APPENDIX 1

OR TUTOR MPL

DOCUMENTATION FOR THE OR COURSEWARE

You will find a wealth of software resources on the CD-ROM packaged in the back of the book. The entire software package is called *OR Courseware.*

The installation instructions and system requirements are specified on the front of the CD-ROM. Although the CD-ROM is designed for use on a Windows-based IBM-compatible PC, much of the software also can be run on a Macintosh (as specified later for the individual cases).

To get started, and to see an overview of the available software resources, refer to the introductory screens on the CD-ROM. The individual software packages also are discussed briefly below.

OR TUTOR

OR Tutor is a Web document consisting of a set of HTML pages that often contain JavaScript. Any browser that supports JavaScript can be used, including Netscape Navigator 4.0 (or higher) or Internet Explorer 4.5 (or higher). It can be viewed with either an IBM-compatible PC or a Macintosh.

This resource has been designed to be your personal tutor by illustrating and illuminating key concepts in an interactive manner. It contains 16 *demonstration examples* that supplement the examples in the book in ways that cannot be duplicated on the printed page. Each one vividly demonstrates one of the algorithms or concepts of OR in action. Most combine an *algebraic description* of each step with a *geometric display* of what is happening. Some of these geometric displays become quite dynamic, with moving points or moving lines, to demonstrate the evolution of the algorithm. The demonstration examples also are integrated with the book, using the same notation and terminology, with references to material in the book, etc. Students find them an enjoyable and effective learning aid.

INTERACTIVE ROUTINES

Another key tutorial feature of the OR Courseware is a set of interactive routines implemented in Excel spreadsheets and/or Visual Basic. These routines can be viewed with recent versions of Microsoft Excel such as Excel 97, 98 (for Macintosh), or 2000. Each one is a self-contained routine that uses prompts or help files to provide the necessary information for execution. Either Excel spreadsheets or graphic interfaces are supplied to allow easy entry of problem data.

Each of these routines enables you to *interactively execute* one of the algorithms of OR. While viewing all relevant information on the computer screen, you make the decision on how the next step of the algorithm should be performed, and then the computer does all the necessary number crunching to execute that step. When a previous mistake is discovered, the routine allows you to quickly backtrack to correct the mistake. To get you started properly, the computer points out any mistake made on the first iteration (where possible). When done, you can print out all the work performed to turn in for homework.

In our judgment, these interactive routines provide the "right" way in this computer age for students to do homework designed to help them learn the algorithms of OR. The routines enable you to focus on concepts rather than mindless number crunching, thereby making the learning process far more efficient and effective as well as stimulating. They also point you in the right direction, including organizing the work to be done. However, the routines do not do the thinking for you. As in any good homework assignment, you are allowed to make mistakes (and to learn from those mistakes), so that hard thinking will need to be done to try to stay on the right path. We have been careful in designing the division of labor between the computer and the student to provide an efficient, complete learning process.

SPECIAL AUTOMATIC ROUTINES

Once you have learned the logic of a particular algorithm with the help of an interactive routine, you will want to be able to apply the algorithm quickly with an automatic routine thereafter. Such a routine is provided by one or more of the software packages discussed below for most of the algorithms described in this book. However, for a few algorithms that are not included in these commercial packages, we have provided special automatic routines in the OR Courseware. Like the interactive routines, these automatic routines are implemented in Excel spreadsheets and/or Visual Basic for viewing with a recent version of Excel.

EXCEL FILES

The OR Courseware includes a separate Excel file for nearly every chapter in this book. Each file typically includes several spreadsheets that will help you formulate and solve the various kinds of models described in the chapter. Two types of spreadsheets are included. First, each time an example is presented that can be solved using Excel, the complete spreadsheet formulation and solution is given in that chapter's Excel file. This provides a convenient reference, or even useful templates, when you set up spreadsheets to solve similar problems with the Excel Solver (or the Premium Solver discussed in the next subsection). Second, for many of the models in the book, template files are provided that already include all the equations necessary to solve the model. You simply enter the data for the model and the solution is immediately calculated.

EXCEL ADD-INS

Four Excel add-ins are included in OR Courseware. One is *Premium Solver* for Education (Version 3.5), which is a more powerful version of the standard Solver in Excel. It works with Excel 5, 95, 97, and 2000 on Windows systems (but not with Excel 98 for Macintosh). Premium Solver offers four times the capacity (800 decision variables) of the standard Solver for linear programming problems, and twice the capacity (400 decision variables) for nonlinear programming problems, plus solution speeds 3 to 10 times faster than the standard Solver. A product of the same organization that developed the standard Solver in Excel (Frontline Systems Inc.), Premium Solver is fully upward compatible with the standard Solver. The organization's website is www.frontsys.com. Technical support currently is provided at (775) 831-0300 or by e-mail at ‹info@frontsys.com›.

The other three Excel add-ins are academic versions of *SensIt* (introduced in Sec. 15.2), *TreePlan* (introduced in Sec. 15.4), and *RiskSim* (introduced in Sec. 22.6). All are shareware developed by Professor Michael R. Middleton for Excel 5, 95, 97, 98, and 2000 for Windows and Macintosh. Documentation is included on the CD-ROM for all three add-ins. The accompanying website is www.usfca.edu/fac-staff/middleton. This software is shareware, so those desiring to use it after the course should register and pay the shareware fee.

As with any Excel add-in, each of these add-ins needs to be installed in Excel before it is operational. (The same is true for the standard Excel Solver.) Installation instructions are included in the OR Courseware for each one.

Another Excel add-in discussed extensively in Sec. 22.6 is *@RISK* for simulation, from Palisade Corporation. Although Palisade declined to make this add-in available on our CD-ROM, it can be downloaded from the website, www.palisade.com, for a 10-day trial period.

MPL/CPLEX

As discussed at length in Secs. 3.7 and 4.8, MPL is a state-of-the-art modeling language and its prime solver CPLEX is a particularly prominent and powerful solver. The student version of MPL and CPLEX is included in the OR Courseware. Although this student version is limited to *much* smaller problems than the massive linear, integer, and quadratic programming problems commonly solved in practice by the full version, it still can handle up to 300 functional constraints and 300 decision variables (including any integer variables). The system requirements for the student version are an IBM-compatible PC with a 486 or Pentium processor, 16 Mb of memory, 4 Mb of free hard-disk space, and Microsoft Windows 95/98, NT (3.51 or higher), or 2000.

The CD-ROM provides an extensive MPL tutorial and documentation, as well as MPL/CPLEX formulations and solutions for virtually every example in the book to which they can be applied. Also included in the OR Courseware is the student version of OptiMax 2000, which enables fully integrating MPL models into Excel and solving with CPLEX. In addition, the powerful nonlinear programming solver CONOPT is included in MPL for solving such problems.

The website for further exploring MPL and its solvers, or for downloading updated student versions of MPL/CPLEX is, www.maximal-usa.com.

LINGO/LINDO FILES

This book also features the popular modeling language LINGO (see especially Appendix 3.1 and the end of Sec. 3.7) and the companion solver LINDO (see Sec. 4.8 and Appendix 4.1). Although they were not available for inclusion in the OR Courseware, student versions of both LINGO and LINDO (as well as the companion spreadsheet solver *What's Best*) can be downloaded from the website, www.lindo.com. Designed for use on a Windows platform, each of these downloads currently can handle up to 150 functional constraints and 300 decision variables. In the case of integer programming or nonlinear programming, they are restricted to 30 integer variables or 30 nonlinear variables. (Extended versions of this software can solve vastly larger problems.)

The OR Courseware includes extensive LINGO/LINDO files or (when LINDO is not relevant) LINGO files for many of the chapters. Each file provides the LINGO and LINDO models and solutions for the various examples in the chapter to which they can be applied. The solutions often are displayed in a What's Best spreadsheet. The CD-ROM also provides LINGO and LINDO tutorials.

MICROSOFT PROJECT

Chapter 10 (especially Sec. 10.2) describes how Microsoft Project can be used to help construct and evaluate a project network while using PERT/CPM. The version included in the OR Courseware is Microsoft Project 98, which is designed for use on a Windows platform. (Microsoft also markets an earlier version, Project 4, for Macintosh). The CD-ROM includes a document READTH~1.HTM in the Project folder with various links that provide extensive documentation of the software. The OR Courseware also includes an MS Project folder that has the main kinds of worksheets that Microsoft Project would generate for the prototype example of Chapter 10.

UPDATES

The software world evolves very rapidly during the lifetime of one edition of a textbook. We believe that the documentation provided in this appendix is accurate at the time of this writing, but changes inevitably will occur as time passes.

With each new printing of this edition, we plan to provide updated versions of the software in the OR Courseware whenever feasible. You can also visit the book's website, www.mhhe.com/hillier, for information about software updates.

APPENDIX 2
CONVEXITY

As introduced in Chap. 13, the concept of *convexity* is frequently used in OR work, especially in the area of nonlinear programming. Therefore, we further introduce the properties of convex or concave functions and convex sets here.

CONVEX OR CONCAVE FUNCTIONS OF A SINGLE VARIABLE

We begin with definitions.

> **Definitions:** A *function* of a single variable $f(x)$ is a **convex function** if, for *each* pair of values of x, say, x' and x'' ($x' < x''$),
>
> $$f[\lambda x'' + (1 - \lambda)x'] \leq \lambda f(x'') + (1 - \lambda)f(x')$$
>
> for all values of λ such that $0 < \lambda < 1$. It is a **strictly convex function** if $\leq$ can be replaced by $<$. It is a **concave function** (or a **strictly concave function**) if this statement holds when $\leq$ is replaced by $\geq$ (or by $>$).

