The cashmere sweater also has limited demand because it is quite expensive, and TrendLines knows it can sell at most 4,000 cashmere sweaters. The silk blouses and camisoles have limited demand because many women think silk is too hard to care for, and TrendLines projects that it can sell at most 12,000 silk blouses and 15,000 silk camisoles.

The demand forecasts also indicate that the wool slacks, tailored skirts, and wool blazers have a great demand because they are basic items needed in every professional wardrobe. Specifically, the demand for wool slacks is 7,000 pairs of slacks, and the demand for wool blazers is 5,000 blazers. Katherine wants to meet at least 60 percent of the demand for these two items in order to maintain her loyal customer base and not lose business in the future. Although the demand for tailored skirts could not be estimated, Katherine feels she should make at least 2,800 of them.

(a) Ted is trying to convince Katherine not to produce any velvet shirts since the demand for this fashion fad is quite low. He argues that this fashion fad alone accounts for $500,000 of the fixed design and other costs. The net contribution (price of clothing item − materials cost − labor cost) from selling the fashion fad should cover these fixed costs. Each velvet shirt generates a net contribution of $22. He argues that given the net contribution, even satisfying the maximum demand will not yield a profit. What do you think of Ted's argument?

(b) Formulate and solve a linear programming problem to maximize profit given the production, resource, and demand constraints.

Before she makes her final decision, Katherine plans to explore the following questions independently except where otherwise indicated.

(c) The textile wholesaler informs Katherine that the velvet cannot be sent back because the demand forecasts show that the demand for velvet will decrease in the future. Katherine can therefore get no refund for the velvet. How does this fact change the production plan?

(d) What is an intuitive economic explanation for the difference between the solutions found in parts (*b*) and (*c*)?

(e) The sewing staff encounters difficulties sewing the arms and lining into the wool blazers since the blazer pattern has an awkward shape and the heavy wool material is difficult to cut and sew. The increased labor time to sew a wool blazer increases the labor and machine cost for each blazer by \$80. Given this new cost, how many of each clothing item should TrendLines produce to maximize profit?

(f) The textile wholesaler informs Katherine that since another textile customer canceled his order, she can obtain an extra 10,000 yards of acetate. How many of each clothing item should TrendLines now produce to maximize profit?

(g) TrendLines assumes that it can sell every item that was not sold during September and October in a big sale in November at 60 percent of the original price. Therefore, it can sell all items in unlimited quantity during the November sale. (The previously mentioned upper limits on demand concern only the sales during September and October.) What should the new production plan be to maximize profit?

PREVIEWS OF ADDED CASES ON OUR WEBSITE (www.mhhe.com/hillier)

CASE 4.2 New Frontiers

AmeriBank will soon begin offering Web banking to its customers. To guide its planning for the services to provide over the Internet, a survey will be conducted with four different age groups in three types of communities. AmeriBank is imposing a number of constraints on how extensively each age group and each community should be surveyed. Linear programming is needed to develop a plan for the survey that will minimize its total cost while meeting all the survey constraints under several different scenarios.

CASE 4.3 Assigning Students to Schools

After deciding to close one of its middle schools, the Springfield school board needs to reassign all of next year's middle school students to the three remaining middle schools. Many of the students will be bussed, so minimizing the total bussing cost is one objective. Another is to minimize the inconvenience and safety concerns for the students who will walk or bicycle to school. Given the capacities of the three schools, as well as the need to roughly balance the number of students in the three grades at each school, how can linear programming be used to determine how many students from each of the city's six residential areas should be assigned to each school? What would happen if each entire residential area must be assigned to the same school? (This case will be continued in Cases 6.3 and 11.4.)