

Optimization Modeling with Spreadsheets

Second Edition

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Network: Production Planning Problem																
2																	
3	Decisions	Crude	Dist	Low Blend	Feed	Cat	High	BR	BP	CR	CP	Reg	Prem				
4		Kbbl	49.3	29.6	19.7	9.6	20.0	12.8	8.0	8.00	1.60	8.00	4.80	16.0	6.4		
5	Objective																
6	price				36			44						50	55		
7	cost		5			6											
8	cost		28														
9	net		-33	0	36	0	-6	0	44	0	0	0	0	50	55	466.400	
10	Constraints															LHS	RHS
11																	
12	Tower	-0.60	1	0	0	0	0	0	0	0	0	0	0	0	0	0.0	= 0
13	Tower capacity	-0.40	0	1	0	0	0	0	0	0	0	0	0	0	0	0.0	= 0
14	Distillate split	0	-1	0	1	1	0	0	0	0	0	0	0	0	0	0.0	= 0
15	Cracker	0	0	0	0	-0.64	1	0	0	0	0	0	0	0	0	0.0	= 0
16	Cracker	0	0	0	0	-0.40	0	1	0	0	0	0	0	0	0	0.0	= 0
17	Blend split	0	0	0	-1	0	0	0	1	1	0	0	0	0	0	0.0	= 0
18	Catalytic split	0	0	0	0	0	-1	0	0	0	1	1	0	0	0	0.0	= 0
19	Reg composition	0	0	0	0	0	0	0	-1	0	-1	0	1	0	0	0.0	= 0
20	Prem composition	0	0	0	0	0	0	0	0	-1	0	1	0	1	0	0.0	= 0
21	Reg quality	0	0	0	0	0	0	0	0.50	0	0.50	0	0	0	0	0.0	= 0
22	Prem quality	0	0	0	0	0	0	0	0	-0.75	0	0.25	0	0	0	0.0	= 0
23	Tower capacity	1	0	0	0	0	0	0	0	0	0	0	0	0	49.3	<=	50
24	Cracker capacity	0	0	0	0	1	0	0	0	0	0	0	0	0	20.0	<=	20
25	Reg sales	0	0	0	0	0	0	0	0	0	0	0	1	0	16.0	<=	16
26	Prem sales	0	0	0	0	0	0	0	0	0	0	0	0	1	6.4	<=	16
27																	

Kenneth R. Baker

Copyright © 2011 by John Wiley & Sons, Inc. All rights reserved

Published by John Wiley & Sons, Inc., Hoboken, New Jersey
Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4470, or on the web at www.copyright.com. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permission>.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services or for technical support, please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in variety of electronic formats. Some content that appears in print may not be available in electronic formats. For more information about Wiley products, visit our web site at www.wiley.com.

Library of Congress Cataloging-in-Publication Data:

Baker, Kenneth R., 1943-

Optimization modeling with spreadsheets/Kenneth R. Baker. — 2nd ed.
p. cm.

Includes bibliographical references and index.

ISBN 978-0-470-92863-9 (cloth)

1. Mathematical optimization.
 2. Managerial economics—Mathematical models.
 3. Electronic spreadsheets.
 4. Programming (Mathematics). I. Title.
- HB143.7.B35 2011
005.54—dc22

2010036840

Printed in Singapore

10 9 8 7 6 5 4 3 2 1

Preface

This is an introductory textbook on optimization—that is, on mathematical programming—intended for undergraduates and graduate students in management or engineering. The principal coverage includes linear programming, nonlinear programming, integer programming, and heuristic programming; and the emphasis is on model building using Excel and Solver.

The emphasis on model building (rather than algorithms) is one of the features that makes this book distinctive. Most textbooks devote more space to algorithmic details than to formulation principles. These days, however, it is not necessary to know a great deal about algorithms in order to apply optimization tools, especially when relying on the spreadsheet as a solution platform.