This definition of a convex function has an enlightening geometric interpretation. Consider the graph of the function $f(x)$ drawn as a function of x, as illustrated in Fig. A2.1 for a function $f(x)$ that decreases for $x < 1$, is constant for $1 \leq x \leq 2$, and increases for $x > 2$. Then $[x', f(x')]$ and $[x'', f(x'')]$ are two points on the graph of $f(x)$, and $[\lambda x'' + (1 - \lambda)x', \lambda f(x'') + (1 - \lambda)f(x')]$ represents the various points on the line segment between these two points (but excluding these endpoints) when $0 < \lambda < 1$. Thus, the $\leq$ inequality in the definition indicates that this line segment lies entirely above or on the graph of the function, as in Fig. A2.1. Therefore, $f(x)$ is *convex* if, for *each* pair of points on the graph of $f(x)$, the line segment joining these two points lies entirely above or on the graph of $f(x)$.

For example, the particular choice of x' and x'' shown in Fig. A2.1 results in the entire line segment (except the two endpoints) lying *above* the graph of $f(x)$. This also occurs for other choices of x' and x'' where either $x' < 1$ or $x'' > 2$ (or both). If $1 \leq x' < x'' \leq 2$, then the entire line segment lies *on* the graph of $f(x)$. Therefore, this $f(x)$ is convex.

This geometric interpretation indicates that $f(x)$ is convex if it only "bends upward" whenever it bends at all. (This condition is sometimes referred to as *concave upward,* as opposed to *concave downward* for a concave function.) To be more precise, if $f(x)$ possesses a second derivative everywhere, then $f(x)$ is convex if and only if $d^2f(x)/dx^2 \geq 0$ for all possible values of x.

The definitions of a *strictly convex function,* a *concave function,* and a *strictly concave function* also have analogous geometric interpretations. These interpretations are summarized below in terms of the second derivative of the function, which provides a convenient test of the status of the function.

> **Convexity test for a function of a single variable:** Consider any function of a single variable $f(x)$ that possesses a second derivative at all possible values of x. Then $f(x)$ is
>
> 1. *Convex* if and only if $\dfrac{d^2f(x)}{dx^2} \geq 0$ for all possible values of x
> 2. *Strictly convex* if and only if $\dfrac{d^2f(x)}{dx^2} > 0$ for all possible values of x

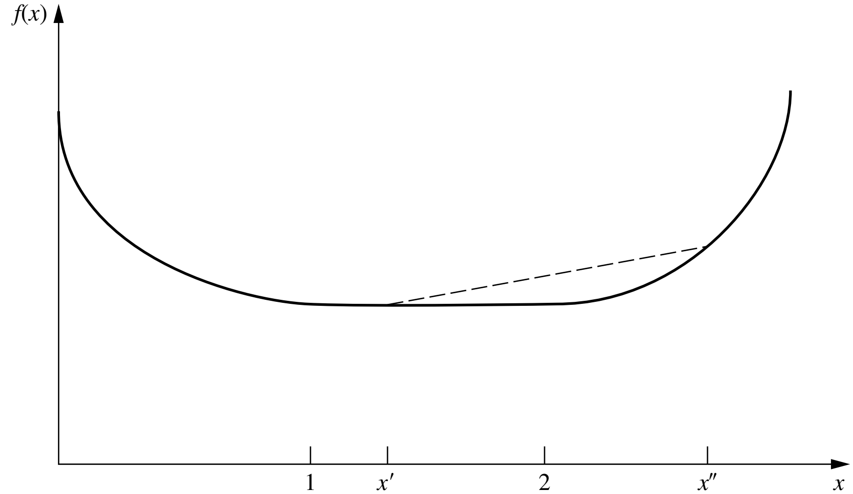


FIGURE A2.1
A convex function.

3. *Concave* if and only if $\frac{d^2f(x)}{dx^2} \leq 0$ for all possible values of x
4. *Strictly concave* if and only if $\frac{d^2f(x)}{dx^2} < 0$ for all possible values of x

Note that a strictly convex function also is convex, but a convex function is *not* strictly convex if the second derivative equals zero for some values of x. Similarly, a strictly concave function is concave, but the reverse need not be true.

Figures A2.1 to A2.6 show examples that illustrate these definitions and this convexity test.

Applying this test to the function in Fig. A2.1, we see that as x is increased, the slope (first derivative) either increases (for $0 \leq x < 1$ and $x > 2$) or remains constant (for $1 \leq x_1 \leq 2$). Therefore, the second derivative always is nonnegative, which verifies that the function is convex. However, it is *not* strictly convex because the second derivative equals zero for $1 \leq x \leq 2$.

However, the function in Fig. A2.2 is strictly convex because its slope always is increasing so its second derivative always is greater than zero.

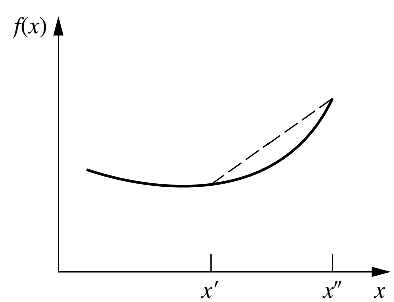
The piecewise linear function shown in Fig. A2.3 changes its slope at $x = 1$. Consequently, it does not possess a first or second derivative at this point, so the convexity test cannot be fully applied. (The fact that the second derivative equals zero for $0 \leq x < 1$ and $x > 1$ makes the function eligible to be either convex or concave, depending upon its behavior at $x = 1$.) Applying the definition of a concave function, we see that if $0 < x' < 1$ and $x'' > 1$ (as shown in Fig. A2.3), then the entire line segment joining $[x', f(x')]$ and $[x'', f(x'')]$ lies *below* the graph of $f(x)$, except for the two endpoints of the line segment. If either $0 \leq x' < x'' \leq 1$ or $1 \leq x' < x''$, then the entire line segment lies *on* the graph of $f(x)$. Therefore, $f(x)$ is concave (but *not* strictly concave).

The function in Fig. A2.4 is strictly concave because its second derivative always is less than zero.

As illustrated in Fig. A2.5, any linear function has its second derivative equal to zero everywhere and so is both convex and concave.

The function in Fig. A2.6 is *neither* convex nor concave because as x increases, the slope fluctuates between de-

FIGURE A2.2
A strictly convex function.



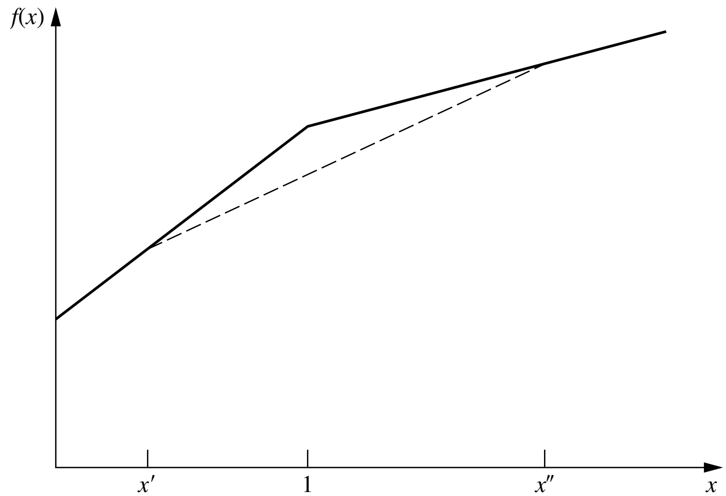


FIGURE A2.3
A concave function.

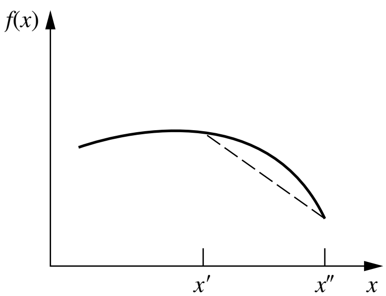


FIGURE A2.4
A strictly concave function.

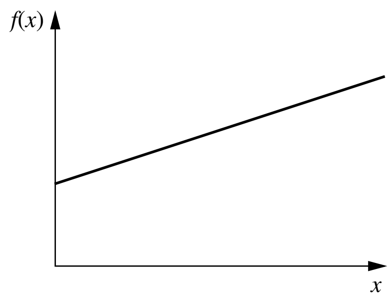


FIGURE A2.5
A function that is both convex and concave.

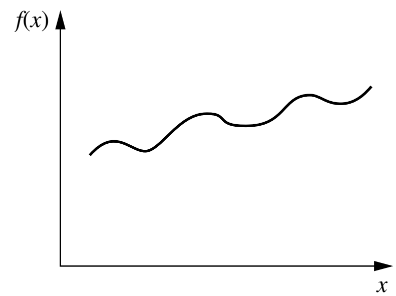


FIGURE A2.6
A function that is neither convex nor concave.

creasing and increasing so the second derivative fluctuates between being negative and positive.

CONVEX OR CONCAVE FUNCTIONS OF SEVERAL VARIABLES

The concept of a convex or concave function of a single variable also generalizes to functions of more than one variable. Thus, if $f(x)$ is replaced by $f(x_1, x_2, \ldots, x_n)$, the definition still applies if x is replaced everywhere by $(x_1, x_2, \ldots, x_n)$. Similarly, the corresponding geometric interpretation is still valid after generalization of the concepts of *points* and *line segments.* Thus, just as a particular value of (x, y) is interpreted as a point in two-dimensional space, each possible value of $(x_1, x_2, \ldots, x_m)$ may be thought of as a point in m-dimensional (Euclidean) space. By letting $m = n + 1$, the points on the graph of $f(x_1, x_2, \ldots, x_n)$ become the possible values of $[x_1, x_2, \ldots, x_n, f(x_1, x_2, \ldots, x_n)]$. Another point, $(x_1, x_2, \ldots, x_n, x_{n+1})$, is said to lie above, on, or below the graph of $f(x_1, x_2, \ldots, x_n)$, according to whether x_{n+1} is larger, equal to, or smaller than $f(x_1, x_2, \ldots, x_n)$, respectively.

Definition: The **line segment** joining any two points $(x_1', x_2', \ldots, x_m')$ and $(x_1'', x_2'', \ldots, x_m'')$ is the collection of points

$$(x_1, x_2, \ldots, x_m) = [\lambda x_1'' + (1 - \lambda)x_1', \lambda x_2'' + (1 - \lambda)x_2', \ldots, \lambda x_m'' + (1 - \lambda)x_m']$$

such that $0 \le \lambda \le 1$.

Thus, a line segment in m-dimensional space is a direct generalization of a line segment in two-dimensional space. For example, if

$$(x_1', x_2') = (2, 6), \qquad (x_1'', x_2'') = (3, 4),$$

then the line segment joining them is the collection of points

$$(x_1, x_2) = [3\lambda + 2(1 - \lambda), 4\lambda + 6(1 - \lambda)],$$

where $0 \leq \lambda \leq 1$.

> **Definition:** $f(x_1, x_2, \ldots, x_n)$ is a **convex function** if, for each pair of points on the graph of $f(x_1, x_2, \ldots, x_n)$, the line segment joining these two points lies entirely above or on the graph of $f(x_1, x_2, \ldots, x_n)$. It is a **strictly convex function** if this line segment actually lies entirely above this graph except at the endpoints of the line segment. **Concave functions** and **strictly concave functions** are defined in exactly the same way, except that *above* is replaced by *below.*

Just as the second derivative can be used (when it exists everywhere) to check whether a function of a single variable is convex, so second partial derivatives can be used to check functions of several variables, although in a more complicated way. For example, if there are two variables and all partial derivatives exist everywhere, then the convexity test assesses whether *all three quantities* in the first column of Table A2.1 satisfy the inequalities shown in the appropriate column for *all possible values* of (x_1, x_2).

When there are more than two variables, the convexity test is a generalization of the one shown in Table A2.1. For example, in mathematical terminology, $f(x_1, x_2, \ldots, x_n)$ is convex if and only if its $n \times n$ Hessian matrix is positive semidefinite for all possible values of $(x_1, x_2, \ldots, x_n)$.

To illustrate the convexity test for two variables, consider the function

$$f(x_1, x_2) = (x_1 - x_2)^2 = x_1^2 - 2x_1x_2 + x_2^2.$$

Therefore,

$$(1) \quad \frac{\partial^2 f(x_1, x_2)}{\partial x_1^2} \frac{\partial^2 f(x_1, x_2)}{\partial x_2^2} - \left[\frac{\partial^2 f(x_1, x_2)}{\partial x_1 \partial x_2}\right]^2 = 2(2) - (-2)^2 = 0,$$

$$(2) \quad \frac{\partial^2 f(x_1, x_2)}{\partial x_1^2} = 2 > 0,$$

$$(3) \quad \frac{\partial^2 f(x_1, x_2)}{\partial x_2^2} = 2 > 0.$$

Since ≥ 0 holds for all three conditions, $f(x_1, x_2)$ is convex. However, it is *not* strictly convex because the first condition only gives $= 0$ rather than > 0.

Now consider the negative of this function

$$g(x_1, x_2) = -f(x_1, x_2) = -(x_1 - x_2)^2 = -x_1^2 + 2x_1x_2 - x_2^2.$$

In this case,

$$(4) \quad \frac{\partial^2 g(x_1, x_2)}{\partial x_1^2} \frac{\partial^2 g(x_1, x_2)}{\partial x_2^2} - \left[\frac{\partial^2 g(x_1, x_2)}{\partial x_1 \partial x_2}\right]^2 = -2(-2) - 2^2 = 0,$$

$$(5) \quad \frac{\partial^2 g(x_1, x_2)}{\partial x_1^2} = -2 < 0,$$

$$(6) \quad \frac{\partial^2 g(x_1, x_2)}{\partial x_2^2} = -2 < 0.$$

TABLE A2.1 Convexity test for a function of two variables

Quantity	Convex	Strictly Convex	Concave	Strictly Concave
$\frac{\partial^2 f(x_1, x_2)}{\partial x_1^2} \frac{\partial^2 f(x_1, x_2)}{\partial x_2^2} - \left[\frac{\partial^2 f(x_1, x_2)}{\partial x_1 \partial x_2}\right]^2$	≥ 0	> 0	≥ 0	> 0
$\frac{\partial^2 f(x_1, x_2)}{\partial x_1^2}$	≥ 0	> 0	≤ 0	< 0
$\frac{\partial^2 f(x_1, x_2)}{\partial x_2^2}$	≥ 0	> 0	≤ 0	< 0
Values of (x_1, x_2)	All possible values			

Because ≥ 0 holds for the first condition and ≤ 0 holds for the other two, $g(x_1, x_2)$ is a concave function. However, it is *not* strictly concave since the first condition gives $= 0$.

Thus far, convexity has been treated as a general property of a function. However, many nonconvex functions do satisfy the conditions for convexity over certain intervals for the respective variables. Therefore, it is meaningful to talk about a function being convex over a certain region. For example, a function is said to be convex within a neighborhood of a specified point if its second derivative or partial derivatives satisfy the conditions for convexity at that point. This concept is useful in Appendix 3.

Finally, two particularly important properties of convex or concave functions should be mentioned. First, if $f(x_1, x_2, \ldots, x_n)$ is a convex function, then $g(x_1, x_2, \ldots, x_n) = -f(x_1, x_2, \ldots, x_n)$ is a concave function, and vice versa, as illustrated by the above example where $f(x_1, x_2) = (x_1 - x_2)^2$. Second, the sum of convex functions is a convex function, and the sum of concave functions is a concave function. To illustrate,

$$f_1(x_1) = x_1^4 + 2x_1^2 - 5x_1$$

and

$$f_2(x_1, x_2) = x_1^2 + 2x_1x_2 + x_2^2$$

are both convex functions, as you can verify by calculating their second derivatives. Therefore, the sum of these functions

$$f(x_1, x_2) = x_1^4 + 3x_1^2 - 5x_1 + 2x_1x_2 + x_2^2$$

is a convex function, whereas its negative

$$g(x_1, x_2) = -x_1^4 - 3x_1^2 + 5x_1 - 2x_1x_2 - x_2^2,$$

is a concave function.

CONVEX SETS

The concept of a convex function leads quite naturally to the related concept of a **convex set.** Thus, if $f(x_1, x_2, \ldots, x_n)$ is a convex function, then the collection of points that lie above or on the graph of $f(x_1, x_2, \ldots, x_n)$ forms a convex set. Similarly, the collection of points that lie below or on the graph of a concave function is a convex set. These cases are illustrated in Figs. A2.7 and A2.8 for the case of a single independent variable. Furthermore, convex sets have the important property that, for any given group of convex sets, the collection of points that lie in all of them (i.e., the intersection of these convex sets) is also a convex set. Therefore, the collection of points that lie both above or on a convex function and below or on a concave function is a convex set, as illustrated in Fig. A2.9. Thus, convex sets may be viewed intuitively as a collection of points whose bottom boundary is a convex function and whose top boundary is a concave function.

Although describing convex sets in terms of convex and concave functions may be helpful for developing intuition about their nature, their actual definition has nothing to do (directly) with such functions.

> **Definition:** A **convex set** is a collection of points such that, for each pair of points in the collection, the entire line segment joining these two points is also in the collection.

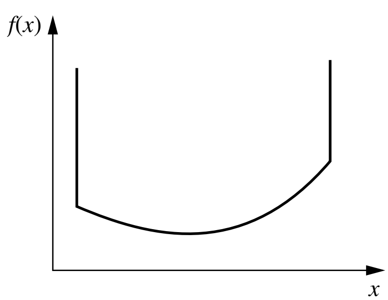


FIGURE A2.7
Example of a convex set determined by a convex function.

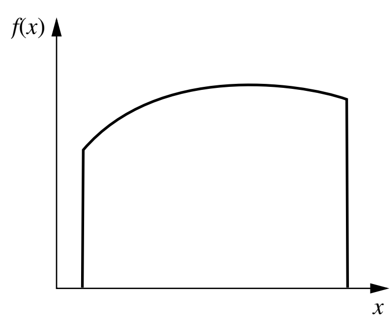


FIGURE A2.8
Example of a convex set determined by a concave function.

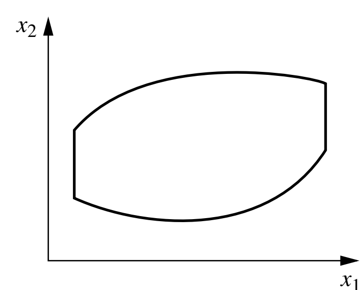


FIGURE A2.9
Example of a convex set determined by both convex and concave functions.

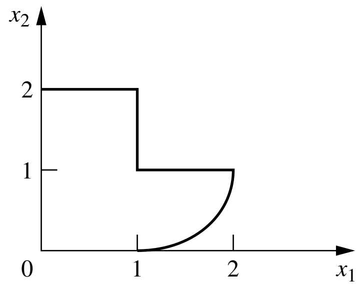


FIGURE A2.10
Example of a set that is not convex.

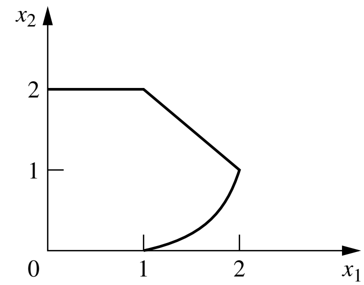


FIGURE A2.11
Example of a convex set.

The distinction between nonconvex sets and convex sets is illustrated in Figs. A2.10 and A2.11. Thus, the set of points shown in Fig. A2.10 is not a convex set because there exist many pairs of these points, for example, (1, 2) and (2, 1), such that the line segment between them does not lie entirely within the set. This is not the case for the set in Fig. A2.11, which is convex.

In conclusion, we introduce the useful concept of an extreme point of a convex set.

Definition: An **extreme point** of a convex set is a point in the set that does not lie on any line segment that joins two other points in the set.

Thus, the extreme points of the convex set in Fig. A2.11 are (0, 0), (0, 2), (1, 2), (2, 1), (1, 0), and all the infinite number of points on the boundary between (2, 1) and (1, 0). If this particular boundary were a line segment instead, then the set would have only the five listed extreme points.