The emphasis on spreadsheets is another feature that makes this book distinctive. Few textbooks devoted to optimization pay much attention to spreadsheet implementation of optimization principles, and most books that emphasize model building ignore spreadsheets entirely. Thus, someone looking for a spreadsheet-based treatment would otherwise have to use a textbook that was designed for some other purpose, like a survey of management science topics, rather than one devoted to optimization.

WHY MODEL BUILDING?

The model building emphasis is an attempt to be realistic about what business and engineering students need most when learning about optimization. At an introductory level, the most practical and motivating theme is the wide applicability of optimization tools. To apply optimization effectively, the student needs more than a brief exposure to a series of numerical examples, which is the way that most mathematical programming books treat applications. With a systematic modeling emphasis, the student can begin to see the basic structures that appear in optimization models and as a result, develop an appreciation for potential applications well beyond the examples presented in the text.

Formulating optimization models is both an art and a science, and this book pays attention to both. The art can be refined with practice, especially supervised practice, just the way a student would learn sculpture or painting. The science is reflected in the structure that organizes the topics in this book. For example, there are several distinct problem types that lend themselves to linear programming formulations, and it makes sense to study these types systematically. In that spirit, the book builds a library of

templates against which new problems can be compared. Analogous structures are developed for the presentation of other topics as well.

WHY SPREADSHEETS?

Now that optimization tools have been made available with spreadsheets (i.e., with Excel), every spreadsheet user is potentially a practitioner of optimization techniques. No longer do practitioners of optimization constitute an elite, highly trained group of quantitative specialists who are well versed in computer software, or their former professor's own code. Now, anyone who builds a spreadsheet model can call on optimization techniques, and can do so without the need to learn about specialized software. The basic optimization tool, in the form of Excel's Standard Solver, is now as readily available as the spellchecker. So why not raise modeling ability up to the level of software access? Let's not pretend that most users of optimization tools will be inclined to shop around for matrix generators and industrial-strength "solvers" if they want to produce numbers. More likely, they will be drawn to Excel.

Students using this book can take advantage of an even more powerful software package called Risk Solver Platform (RSP) that was developed by the creators of Excel's built-in Standard Solver. The educational version of RSP is available at no cost (see Appendix 1), and it introduces students to the capabilities of a sophisticated optimization package. Although this book is not organized as a user's manual, it nevertheless provides most of the information the student needs to become a sophisticated user of RSP.

WHAT'S SPECIAL?

Mathematical programming techniques have been invented and applied for more than half a century, so by now they represent a relatively mature area of applied mathematics. There is not much new that can be said in an introductory textbook regarding the underlying concepts. The innovations in this book can be found instead in the delivery and elaboration of certain topics, making them accessible and understandable to the novice. The most distinctive of these features are as follows.

- The major topics are not illustrated merely with a series of numerical examples. Instead, the chapters introduce a classification for the problem types. An early example is the organization of basic linear programming models in Chapter 2 along the lines of allocation, covering, and blending models. This classification strategy, which extends throughout the book, helps the student to see beyond the particular examples to the breadth of possible applications.
- Network models are a special case of linear programming models. If they are singled out for special treatment at all in optimization books, they are defined by a strict requirement for mass balance. Here, in Chapter 3, network models are presented in a broader framework, which allows for a more general form

of mass balance, thereby extending the reader's capability for recognizing and analyzing network problems.

- Interest has been growing in Data Envelopment Analysis (DEA), a special kind of linear programming application. Although some books illustrate DEA with a single example, this book provides a systematic introduction to the topic by providing a patient, comprehensive treatment in Chapter 5.
- Analysis of an optimization problem does not end when the computer displays the numbers in an optimal solution. Finding a solution must be followed with a meaningful interpretation of the results, especially if the optimization model was built to serve a client. An important framework for interpreting linear programming solutions is the identification of patterns, which is discussed in detail in Chapter 4.
- The topic of heuristic programming has evolved somewhat outside the field of optimization. Although a variety of specialized heuristic approaches have been developed, generic software has seldom been available. Now, however, the advent of the evolutionary solver in Solver and RSP brings heuristic programming alongside linear and nonlinear programming as a generic software tool for pursuing optimal decisions. The evolutionary solver is covered in Chapter 9.
- The topic of stochastic programming has been of interest to researchers for quite a while but promises to become a factor in applications now that the latest versions of RSP contain such capabilities as the solution of stochastic programs with recourse. Appendix 4 provides an introduction to this topic area and a glimpse of how to use RSP to solve problems of this type.

Beyond these specific innovations, as this book goes to print, there is no optimization textbook exclusively devoted to model building rather than algorithms that relies on the spreadsheet platform. The reliance on spreadsheets and on a model-building emphasis is the most effective way to bring optimization capability to the many users of Excel.

THE AUDIENCE

This book is aimed at management students and secondarily engineering students. In business curricula, a course focused on optimization is viable in two situations. If there is no required introduction to Management Science at all, then the treatment of Management Science at the elective level is probably best done with specialized courses on deterministic and probabilistic models. This book is an ideal text for a first course dedicated to deterministic models. If instead there is a required introduction to Management Science, chances are that the coverage of optimization glides by so quickly that even the motivated student is left wanting more detail, more concepts and more practice. This book is also well suited to a second-level course that delves specifically into mathematical programming applications.

In engineering curricula, it is still typical to find a full course on optimization, usually as the first course on (deterministic) modeling. Even in this setting, though,

traditional textbooks tend to leave it to the student to seek out spreadsheet approaches to the topic, while covering the theory and perhaps encouraging students to write code for algorithms. This book will capture the energies of students by covering what they would be spending most of their time doing in the real world—building and solving optimization problems on spreadsheets.

This book has been developed around the syllabi of two courses at Dartmouth College that have been delivered for several years. One course is a second-year elective for MBA students who have had a brief, previous exposure to optimization during a required core course that surveyed other analytic topics. A second course is a required course for Engineering Management students in a graduate program at the interface between business and engineering. These students have had no formal exposure to spreadsheet modeling, although some may previously have taken a survey course in Operations Research. Thus, the book is appropriate for students who are about to study optimization with only a brief, or even nonexistent exposure to the subject.

ACKNOWLEDGEMENTS

I can trace the roots of this book to my collaboration with Steve Powell in teaching and writing. Using spreadsheets to teach optimization is part of a broader activity in which Steve has been an active and inspiring leader. Were it not for his professional companionship, this book would not have been undertaken. Other authors have been successful in creating spreadsheet-based approaches to familiar topics in Management Science, yet there was much for me to learn about writing effective text material. For their suggestions and guidance, I particularly thank my reviewers on the first edition, Jiu Ding, A. Thomas Mason, James G. Morris, James Pratt, Paul Savory, and Wilbert E. Wilhelm. In addition, Rich Metters, Clay Whybark, Alan Neebe, and Kusum Ailawadi graciously supported my adaptations of their original case material and extended their support to the second edition. Two department chairs, Ron Askin and Asoo Vakharia, generously hosted my short stays in their departments during the development of the first edition. Scott Webster class-tested the first versions of the book and provided a careful and patient accuracy check while suggesting several improvements. Curt Hinrichs, my original editor, orchestrated the initial cycles of review, refinement, and improvement that led to the publication of the first edition. Another very helpful influence has been the feedback from my students at the Tuck School and the Thayer School. Rather than list their names here, I've acknowledged them indirectly throughout the book.

The main technical change in the second edition is the use of Risk Solver Platform, and I am indebted to Dan Fylstra and Edwin Straver at Frontline Systems for being responsive to my queries as they focused on the introduction of an excellent new product. As the second edition comes to fruition, I wish to thank my current editor, Susanne Steitz-Filler, for providing the opportunity to refine and update my work and ultimately to reach out to a new and broader audience.