APPENDIX 3

CLASSICAL OPTIMIZATION METHODS

This appendix reviews the classical methods of calculus for finding a solution that maximizes or minimizes (1) a function of a single variable, (2) a function of several variables, and (3) a function of several variables subject to equality constraints on the values of these variables. It is assumed that the functions considered possess continuous first and second derivatives and partial derivatives everywhere. Some of the concepts discussed next have been introduced briefly in Secs. 13.2 and 13.3.

UNCONSTRAINED OPTIMIZATION OF A FUNCTION OF A SINGLE VARIABLE

Consider a function of a single variable, such as that shown in Fig. A3.1. A necessary condition for a particular solution $x = x^*$ to be either a minimum or a maximum is that

$$\frac{df(x)}{dx} = 0 \qquad \text{at } x = x^*.$$

Thus, in Fig. A3.1 there are five solutions satisfying these conditions. To obtain more information about these five **critical points,** it is necessary to examine the second derivative. Thus, if

$$\frac{d^2f(x)}{dx^2} > 0 \qquad \text{at } x = x^*,$$

then x^* must be at least a **local minimum** [that is, $f(x^*) \leq f(x)$ for all x sufficiently close to x^*]. Using the language introduced in Appendix 2, we can say that x^* must be a local minimum if $f(x)$ is *strictly convex* within a neighborhood of x^*. Similarly, a sufficient condition for x^* to be a **local maximum** (given that it satisfies the necessary condition) is that $f(x)$ be *strictly concave* within a neighborhood of x^* (that is, the second derivative is *negative* at x^*). If the second derivative is zero, the issue is not resolved (the point may even be an *inflection point*), and it is necessary to examine higher derivatives.

To find a **global minimum** [i.e., a solution x^* such that $f(x^*) \leq f(x)$ for all x], it is necessary to compare the local minima and identify the one that yields the smallest value of $f(x)$. If this value is less than $f(x)$ as $x \to -\infty$ and as $x \to +\infty$ (or at the endpoints of the function, if it is defined only over a finite interval), then this point is a global minimum. Such a point is shown in Fig. A3.1, along with the **global maximum,** which is identified in an analogous way.

However, if $f(x)$ is known to be either a convex or a concave function (see Appendix 2 for a description of such functions), the analysis becomes much simpler. In particular, if $f(x)$ is a *convex* function, such as the one shown in Fig. A2.1, then any solution x^* such that

$$\frac{df(x)}{dx} = 0 \qquad \text{at } x = x^*$$

is known automatically to be a *global minimum*. In other words, this condition is not only a *necessary* but also a *sufficient* condition for a global minimum of a convex func-

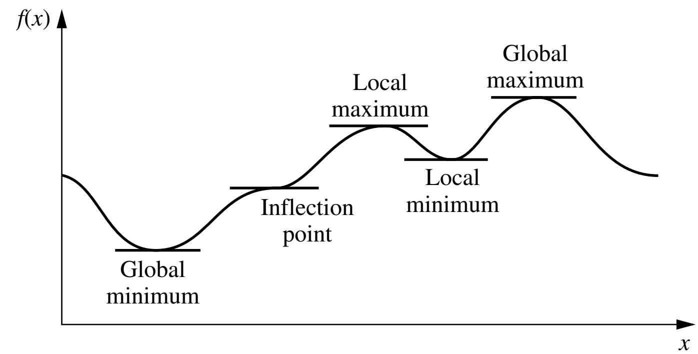


FIGURE A3.1
A function having several maxima and minima.

tion. This solution need not be unique, since there could be a tie for the global minimum over a single interval where the derivative is zero. On the other hand, if $f(x)$ actually is *strictly convex,* then this solution must be the only global minimum. (However, if the function is either always decreasing or always increasing, so the derivative is nonzero for all values of x, then there will be no global minimum at a finite value of x.)

Similarly, if $f(x)$ is a *concave* function, then having

$$\frac{df(x)}{dx} = 0 \qquad \text{at } x = x^*$$

becomes both a *necessary* and *sufficient* condition for x^* to be a *global maximum.*

UNCONSTRAINED OPTIMIZATION OF A FUNCTION OF SEVERAL VARIABLES

The analysis for an unconstrained function of several variables $f(\mathbf{x})$, where $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, is similar. Thus, a *necessary* condition for a solution $\mathbf{x} = \mathbf{x}^*$ to be either a minimum or a maximum is that

$$\frac{\partial f(\boldsymbol{x})}{\partial x_j} = 0 \qquad \text{at } \mathbf{x} = \mathbf{x}^*, \text{ for } j = 1, 2, \ldots, n.$$

After the critical points that satisfy this condition are identified, each such point is then classified as a local minimum or maximum if the function is *strictly convex* or *strictly concave,* respectively, within a neighborhood of the point. (Additional analysis is required if the function is neither.) The *global minimum* and *maximum* would be found by comparing the local minima and maxima and then checking the value of the function as some of the variables approach $-\infty$ or $+\infty$. However, if the function is known to be *convex* or *concave,* then a critical point must be a *global minimum* or a *global maximum,* respectively.

CONSTRAINED OPTIMIZATION WITH EQUALITY CONSTRAINTS

Now consider the problem of finding the *minimum* or *maximum* of the function $f(\mathbf{x})$, subject to the restriction that $\mathbf{x}$ must satisfy all the equations

$$\begin{aligned} g_1(\mathbf{x}) &= b_1 \\ g_2(\mathbf{x}) &= b_2 \\ &\vdots \\ g_m(\mathbf{x}) &= b_m, \end{aligned}$$

where $m < n$. For example, if $n = 2$ and $m = 1$, the problem might be

$$\text{Maximize} \qquad f(x_1, x_2) = x_1^2 + 2x_2,$$

subject to

$$g(x_1, x_2) = x_1^2 + x_2^2 = 1.$$

In this case, (x_1, x_2) is restricted to be on the circle of radius 1, whose center is at the origin, so that the goal is to find the point on this circle that yields the largest value of $f(x_1, x_2)$. This example will be solved after a general approach to the problem is outlined.

A classical method of dealing with this problem is the **method of Lagrange multipliers.** This procedure begins by formulating the **Lagrangian function**

$$h(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) - \sum_{i=1}^{m} \lambda_i[g_i(\mathbf{x}) - b_i],$$

where the new variables $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_m)$ are called *Lagrange multipliers.* Notice the key fact that for the *feasible* values of $\mathbf{x}$,

$$g_i(\mathbf{x}) - b_i = 0, \qquad \text{for all } i,$$

so $h(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x})$. Therefore, it can be shown that if $(\mathbf{x}, \boldsymbol{\lambda}) = (\mathbf{x}^*, \boldsymbol{\lambda}^*)$ is a *local* or *global minimum* or *maximum* for the unconstrained function $h(\mathbf{x}, \boldsymbol{\lambda})$, then $\mathbf{x}^*$ is a corresponding *critical point* for the original problem. As a result, the method now reduces to analyzing $h(\mathbf{x}, \boldsymbol{\lambda})$ by the procedure just described for unconstrained optimization. Thus, the $n + m$ partial derivatives would be set equal to zero

$$\frac{\partial h}{\partial x_j} = \frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j} = 0, \qquad \text{for } j = 1, 2, \ldots, n,$$

$$\frac{\partial h}{\partial \lambda_i} = -g_i(\mathbf{x}) + b_i = 0, \qquad \text{for } i = 1, 2, \ldots, m,$$

and then the critical points would be obtained by solving these equations for $(\mathbf{x}, \boldsymbol{\lambda})$. Notice that the last m equations are equivalent to the constraints in the original problem, so only feasible solutions are considered. After further analysis to identify the *global minimum* or *maximum* of $h(\cdot)$, the resulting value of $\mathbf{x}$ is then the desired solution to the original problem.

From a practical computational viewpoint, the method of Lagrange multipliers is not a particularly powerful procedure. It is often essentially impossible to solve the equations to obtain the critical points. Furthermore, even when the points can be obtained, the number of critical points may be so large (often infinite) that it is impractical to attempt to identify a global minimum or maximum. However, for certain types of small problems, this method can sometimes be used successfully.

To illustrate, consider the example introduced earlier. In this case,

$$h(x_1, x_2) = x_1^2 + 2x_2 - \lambda(x_1^2 + x_2^2 - 1),$$

so that

$$\frac{\partial h}{\partial x_1} = 2x_1 - 2\lambda x_1 = 0,$$

$$\frac{\partial h}{\partial x_2} = 2 - 2\lambda x_2 = 0,$$

$$\frac{\partial h}{\partial \lambda} = -(x_1^2 + x_2^2 - 1) = 0.$$

The first equation implies that either $\lambda = 1$ or $x_1 = 0$. If $\lambda = 1$, then the other two equations imply that $x_2 = 1$ and $x_1 = 0$. If $x_1 = 0$, then the third equation implies that $x_2 = \pm 1$. Therefore, the two critical points for the original problem are $(x_1, x_2) = (0, 1)$ *and* $(0, -1)$. Thus, it is apparent that these points are the global maximum and minimum, respectively.

THE DERIVATIVE OF A DEFINITE INTEGRAL

In presenting the classical optimization methods just described, we have assumed that you are already familiar with derivatives and how to obtain them. However, there is a special case of importance in OR work that warrants additional explanation, namely, the derivative of a definite integral. In particular, consider how to find the derivative of the function

$$F(y) = \int_{g(y)}^{h(y)} f(x, y)\, dx,$$

where $g(y)$ and $h(y)$ are the limits of integration expressed as functions of y.

To begin, suppose that these limits of integration are constants, so that $g(y) = a$ and $h(y) = b$, respectively. For this special case, it can be shown that, given the regularity conditions assumed at the beginning of this appendix, the derivative is

$$\frac{d}{dy} \int_a^b f(x, y)\, dx = \int_a^b \frac{\partial f(x, y)}{\partial y}\, dx.$$

For example, if $f(x, y) = e^{-xy}$, $a = 0$, and $b = \infty$, then

$$\frac{d}{dy} \int_0^\infty e^{-xy}\, dx = \int_0^\infty (-x)e^{-xy}\, dx = -\frac{1}{y^2}$$

at any positive value of y. Thus, the intuitive procedure of interchanging the order of differentiation and integration is valid for this case.