Table of Contents

Preface xi

1. Introduction to Spreadsheet Models for Optimization **1**

1.1	Elements of a Model	2
1.2	Spreadsheet Models	4
1.3	A Hierarchy for Analysis	7
1.4	Optimization Software	8
1.5	Using Solver	10
	Summary	17
	Exercises	18
	References	20

2. Linear Programming: Allocation, Covering, and Blending Models **21**

2.1	Linear Models	22
2.1.1	Linear Constraints	24
2.1.2	Formulation	25
2.1.3	Layout	26
2.1.4	Results	28
2.2	Allocation Models	29
2.2.1	The Product Mix Problem	35
2.3	Covering Models	39
2.3.1	The Staff-Scheduling Problem	43
2.4	Blending Models	47
2.5	Modeling Errors in Linear Programming	53
2.5.1	Exceptions	53
2.5.2	Debugging	55
2.5.3	Logic	56
	Summary	57
	Exercises	57
	Case: JetGreen	68

3. Linear Programming: Network Models 71

3.1	The Transportation Model	72
3.2	The Assignment Model	77
3.3	The Transshipment Model	81
3.4	Features of Special Network Models	85
3.5	Building Network Models with Balance Equations	86
3.6	General Network Models with Yields	91
3.6.1	Models with Yield Losses	91
3.6.2	Models with Yield Gains	93
3.7	General Network Models with Transformed Flows	98
	Summary	103
	Exercises	103
	Case: Casey's Famous Roast Beef	113
	Case: Hollingsworth Paper Company	114
	Production and Distribution Facilities	115
	Patterns of Distribution	115
	Expansion Proposals	116

4. Sensitivity Analysis in Linear Programs 119

4.1	Parameter Analysis in the Transportation Example	120
4.2	Parameter Analysis in the Allocation Example	127
4.3	The Sensitivity Report and the Transportation Example	135
4.4	The Sensitivity Report and the Allocation Example	138
4.5	Degeneracy and Alternative Optima	140
4.6	Patterns in Linear Programming Solutions	144
4.6.1	The Transportation Model	145
4.6.2	The Product Portfolio Model	149
4.6.3	The Investment Model	152
4.6.4	The Allocation Model	154
4.6.5	The Refinery Model	155
	Summary	159
	Exercises	160
	Case: Cox Cable and Wire Company	171
	Background	171
	The Contract	172
	The Analysis	173

5. Linear Programming: Data Envelopment Analysis 175

5.1	A Graphical Perspective on DEA	177
5.2	An Algebraic Perspective on DEA	181
5.3	A Spreadsheet Model for DEA	183
5.4	Indexing	188

5.5	Finding Reference Sets and HCUs	190
5.6	Assumptions and Limitations of DEA	193
	Summary	196
	Exercises	196
	Case: Branch Performance at Nashville National Bank	205
	Branch Growth at Nashville National Bank	205
	Assessing Branch Productivity	206
	Branch Managers Revolt	206
	Measuring Branches: Available Techniques	207
	The DEA Study	207

6. Integer Programming: Binary Choice Models **211**

6.1	Using Solver with Integer Requirements	213
6.2	The Capital Budgeting Problem	217
6.3	Set Covering	221
6.4	Set Packing	224
6.5	Set Partitioning	227
6.6	Playoff Scheduling	229
6.7	Solving a Large-Scale Set Partitioning Problem	234
6.8	The Algorithm for Solving Integer Programs	237
	Summary	243
	Exercises	243
	Case: Motel Location for Nature's Inn	249

7. Integer Programming: Logical Constraints **251**

7.1	Simple Logical Constraints: Exclusivity and Contingency	253
7.2	Linking Constraints: The Fixed Cost Problem	255
7.3	Linking Constraints: The Threshold Level Problem	260
7.4	Linking Constraints: The Facility Location Model	261
	7.4.1 Capacitated Version	263
	7.4.2 Uncapacitated Version	267
7.5	Disjunctive Constraints: The Machine Sequencing Problem	270
7.6	Tour and Subset Constraints: The Traveling Salesperson Problem	274
	Summary	282
	Exercises	283
	Case: Hornby Products Company	291
	History	291
	Alternatives	293

8. Nonlinear Programming **297**

8.1	One-variable Models	298
	8.1.1 An Inventory Example	300
	8.1.2 A Quantity Discount Example	302

8.2	Local Optima and the Search for an Optimum	304
8.3	Two-Variable Models	307
8.3.1	Curve Fitting	307
8.3.2	Two-dimensional Location	310
8.4	Nonlinear Models with Constraints	312
8.4.1	A Pricing Example	313
8.4.2	Sensitivity Analysis for Nonlinear Programs	315
8.4.3	The Portfolio Optimization Model	316
8.5	Linearizations	320
8.5.1	Linearizing the Maximum	320
8.5.2	Linearizing the Absolute Value	324
	Summary	327
	Exercises	328
	Case: Delhi Foods	335

9. Heuristic Solutions with the Evolutionary Solver 337

9.1	Features of the Evolutionary Solver	338
9.2	An Illustrative Example: Nonlinear Regression	339
9.3	The Machine-Sequencing Problem Revisited	346
9.4	The Traveling Salesperson Problem Revisited	349
9.5	Two-dimensional Location	352
9.6	Line Balancing	354
9.7	Group Assignment	358
	Summary	362
	Exercises	362
	Case: Colgate Wave (Abridged)	370
	Introduction	370
	The Study	370
	Case Appendix: Market Share Simulation Model (Colgate.xls)	373
	Data	373
	Calculations	373
	Simulation	374

Appendices

1. Optimization Software and Supplemental Files 375

A1.1	Risk Solver Platform	375
A1.2	Supplemental Excel Files	376

2. Graphical Methods in Linear Programming 377

A2.1	An Example	377
A2.2	Generalities	382

3. The Simplex Method	385
A3.1 An Example	385
A3.2 Variations of the Algorithm	390
References	393
4. Stochastic Programming	395
A4.1 One-Stage Decisions with Uncertainty	395
A4.2 Two-Stage Decisions with Uncertainty	399
A4.3 Using Solver	402
Index	407

Chapter 1

Introduction to Spreadsheet Models for Optimization

This is a book about optimization with an emphasis on building models and using spreadsheets. Each facet of this theme—models, spreadsheets, and optimization—has a role in defining the emphasis of our coverage.

A *model* is a simplified representation of a situation or problem. Models attempt to capture the essential features of a complicated situation so that it can be studied and understood more completely. In the worlds of business, engineering, and science, models aim to improve our understanding of practical situations. Models can be built with tangible materials, or words, or mathematical symbols and expressions. A *mathematical model* is a model that is constructed—and also analyzed—using mathematics. In this book, we focus on mathematical models. Moreover, we work with *decision models*, or models that contain representations of decisions. The term also refers to models that support decision-making activities.

A *spreadsheet* is a row-and-column layout of text, numerical data, and logical information. The spreadsheet version of a model contains the model's elements, linked together by specific logical information. Electronic spreadsheets, like those built using Microsoft Excel®, have become familiar tools in the business, engineering, and scientific worlds. Spreadsheets are relatively easy to understand, and people often rely on spreadsheets to communicate their analyses. In this book, we focus on the use of spreadsheets to represent and analyze mathematical models.

This text is written for an audience that already has some familiarity with Excel. Our coverage assumes a level of facility with Excel comparable to a beginner's level. Someone who has used other people's spreadsheets and built simple spreadsheets for some purpose—either personal or organizational—has probably developed this skill level. Box 1.1 describes the Excel skill level assumed. Readers without this level of background are encouraged to first work through some introductory materials, such as the books by McFedries (1) and Reding and Wermers (2).