However, finding the derivative becomes a little more complicated than this when the limits of integration are functions. In particular,

$$\frac{d}{dy}\int_{g(y)}^{h(y)} f(x, y)\, dx = \int_{g(y)}^{h(y)} \frac{\partial f(x, y)}{\partial y}\, dx + f(h(y), y)\,\frac{dh(y)}{dy} - f(g(y), y)\,\frac{dg(y)}{dy},$$

where $f(h(y), y)$ is obtained by writing out $f(x, y)$ and then replacing x by $h(y)$ wherever it appears, and similarly for $f(g(y), y)$. To illustrate, if $f(x, y) = x^2y^3$, $g(y) = y$, and $h(y) = 2y$, then

$$\begin{aligned}\frac{d}{dy}\int_{y}^{2y} x^2y^3\, dx &= \int_{y}^{2y} 3x^2y^2\, dx + (2y)^2y^3(2) - y^2y^3(1)\\ &= 14y^5\end{aligned}$$

at any positive value of y.

APPENDIX 4

MATRICES AND MATRIX OPERATIONS

A **matrix** is a rectangular array of numbers. For example,

$$\mathbf{A} = \begin{bmatrix} 2 & 5 \\ 3 & 0 \\ 1 & 1 \end{bmatrix}$$

is a 3×2 matrix (where 3×2 is said "3 by 2") because it is a rectangular array of numbers with three rows and two columns. (Matrices are denoted in this book by **boldface capital letters.**) The numbers in the rectangular array are called the **elements** of the matrix. For example,

$$\mathbf{B} = \begin{bmatrix} 1 & 2.4 & 0 & \sqrt{3} \\ -4 & 2 & -1 & 15 \end{bmatrix}$$

is a 2×4 matrix whose elements are 1, 2.4, 0, $\sqrt{3}$, -4, 2, -1, and 15. Thus, in more general terms,

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = \|a_{ij}\|$$

is an $m \times n$ matrix, where $a_{11}, \ldots, a_{mn}$ represent the numbers that are the elements of this matrix; $\|a_{ij}\|$ is shorthand notation for identifying the matrix whose element in row i and column j is a_{ij} for every $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$.

MATRIX OPERATIONS

Because matrices do not possess a numerical value, they cannot be added, multiplied, and so on as if they were individual numbers. However, it is sometimes desirable to perform certain manipulations on arrays of numbers. Therefore, rules have been developed for performing operations on matrices that are analogous to arithmetic operations. To describe these, let $\mathbf{A} = \|a_{ij}\|$ and $\mathbf{B} = \|b_{ij}\|$ be two matrices having the same number of rows and the same number of columns. (We shall change this restriction on the size of **A** and **B** later when discussing matrix multiplication.)

Matrices **A** and **B** are said to be *equal* (**A = B**) if and only if *all* the corresponding elements are equal ($a_{ij} = b_{ij}$ for all i and j).

The operation of *multiplying a matrix by a number* (denote this number by k) is performed by multiplying each element of the matrix by k, so that

$$k\mathbf{A} = \|ka_{ij}\|.$$

For example,

$$3\begin{bmatrix} 1 & \frac{1}{3} & 2 \\ 5 & 0 & -3 \end{bmatrix} = \begin{bmatrix} 3 & 1 & 6 \\ 15 & 0 & -9 \end{bmatrix}.$$

To add two matrices **A** and **B**, simply add the corresponding elements, so that

$$\mathbf{A} + \mathbf{B} = \|a_{ij} + b_{ij}\|.$$

To illustrate,

$$\begin{bmatrix} 5 & 3 \\ 1 & 6 \end{bmatrix} + \begin{bmatrix} 2 & 0 \\ 3 & 1 \end{bmatrix} = \begin{bmatrix} 7 & 3 \\ 4 & 7 \end{bmatrix}.$$

Similarly, *subtraction* is done as follows:

$$\mathbf{A} - \mathbf{B} = \mathbf{A} + (-1)\mathbf{B},$$

so that

$$\mathbf{A} - \mathbf{B} = \|a_{ij} - b_{ij}\|.$$

For example,

$$\begin{bmatrix} 5 & 3 \\ 1 & 6 \end{bmatrix} - \begin{bmatrix} 2 & 0 \\ 3 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 3 \\ -2 & 5 \end{bmatrix}.$$

Note that, with the exception of multiplication by a number, all the preceding operations are defined only when the two matrices involved are the same size. However, all of these operations are straightforward because they involve performing only the same comparison or arithmetic operation on the corresponding elements of the matrices.

There exists one additional elementary operation that has not been defined—**matrix multiplication**—but it is considerably more complicated. To find the element in row i, column j of the matrix resulting from multiplying matrix $\mathbf{A}$ times matrix $\mathbf{B}$, it is necessary to multiply each element in row i of $\mathbf{A}$ by the corresponding element in column j of $\mathbf{B}$ and then to add these products. To do this element-by-element multiplication, we need the following restriction on the sizes of $\mathbf{A}$ and $\mathbf{B}$:

> Matrix multiplication $\mathbf{AB}$ is defined if and only if the *number of columns* of $\mathbf{A}$ equals the *number of rows* of $\mathbf{B}$.

Thus, if $\mathbf{A}$ is an $m \times n$ matrix and $\mathbf{B}$ is an $n \times s$ matrix, then their product is

$$\mathbf{AB} = \left\| \sum_{k=1}^{n} a_{ik} b_{kj} \right\|,$$

where this product is an $m \times s$ matrix. However, if $\mathbf{A}$ is an $m \times n$ matrix and $\mathbf{B}$ is an $r \times s$ matrix, where $n \neq r$, then $\mathbf{AB}$ is not defined.

To illustrate matrix multiplication,

$$\begin{bmatrix} 1 & 2 \\ 4 & 0 \\ 2 & 3 \end{bmatrix} \begin{bmatrix} 3 & 1 \\ 2 & 5 \end{bmatrix} = \begin{bmatrix} 1(3) + 2(2) & 1(1) + 2(5) \\ 4(3) + 0(2) & 4(1) + 0(5) \\ 2(3) + 3(2) & 2(1) + 3(5) \end{bmatrix} = \begin{bmatrix} 7 & 11 \\ 12 & 4 \\ 12 & 17 \end{bmatrix}.$$

On the other hand, if one attempts to multiply these matrices in the reverse order, the resulting product

$$\begin{bmatrix} 3 & 1 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 0 \\ 2 & 3 \end{bmatrix}$$

is not even defined.

Even when both $\mathbf{AB}$ and $\mathbf{BA}$ are defined,

$$\mathbf{AB} \neq \mathbf{BA}$$

in general. Thus, *matrix multiplication* should be viewed as a specially designed operation whose properties are quite different from those of *arithmetic multiplication.* To understand why this special definition was adopted, consider the following system of equations:

$$\begin{aligned} 2x_1 - x_2 + 5x_3 + x_4 &= 20 \\ x_1 + 5x_2 + 4x_3 + 5x_4 &= 30 \\ 3x_1 + x_2 - 6x_3 + 2x_4 &= 20. \end{aligned}$$

Rather than write out these equations as shown here, they can be written much more concisely in matrix form as

$$\mathbf{Ax} = \mathbf{b},$$

where

$$\mathbf{A} = \begin{bmatrix} 2 & -1 & 5 & 1 \\ 1 & 5 & 4 & 5 \\ 3 & 1 & -6 & 2 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 20 \\ 30 \\ 20 \end{bmatrix}.$$

It is this kind of multiplication for which matrix multiplication is designed.

Carefully note that *matrix division* is *not* defined.

Although the matrix operations described here do not possess certain of the properties of arithmetic operations, they do satisfy these laws

$$\begin{aligned} \mathbf{A} + \mathbf{B} &= \mathbf{B} + \mathbf{A}, \\ (\mathbf{A} + \mathbf{B}) + \mathbf{C} &= \mathbf{A} + (\mathbf{B} + \mathbf{C}), \\ \mathbf{A}(\mathbf{B} + \mathbf{C}) &= \mathbf{AB} + \mathbf{AC}, \\ \mathbf{A}(\mathbf{BC}) &= (\mathbf{AB})\mathbf{C}, \end{aligned}$$

when the relative sizes of these matrices are such that the indicated operations are defined.

Another type of matrix operation, which has no arithmetic analog, is the **transpose operation.** This operation involves nothing more than interchanging the rows and columns of the matrix, which is frequently useful for performing the multiplication operation in the desired way. Thus, for any matrix $\mathbf{A} = \|a_{ij}\|$, its transpose $\mathbf{A}^T$ is

$$\mathbf{A}^T = \|a_{ji}\|.$$

For example, if

$$\mathbf{A} = \begin{bmatrix} 2 & 5 \\ 1 & 3 \\ 4 & 0 \end{bmatrix},$$

then

$$\mathbf{A}^T = \begin{bmatrix} 2 & 1 & 4 \\ 5 & 3 & 0 \end{bmatrix}.$$

SPECIAL KINDS OF MATRICES

In arithmetic, 0 and 1 play a special role. There also exist special matrices that play a similar role in matrix theory. In particular, the matrix that is analogous to 1 is the **identity matrix I,** which is a *square* matrix whose elements are 0s except for 1s along the main diagonal. Thus,

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

The number of rows or columns of **I** can be specified as desired. The analogy of **I** to 1 follows from the fact that for any matrix **A**,

$$\mathbf{IA} = \mathbf{A} = \mathbf{AI},$$

where **I** is assigned the appropriate number of rows and columns in each case for the multiplication operation to be defined.

Similarly, the matrix that is analogous to 0 is the **null matrix 0,** which is a matrix of any size whose elements are *all* 0s. Thus,

$$\mathbf{0} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

Therefore, for any matrix **A**,

$$\mathbf{A} + \mathbf{0} = \mathbf{A}, \qquad \mathbf{A} - \mathbf{A} = \mathbf{0}, \qquad \text{and}$$
$$\mathbf{0A} = \mathbf{0} = \mathbf{A0},$$

where **0** is the appropriate size in each case for the operations to be defined.