Optimization is the process of finding the best values of the variables for a particular criterion or, in our context, the best decisions for a particular measure of performance. The elements of an optimization problem are a set of decisions, a criterion, and

BOX 1.1 *Excel Skills Assumed as Background for this Book*

Navigating in workbooks, worksheets, and windows.
 Using the cursor to select cells, rows, columns, and noncontiguous cell ranges.
 Entering text and data; copying and pasting; filling down or across.
 Formatting cells (number display, alignment, font, border, and protection).
 Editing cells (using the formula bar and cell-edit capability [F2]).
 Entering formulas and using the function wizard.
 Using relative and absolute addresses.
 Using range names.
 Creating charts and graphs.

perhaps a set of required conditions, or *constraints*, that the decisions must satisfy. These elements lend themselves to description in a mathematical model. The term optimization sometimes refers specifically to a procedure that is implemented by software. However, in this book, we expand that perspective to include the model-building process as well as the process of finding the best decisions.

Not all mathematical models are optimization models. Some models merely describe the logical relationship between inputs and outputs. Optimization models are a special kind of model in which the purpose is to find the best value of a particular output measure and the choices that produce it. Optimization problems abound in the real world, and if we're at all ambitious or curious, we often find ourselves seeking solutions to those problems. Business firms are very interested in optimization because making good decisions helps a firm run efficiently, perform profitably, and compete effectively. In this book, we focus on optimization problems expressed in the form of spreadsheet models and solved using a spreadsheet-based approach.

1.1. ELEMENTS OF A MODEL

To restate our premise, we are interested in mathematical models. Specifically, we are interested in two forms—algebraic and spreadsheet models. In the former, we use algebraic notation to represent elements and relationships, and in the latter, we use spreadsheet entries and structure. For example, in an algebraic statement, we might use the variable x to represent a quantitative decision, and we might use some function $f(x)$ to represent the measure of performance that results from choosing decision x . Then we might adopt the letter z to represent a criterion for decision making and construct the equation $z = f(x)$ to guide the choice of a decision. Algebra is the basic language of analysis largely because it is precise and compact.

As an introductory modeling example, let's consider the price decision in the scenario of Example 1.1.

EXAMPLE 1.1 *Price, Demand, and Profit*

Our firm's production department has carried out a cost accounting study and found that the unit cost for one of its main products is \$40. Meanwhile, the marketing department has estimated the relationship between price and sales volume (the so called *demand curve* for the product) as follows:

$$y = 800 - 5x \quad (1.1)$$

where y represents quarterly demand and x represents the selling price per unit. We wish to determine a selling price for this product, given the information available. ■

In Example 1.1, the decision is the unit price, and the consequence of that decision is the level of demand. The demand curve in Equation 1.1 expresses the relationship of demand and price in algebraic terms. Another equation expresses the calculation of profit contribution, by multiplying the demand y by the unit profit contribution ($x - 40$) on each item

$$z = (x - 40)y \quad (1.2)$$

where z represents our product's quarterly profit contribution.

We can substitute Equation 1.1 into 1.2 if we want to write z algebraically as a function of x alone. As a result, we can express the profit contribution as

$$z = 1000x - 5x^2 - 32,000 \quad (1.3)$$

This step embodies the algebraic principle that simplification is always desirable. Here, simplification reduces the number of variables in the expression for profit contribution. Simplification, however, is not necessarily a virtue when we use a spreadsheet model.

Example 1.1, simple as it is, has some important features. First, our model contains three numerical inputs: 40 (the unit cost), -5 (the marginal effect of price on demand) and 800 (the maximum demand). Numerical inputs such as these are called *parameters*. In some models, parameters correspond to raw data, but in many cases, parameters are summaries drawn from a more primitive data set. They may also be estimates made by a knowledgeable party, forecasts derived from statistical analyses, or predictions chosen to reflect a future scenario.

Our model also contains a decision—an unknown quantity yet to be determined. In traditional algebraic formulations, unknowns are represented as variables. Quantitative representations of decisions are therefore called *decision variables*. The decision variable in our model is the unit price x .

Our model contains the equation that relates demand to price. We can think of this relationship as part of the model's logic. In that role, the demand curve prescribes a relationship between two variables—price and demand—that must always hold. Thus, in our model, the only admissible values of x and y are those that satisfy Equation 1.1.

Finally, our model contains a calculation of quarterly profit contribution, which is the performance measure of interest and a quantity that we wish to maximize. This output variable measures the consequence of selecting any particular price decision

in the model. In optimization models, we are concerned with maximizing or minimizing some measure of performance, expressed as a mathematical function, and we refer to it as the *objective function*, or simply the *objective*.

1.2. SPREADSHEET MODELS

Algebra is an established language that works well for describing problems, but not always for obtaining solutions. Algebraic solutions tend to occur in formulas, not numbers, but numbers most often represent decisions in the practical world. By contrast, spreadsheets represent a practical language—one that works very effectively with numbers. Like algebraic models, spreadsheets can be precise and compact, but there are also complications that are unique to spreadsheets. For example, there is a difference between form and content in a spreadsheet. Two spreadsheets may look the same in terms of layout and the numbers displayed on a computer screen, but the underlying formulas in corresponding cells could differ. Because the information behind the display can be different even when two spreadsheets have the same on-screen appearance, we can't always tell the logical content from the form of the display. Another complication is the lack of a single, well accepted way to build a spreadsheet representation of a given model. In an optimization model, we want to represent decision variables, an objective function, and constraints. However, that still leaves a lot of flexibility in choosing how the logic of a particular model is incorporated into a spreadsheet. Such flexibility would ordinarily be advantageous if the only use of a spreadsheet were to help individuals solve problems. However, spreadsheets are perhaps even more important as vehicles for communication. When we use spreadsheets in this role, flexibility can sometimes lead to confusion and disrupt the intended communication.

We will try to mitigate these complications with some design guidelines. For example, it is helpful to create separate modules in the spreadsheet for decision variables, objective function, and constraints. To the extent that we follow such guidelines, we may lose some flexibility in building a spreadsheet model. Moving the design process toward standardization will, however, make the content of a spreadsheet more understandable from its form, so differences between form and content become less problematic.

With optimization, a spreadsheet model contains the analysis that ultimately provides decision support. For this reason, the spreadsheet model should be intelligible to its users, not just to its developer. On some occasions, a spreadsheet might come into routine use in an organization, even when the developer moves on. New analysts may inherit the responsibilities associated with the model, so it is vital that they, too, understand how the spreadsheet works. For that matter, the decision maker may also move on. For the organization to retain the learning that has taken place, successive decision makers must also understand the spreadsheet. In yet another scenario, the analyst develops a model for one-time use but then discovers a need to reuse it several months later in a different context. In such a situation, it's important that the analyst understands the original model, lest the passage of time obscure its purpose and

logic. In all of these cases, the spreadsheet model fills a significant communications need. Thus, it is important to keep the role of communication in mind while developing a spreadsheet.

A spreadsheet version of our pricing model might look like the one in Figure 1.1. This spreadsheet contains a cell (C9) that holds the unit price, a cell (C12) that holds the level of demand, and a cell (C15) that holds the total profit contribution. Actually, cell C12 holds Equation 1.1 in the form of the Excel formula $=C4+C5*C9$. Similarly, cell C15 holds Equation 1.2 with the formula $=(C9-C6)*C12$. In cell C9, the unit price is initially set to \$80. For this choice, demand is 400. The quarterly profit contribution is \$16,000.