On certain occasions, it is useful to partition a matrix into several smaller matrices, called **submatrices.** For example, one possible way of partitioning a 3×4 matrix would be

$$\mathbf{A} = \left[\begin{array}{c|ccc} a_{11} & a_{12} & a_{13} & a_{14} \\ \hline a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{array}\right] = \begin{bmatrix} a_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix},$$

where

$$\mathbf{A}_{12} = [a_{12}, \quad a_{13}, \quad a_{14}], \qquad \mathbf{A}_{21} = \begin{bmatrix} a_{21} \\ a_{31} \end{bmatrix},$$

$$\mathbf{A}_{22} = \begin{bmatrix} a_{22} & a_{23} & a_{24} \\ a_{32} & a_{33} & a_{34} \end{bmatrix}$$

all are submatrices. Rather than perform operations element by element on such partitioned matrices, we can do them in terms of the submatrices, provided the partitionings are such that the operations are defined. For example, if **B** is a partitioned 4×1 matrix such that

$$\mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ \mathbf{B}_2 \end{bmatrix},$$

then

$$\mathbf{AB} = \begin{bmatrix} a_{11}b_1 + \mathbf{A}_{12}\mathbf{B}_2 \\ \mathbf{A}_{21}b_1 + \mathbf{A}_{22}\mathbf{B}_2 \end{bmatrix}.$$

VECTORS

A special kind of matrix that plays an important role in matrix theory is the kind that has either a *single row* or a *single column.* Such matrices are often referred to as **vectors.** Thus,

$$\mathbf{x} = [x_1, x_2, \ldots, x_n]$$

is a **row vector,** and

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

is a **column vector.** (Vectors are denoted in this book by **boldface lowercase letters.**) These vectors also are sometimes called *n-vectors* to indicate that they have n elements. For example,

$$\mathbf{x} = [1, 4, -2, \tfrac{1}{3}, 7]$$

is a 5-vector.

A **null vector 0** is either a row vector or a column vector whose elements are *all* 0s, that is,

$$\mathbf{0} = [0, 0, \ldots, 0] \qquad \text{or} \qquad \mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

(Although the same symbol **0** is used for either kind of *null vector,* as well as for a *null matrix,* the context normally will identify which it is.)

One reason vectors play an important role in matrix theory is that any $m \times n$ matrix can be partitioned into either m row vectors or n column vectors, and important properties of the matrix can be analyzed in terms of these vectors. To amplify, consider a set of n-vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ of the same type (i.e., they are either all row vectors or all column vectors).

Definition: A set of vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ is said to be **linearly dependent** if there exist m numbers (denoted by $c_1, c_2, \ldots, c_m$), some of which are not zero, such that

$$c_1\mathbf{x}_1 + c_2\mathbf{x}_2 + \cdots + c_m\mathbf{x}_m = \mathbf{0}.$$

Otherwise, the set is said to be **linearly independent.**

To illustrate, if $m = 3$ and

$$\mathbf{x}_1 = [1, 1, 1], \qquad \mathbf{x}_2 = [0, 1, 1], \qquad \mathbf{x}_3 = [2, 5, 5],$$

then there exist three numbers, namely, $c_1 = 2$, $c_2 = 3$, and $c_3 = -1$, such that

$$\begin{aligned} 2\mathbf{x}_1 + 3\mathbf{x}_2 - \mathbf{x}_3 &= [2, 2, 2] + [0, 3, 3] - [2, 5, 5] \\ &= [0, 0, 0], \end{aligned}$$

so, $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ are linearly dependent. Note that showing they are linearly dependent required finding three particular numbers (c_1, c_2, c_3) that make $c_1\mathbf{x}_1 + c_2\mathbf{x}_2 + c_3\mathbf{x}_3 = \mathbf{0}$, which is not always easy. Also note that this equation implies that

$$\mathbf{x}_3 = 2\mathbf{x}_1 + 3\mathbf{x}_2.$$

Thus, $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ can be interpreted as being linearly dependent because one of them is a linear combination of the others. However, if $\mathbf{x}_3$ were changed to

$$\mathbf{x}_3 = [2, 5, 6]$$

instead, then $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ would be linearly independent because it is impossible to express one of these vectors (say, $\mathbf{x}_3$) as a linear combination of the other two.

Definition: The **rank** of a *set* of vectors is the largest number of *linearly independent vectors* that can be chosen from the set.

Continuing the preceding example, we see that the rank of the set of vectors $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ was 2 (any pair of the vectors is linearly independent), but it became 3 after $\mathbf{x}_3$ was changed.

Definition: A **basis** for a *set* of vectors is a collection of linearly independent vectors taken from the set such that every vector in the set is a linear combination of the vectors in the collection (i.e., every vector in the set equals the sum of certain multiples of the vectors in the collection).

To illustrate, any pair of the vectors (say, $\mathbf{x}_1$ and $\mathbf{x}_2$) constituted a basis for $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ in the preceding example before $\mathbf{x}_3$ was changed. After $\mathbf{x}_3$ is changed, the basis becomes all three vectors.

The following theorem relates the last two definitions.

Theorem A4.1: A collection of r linearly independent vectors chosen from a set of vectors is a basis for the set if and only if the set has rank r.

SOME PROPERTIES OF MATRICES

Given the preceding results regarding vectors, it is now possible to present certain important concepts regarding matrices.

Definition: The **row rank** of a matrix is the rank of its set of row vectors. The **column rank** of a matrix is the rank of its column vectors.

For example, if matrix **A** is

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 2 & 5 & 5 \end{bmatrix},$$

then the preceding example of linearly dependent vectors shows that the row rank of **A** is 2. The column rank of **A** is also 2. (The first two column vectors are linearly independent but the second column vector minus the third equals **0**.) Having the same column rank and row rank is no coincidence, as the following general theorem indicates.

Theorem A4.2: The row rank and column rank of a matrix are equal.

Thus, it is only necessary to speak of the rank of a matrix.

The final concept to be discussed is the **inverse of a matrix.** For any nonzero number k, there exists a reciprocal or inverse $k^{-1} = 1/k$ such that

$$kk^{-1} = 1 = k^{-1}k.$$

Is there an analogous concept that is valid in matrix theory? In other words, for a given matrix **A** other than the null matrix, does there exist a matrix $\mathbf{A}^{-1}$ such that

$$\mathbf{AA}^{-1} = \mathbf{I} = \mathbf{A}^{-1}\mathbf{A}?$$

If **A** is not a square matrix (i.e., if the number of rows and the number of columns of **A** differ), the answer is *never,* because these matrix products would necessarily have a different number of rows for the multiplication to be defined (so that the equality operation would not be defined). However, if **A** is square, then the answer is *under certain circumstances,* as described by the following definition and Theorem A4.3.

Definition: A matrix is **nonsingular** if its rank equals both the number of rows and the number of columns. Otherwise, it is **singular.**

Thus, only square matrices can be *nonsingular.* A useful way of testing for nonsingularity is provided by the fact that a square matrix is nonsingular if and only if *its determinant is nonzero.*

Theorem A4.3: (*a*) If **A** is nonsingular, there is a unique nonsingular matrix $\mathbf{A}^{-1}$, called the **inverse** of **A**, such that $\mathbf{AA}^{-1} = \mathbf{I} = \mathbf{A}^{-1}\mathbf{A}$.
(*b*) If **A** is nonsingular and **B** is a matrix for which either $\mathbf{AB} = \mathbf{I}$ or $\mathbf{BA} = \mathbf{I}$, then $\mathbf{B} = \mathbf{A}^{-1}$.
(*c*) Only nonsingular matrices have inverses.

To illustrate matrix inverses, consider the matrix

$$\mathbf{A} = \begin{bmatrix} 5 & -4 \\ 1 & -1 \end{bmatrix}.$$

Notice that **A** is nonsingular since its determinant, $5(-1) - 1(-4) = -1$, is nonzero. Therefore, **A** must have an inverse, which happens to be

$$\mathbf{A}^{-1} = \begin{bmatrix} 1 & -4 \\ 1 & -5 \end{bmatrix}.$$

Hence,

$$\mathbf{AA}^{-1} = \begin{bmatrix} 5 & -4 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 & -4 \\ 1 & -5 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

and

$$\mathbf{A}^{-1}\mathbf{A} = \begin{bmatrix} 1 & -4 \\ 1 & -5 \end{bmatrix}\begin{bmatrix} 5 & -4 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

APPENDIX 5

TABLES

TABLE A5.1 Areas under the normal curve from K_α to ∞

$$P\{\text{standard normal} > K_\alpha\} = \int_{K_\alpha}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2}\, dx = \alpha$$

K_α	.00	.01	.02	.03	.04	.05	.06	.07	.08	.09
0.0	.5000	.4960	.4920	.4880	.4840	.4801	.4761	.4721	.4681	.4641
0.1	.4602	.4562	.4522	.4483	.4443	.4404	.4364	.4325	.4286	.4247
0.2	.4207	.4168	.4129	.4090	.4052	.4013	.3974	.3936	.3897	.3859
0.3	.3821	.3783	.3745	.3707	.3669	.3632	.3594	.3557	.3520	.3483
0.4	.3446	.3409	.3372	.3336	.3300	.3264	.3228	.3192	.3156	.3121
0.5	.3085	.3050	.3015	.2981	.2946	.2912	.2877	.2843	.2810	.2776
0.6	.2743	.2709	.2676	.2643	.2611	.2578	.2546	.2514	.2483	.2451
0.7	.2420	.2389	.2358	.2327	.2296	.2266	.2236	.2206	.2177	.2148
0.8	.2119	.2090	.2061	.2033	.2005	.1977	.1949	.1922	.1894	.1867
0.9	.1841	.1814	.1788	.1762	.1736	.1711	.1685	.1660	.1635	.1611
1.0	.1587	.1562	.1539	.1515	.1492	.1469	.1446	.1423	.1401	.1379
1.1	.1357	.1335	.1314	.1292	.1271	.1251	.1230	.1210	.1190	.1170
1.2	.1151	.1131	.1112	.1093	.1075	.1056	.1038	.1020	.1003	.0985
1.3	.0968	.0951	.0934	.0918	.0901	.0885	.0869	.0853	.0838	.0823
1.4	.0808	.0793	.0778	.0764	.0749	.0735	.0721	.0708	.0694	.0681
1.5	.0668	.0655	.0643	.0630	.0618	.0606	.0594	.0582	.0571	.0559
1.6	.0548	.0537	.0526	.0516	.0505	.0495	.0485	.0475	.0465	.0455
1.7	.0446	.0436	.0427	.0418	.0409	.0401	.0392	.0384	.0375	.0367
1.8	.0359	.0351	.0344	.0336	.0329	.0322	.0314	.0307	.0301	.0294
1.9	.0287	.0281	.0274	.0268	.0262	.0256	.0250	.0244	.0239	.0233
2.0	.0228	.0222	.0217	.0212	.0207	.0202	.0197	.0192	.0188	.0183
2.1	.0179	.0174	.0170	.0166	.0162	.0158	.0154	.0150	.0146	.0143
2.2	.0139	.0136	.0132	.0129	.0125	.0122	.0119	.0116	.0113	.0110
2.3	.0107	.0104	.0102	.00990	.00964	.00939	.00914	.00889	.00866	.00842
2.4	.00820	.00798	.00776	.00755	.00734	.00714	.00695	.00676	.00657	.00639
2.5	.00621	.00604	.00587	.00570	.00554	.00539	.00523	.00508	.00494	.00480
2.6	.00466	.00453	.00440	.00427	.00415	.00402	.00391	.00379	.00368	.00357
2.7	.00347	.00336	.00326	.00317	.00307	.00298	.00289	.00280	.00272	.00264
2.8	.00256	.00248	.00240	.00233	.00226	.00219	.00212	.00205	.00199	.00193
2.9	.00187	.00181	.00175	.00169	.00164	.00159	.00154	.00149	.00144	.00139

K_α	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9
3	.00135	$.0^{3}968$	$.0^{3}687$	$.0^{3}483$	$.0^{3}337$	$.0^{3}233$	$.0^{3}159$	$.0^{3}108$	$.0^{4}723$	$.0^{4}481$
4	$.0^{4}317$	$.0^{4}207$	$.0^{4}133$	$.0^{5}854$	$.0^{5}541$	$.0^{5}340$	$.0^{5}211$	$.0^{5}130$	$.0^{6}793$	$.0^{6}479$
5	$.0^{6}287$	$.0^{6}170$	$.0^{7}996$	$.0^{7}579$	$.0^{7}333$	$.0^{7}190$	$.0^{7}107$	$.0^{8}599$	$.0^{8}332$	$.0^{8}182$
6	$.0^{9}987$	$.0^{9}530$	$.0^{9}282$	$.0^{9}149$	$.0^{10}777$	$.0^{10}402$	$.0^{10}206$	$.0^{10}104$	$.0^{11}523$	$.0^{11}260$

Source: F. E. Croxton, *Tables of Areas in Two Tails and in One Tail of the Normal Curve.* Copyright 1949 by Prentice-Hall, Inc., Englewood Cliffs, NJ.

TABLE A5.2 100 α percentage points of Student's t distribution

P\{Student's t with v Degrees of Freedom $\geq$ Tabled Value\} = α

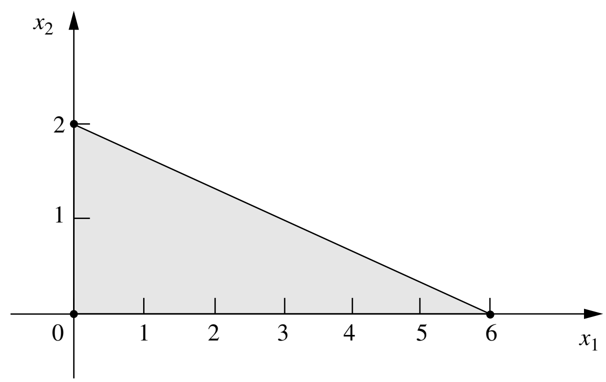
v \ α	0.40	0.25	0.10	0.05	0.025	0.01	0.005	0.0025	0.001	0.0005
1	0.325	1.000	3.078	6.314	12.706	31.821	63.657	127.32	318.31	636.62
2	0.289	0.816	1.886	2.920	4.303	6.965	9.925	14.089	22.327	31.598
3	0.277	0.765	1.638	2.353	3.182	4.541	5.841	7.453	10.214	12.924
4	0.271	0.741	1.533	2.132	2.776	3.747	4.604	5.598	7.173	8.610
5	0.267	0.727	1.476	2.015	2.571	3.365	4.032	4.773	5.893	6.869
6	0.265	0.718	1.440	1.943	2.447	3.143	3.707	4.317	5.208	5.959
7	0.263	0.711	1.415	1.895	2.365	2.998	3.499	4.029	4.785	5.408
8	0.262	0.706	1.397	1.860	2.306	2.896	3.355	3.833	4.501	5.041
9	0.261	0.703	1.383	1.833	2.262	2.821	3.250	3.690	4.297	4.781
10	0.260	0.700	1.372	1.812	2.228	2.764	3.169	3.581	4.144	4.587
11	0.260	0.697	1.363	1.796	2.201	2.718	3.106	3.497	4.025	4.437
12	0.259	0.695	1.356	1.782	2.179	2.681	3.055	3.428	3.930	4.318
13	0.259	0.694	1.350	1.771	2.160	2.650	3.012	3.372	3.852	4.221
14	0.258	0.692	1.345	1.761	2.145	2.624	2.977	3.326	3.787	4.140
15	0.258	0.691	1.341	1.753	2.131	2.602	2.947	3.286	3.733	4.073
16	0.258	0.690	1.337	1.746	2.120	2.583	2.921	3.252	3.686	4.015
17	0.257	0.689	1.333	1.740	2.110	2.567	2.898	3.222	3.646	3.965
18	0.257	0.688	1.330	1.734	2.101	2.552	2.878	3.197	3.610	3.922
19	0.257	0.688	1.328	1.729	2.093	2.539	2.861	3.174	3.579	3.883
20	0.257	0.687	1.325	1.725	2.086	2.528	2.845	3.153	3.552	3.850
21	0.257	0.686	1.323	1.721	2.080	2.518	2.831	3.135	3.527	3.819
22	0.256	0.686	1.321	1.717	2.074	2.508	2.819	3.119	3.505	3.792
23	0.256	0.685	1.319	1.714	2.069	2.500	2.807	3.104	3.485	3.767
24	0.256	0.685	1.318	1.711	2.064	2.492	2.797	3.091	3.467	3.745
25	0.256	0.684	1.316	1.708	2.060	2.485	2.787	3.078	3.450	3.725
26	0.256	0.684	1.315	1.706	2.056	2.479	2.779	3.067	3.435	3.707
27	0.256	0.684	1.314	1.703	2.052	2.473	2.771	3.057	3.421	3.690
28	0.256	0.683	1.313	1.701	2.048	2.467	2.763	3.047	3.408	3.674
29	0.256	0.683	1.311	1.699	2.045	2.462	2.756	3.038	3.396	3.659
30	0.256	0.683	1.310	1.697	2.042	2.457	2.750	3.030	3.385	3.646
40	0.255	0.681	1.303	1.684	2.021	2.423	2.704	2.971	3.307	3.551
60	0.254	0.679	1.296	1.671	2.000	2.390	2.660	2.915	3.232	3.460
120	0.254	0.677	1.289	1.658	1.980	2.358	2.617	2.860	3.160	3.373
∞	0.253	0.674	1.282	1.645	1.960	2.326	2.576	2.807	3.090	3.291

Source: Table 12 of *Biometrika Tables for Statisticians,* vol. I, 3d ed., 1966, by permission of the Biometrika Trustees.

PARTIAL ANSWERS TO SELECTED PROBLEMS

CHAPTER 3

3.1-1. **(a)**



3.1-4. $(x_1, x_2) = (13, 5)$; $Z = 31$.

3.1-11. **(b)** $(x_1, x_2, x_3) = (26.19, 54.76, 20)$; $Z = 2{,}904.76$.

3.2-3. **(b)** Maximize $\quad Z = 4{,}500x_1 + 4{,}500x_2,$

subject to

$$\begin{aligned} x_1 \qquad\qquad &\le 1 \\ x_2 &\le 1 \\ 5{,}000x_1 + 4{,}000x_2 &\le 6{,}000 \\ 400x_1 + 500x_2 &\le 600 \end{aligned}$$

and

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

3.4-1. **(a)** *Proportionality*: OK since it is implied that a fixed fraction of the radiation dosage at a given entry point is absorbed by a given area.

Additivity: OK since it is stated that the radiation absorption from multiple beams is additive.

Divisibility: OK since beam strength can be any fractional level.

Certainty: Due to the complicated analysis required to estimate the data on radiation absorption in different tissue types, there is considerable uncertainty about the data, so sensitivity analysis should be used.

3.4-11. (b) From Factory 1, ship 200 units to Customer 2 and 200 units to Customer 3. From Factory 2, ship 300 units to Customer 1 and 200 units to Customer 3.

3.4-13. (c) $Z = \$152{,}880$; $A_1 = 60{,}000$; $A_3 = 84{,}000$; $D_5 = 117{,}600$. All other decision variables are 0.

3.4-16. (b) Each optimal solution has $Z = \$13{,}330$.

3.6-1. (c, e)

	Resource Usage per Unit of Each Activity				
Resource	**Activity 1**	**Activity 2**	**Totals**		**Resource Available**
1	2	1	10	≤	10
2	3	3	20	≤	20
3	2	4	20	≤	20
Unit Profit	20	30	\$166.67		
Solution	3.333	3.333			

3.6-4. (a) Minimize $\quad Z = 84C + 72T + 60A,$

subject to

$$90C + 20T + 40A \geq 200$$
$$30C + 80T + 60A \geq 180$$
$$10C + 20T + 60A \geq 150$$

and

$$C \geq 0, \quad T \geq 0, \quad A \geq 0.$$

CHAPTER 4

4.1-1. (a) The corner-point solutions that are *feasible* are (0, 0), (0, 1), $(\frac{1}{4}, 1)$, $(\frac{2}{3}, \frac{2}{3})$, $(1, \frac{1}{4})$, and (1, 0).