In a spreadsheet model, there is usually no premium on being concise, as there is when we use algebra. In fact, when conciseness begins to interfere with a model's transparency, it becomes undesirable. Thus, in Figure 1.1, the model retains the demand equation and displays the demand quantity explicitly; we have not tried to incorporate Equation 1.3. This form allows a user to see how price influences profit contribution through demand because all of these quantities are explicit. Furthermore, it is straightforward to trace the connection between the three input parameters and the calculation of profit contribution.

To summarize, our model consists of three parameters and a decision variable, together with some intermediate calculations, all leading to an objective function that we want to maximize. In algebraic terms, the model consists of Equations 1.1 and 1.2, with the prescription that we want to maximize Equation 1.2. In spreadsheet terms, the model consists of the spreadsheet in Figure 1.1, with the prescription that we want to maximize the value in cell C15.

	A	B	C
1	Price, Demand, and Profit		
2			
3	Inputs		
4		Max. demand	800
5		Slope	-5
6		Unit cost	40
7			
8	Decision		
9		Price	\$ 80
10			
11	Calculation		
12		Demand	400
13			
14	Outcome		
15		Profit	\$ 16,000
16			

Figure 1.1. Spreadsheet model for determining price.

The spreadsheet is organized into four modules: Inputs, Decision, Calculation, and Outcome, separating different kinds of information. In spreadsheet models, it is a good idea to separate input data from decisions and decisions from outcome measures. Intermediate calculations that do not lead directly to the outcome measure should also be kept separate.

In the spreadsheet model, cell borders and shading draw attention to the decision (cell C9) and the objective (cell C15) as the two most important elements of the optimization model. No matter how complicated a spreadsheet model may become, we want the decisions and the objective to be located easily by someone who looks at the display.

In the spreadsheet of Figure 1.1, the input parameters appear explicitly. It would not be difficult to skip the Inputs section entirely and express the demand function in cell C12 with the formula $=800 - 5 * C9$, or to express the profit contribution in cell C15 with the formula $=(C9 - 40) * C12$. This approach, however, places the numerical parameters in formulas, so a user would not see them at all when looking at the spreadsheet. Good practice calls for displaying parameters explicitly in the spreadsheet, as we have done in Figure 1.1, rather than burying them in formulas.

The basic version of our model, shown in Figure 1.1, is ready for optimization. But let's look at an alternative, shown in Figure 1.2. This version contains the four modules, and the numerical inputs are explicit but placed differently than in Figure 1.1. The main difference is that demand is treated as a decision variable, and the demand curve is expressed as an explicit constraint. Specifically, this form of the model treats both price and demand as variables in cells C9:C10, as if the two choices could be made arbitrarily. However, the Constraints module describes a relationship between the two variables in the form of Equation 1.1, which can

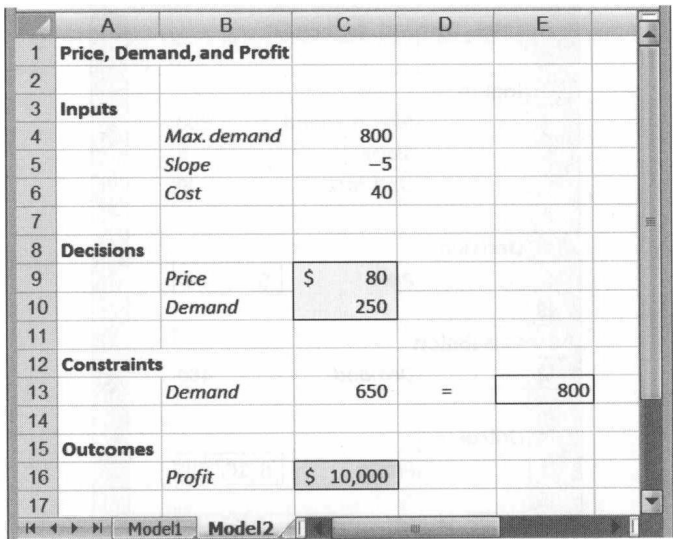


Figure 1.2. Alternative spreadsheet model for determining price.