4.3-4. $(x_1, x_2, x_3) = (0, 10, 6\frac{2}{3})$; $Z = 70$.

4.6-1. (a, c) $(x_1, x_2) = (2, 1)$; $Z = 7$.

4.6-4. (a, c, e) $(x_1, x_2, x_3) = (\frac{4}{5}, \frac{9}{5}, 0)$; $Z = 7$.

4.6-10. (a, b, d) $(x_1, x_2, x_3) = (0, 15, 15)$; $Z = 90$.
(c) For both the Big M method and the two-phase method, only the final tableau represents a feasible solution for the real problem.

4.6-15. (a, c) $(x_1, x_2) = (-\frac{8}{7}, \frac{18}{7})$; $Z = \frac{80}{7}$.

4.7-6. (a) $(x_1, x_2, x_3) = (0, 1, 3)$; $Z = 7$.
(b) $y_1^* = \frac{1}{2}$, $y_2^* = \frac{5}{2}$, $y_3^* = 0$. These are the marginal values of resources 1, 2, and 3, respectively.

CHAPTER 5

5.1-1. **(a)** $(x_1, x_2) = (2, 2)$ is optimal. Other CPF solutions are (0, 0), (3, 0), and (0, 3).

5.1-14. $(x_1, x_2, x_3) = (0, 15, 15)$ is optimal.

5.2-2. $(x_1, x_2, x_3, x_4, x_5) = (0, 5, 0, \frac{5}{2}, 0)$; Z = 50.

5.3-1. **(a)** Right side is $Z = 8$, $x_2 = 14$, $x_6 = 5$, $x_3 = 11$.
(b) $x_1 = 0$, $2x_1 - 2x_2 + 3x_3 = 5$, $x_1 + x_2 - x_3 = 3$.

CHAPTER 6

6.1-2. **(a)** Minimize $W = 15y_1 + 12y_2 + 45y_3$,

subject to

$$-y_1 + y_2 + 5y_3 \geq 10$$
$$2y_1 + y_2 + 3y_3 \geq 20$$

and

$$y_1 \geq 0, \quad y_2 \geq 0, \quad y_3 \geq 0.$$

6.3-1. **(c)**

Complementary Basic Solutions

Primal Problem			Dual Problem	
Basic Solution	Feasible?	$Z = W$	Feasible?	Basic Solution
(0, 0, 20, 10)	Yes	0	No	(0, 0, −6, −8)
(4, 0, 0, 6)	Yes	24	No	$\left(1\frac{1}{5}, 0, 0, -5\frac{3}{5}\right)$
(0, 5, 10, 0)	Yes	40	No	(0, 4, −2, 0)
$\left(2\frac{1}{2}, 3\frac{3}{4}, 0, 0\right)$	Yes and optimal	45	Yes and optimal	$\left(\frac{1}{2}, 3\frac{1}{2}, 0, 0\right)$
(10, 0, −30, 0)	No	60	Yes	(0, 6, 0, 4)
(0, 10, 0, −10)	No	80	Yes	(4, 0, 14, 0)

6.3-7. **(c)** Basic variables are x_1 and x_2. The other variables are nonbasic.
(e) $x_1 + 3x_2 + 2x_3 + 3x_4 + x_5 = 6$, $4x_1 + 6x_2 + 5x_3 + 7x_4 + x_5 = 15$, $x_3 = 0$, $x_4 = 0$, $x_5 = 0$. Optimal CPF solution is $(x_1, x_2, x_3, x_4, x_5) = (\frac{3}{2}, \frac{3}{2}, 0, 0, 0)$.

6.4-3. Maximize $W = 8y_1 + 6y_2$,

subject to

$$y_1 + 3y_2 \leq 2$$
$$4y_1 + 2y_2 \leq 3$$
$$2y_1 \leq 1$$

and

$$y_1 \geq 0, \quad y_2 \geq 0.$$

6.4-8. **(a)** Minimize $W = 120y_1 + 80y_2 + 100y_3$,

subject to

$$\begin{aligned} y_2 - 3y_3 &= -1 \\ 3y_1 - y_2 + y_3 &= 2 \\ y_1 - 4y_2 + 2y_3 &= 1 \end{aligned}$$

and

$$y_1 \geq 0, \quad y_2 \geq 0, \quad y_3 \geq 0.$$

6.6-1. **(d)** Not optimal, since $2y_1 + 3y_2 \geq 3$ is violated for $y_1^* = \frac{1}{5}$, $y_2^* = \frac{3}{5}$.
(f) Not optimal, since $3y_1 + 2y_2 \geq 2$ is violated for $y_1^* = \frac{1}{5}$, $y_2^* = \frac{3}{5}$.

6.7-1.

Part	New Basic Solution $(x_1, x_2, x_3, x_4, x_5)$	Feasible?	Optimal?
(*a*)	(0, 30, 0, 0, −30)	No	No
(*b*)	(0, 20, 0, 0, −10)	No	No
(*c*)	(0, 10, 0, 0, 60)	Yes	Yes
(*d*)	(0, 20, 0, 0, 10)	Yes	Yes
(*e*)	(0, 20, 0, 0, 10)	Yes	Yes
(*f*)	(0, 10, 0, 0, 40)	Yes	No
(*g*)	(0, 20, 0, 0, 10)	Yes	Yes
(*h*)	(0, 20, 0, 0, 10, $x_6 = -10$)	No	No
(*i*)	(0, 20, 0, 0, 0)	Yes	Yes

6.7-2. $-10 \leq \theta \leq \frac{10}{9}$

6.7-16. **(a)** $b_1 \geq 2$, $6 \leq b_2 \leq 18$, $12 \leq b_3 \leq 24$
(b) $0 \leq c_1 \leq \frac{15}{2}$, $c_2 \geq 2$

CHAPTER 7

7.1-2. $(x_1, x_2, x_3) = (\frac{2}{3}, 2, 0)$ with $Z = \frac{22}{3}$ is optimal.

7.1-6.

Part	New Optimal Solution	Value of Z
(*a*)	$(x_1, x_2, x_3, x_4, x_5) = (0, 0, 9, 3, 0)$	117
(*b*)	$(x_1, x_2, x_3, x_4, x_5) = (0, 5, 5, 0, 0)$	90

7.2-1. **(a, b)**

Range of θ	Optimal Solution	$Z(\theta)$
$0 \leq \theta \leq 2$	$(x_1, x_2) = (0, 5)$	$120 - 10\theta$
$2 \leq \theta \leq 8$	$(x_1, x_2) = \left(\frac{10}{3}, \frac{10}{3}\right)$	$\frac{320 - 10\theta}{3}$
$8 \leq \theta$	$(x_1, x_2) = (5, 0)$	$40 + 5\theta$

7.2-4.

Range of θ	Optimal Solution x_1	x_2	$Z(\theta)$
$0 \le \theta \le 1$	$10 + 2\theta$	$10 + 2\theta$	$30 + 6\theta$
$1 \le \theta \le 5$	$10 + 2\theta$	$15 - 3\theta$	$35 + \theta$
$5 \le \theta \le 25$	$25 - \theta$	0	$50 - 2\theta$

7.3-3. $(x_1, x_2, x_3) = (1, 3, 1)$ with $Z = 8$ is optimal.

7.5-6. $(x_1, x_2) = (15, 0)$ is optimal.

CHAPTER 8

8.1-2. (b)

		Destination			
		Today	Tomorrow	Dummy	Supply
Source	Dick	3.0	2.7	0	5
	Harry	2.9	2.8	0	4
	Demand	3	4	2	

8.2-2. (a) Basic variables: $x_{11} = 4$, $x_{12} = 0$, $x_{22} = 4$, $x_{23} = 2$, $x_{24} = 0$, $x_{34} = 5$, $x_{35} = 1$, $x_{45} = 0$; $Z = 53$.
(b) Basic variables: $x_{11} = 4$, $x_{23} = 2$, $x_{25} = 4$, $x_{31} = 0$, $x_{32} = 0$, $x_{34} = 5$, $x_{35} = 1$, $x_{42} = 4$; $Z = 45$.
(c) Basic variables: $x_{11} = 4$, $x_{23} = 2$, $x_{25} = 4$, $x_{32} = 0$, $x_{34} = 5$, $x_{35} = 1$, $x_{41} = 0$, $x_{42} = 4$; $Z = 45$.

8.2-8. (a) $x_{11} = 3$, $x_{12} = 2$, $x_{22} = 1$, $x_{23} = 1$, $x_{33} = 1$, $x_{34} = 2$; three iterations to reach optimality.
(b, c) $x_{11} = 3$, $x_{12} = 0$, $x_{13} = 0$, $x_{14} = 2$, $x_{23} = 2$, $x_{32} = 3$; already optimal.

8.2-11. $x_{11} = 10$, $x_{12} = 15$, $x_{22} = 0$, $x_{23} = 5$, $x_{25} = 30$, $x_{33} = 20$, $x_{34} = 10$, $x_{44} = 10$; cost = \$77.30. Also have other tied optimal solutions.

8.2-12. (b) Let x_{ij} be the shipment from plant i to distribution center j. Then $x_{13} = 2$, $x_{14} = 10$, $x_{22} = 9$, $x_{23} = 8$, $x_{31} = 10$, $x_{32} = 1$; cost = \$20,200.

8.3-4. (a)

		Task				
		Backstroke	Breaststroke	Butterfly	Freestyle	Dummy
Assignee	Carl	37.7	43.4	33.3	29.2	0
	Chris	32.9	33.1	28.5	26.4	0
	David	33.8	42.2	38.9	29.6	0
	Tony	37.0	34.7	30.4	28.5	0
	Ken	35.4	41.8	33.6	31.1	0

CHAPTER 9

9.3-3. (a) $O \rightarrow A \rightarrow B \rightarrow D \rightarrow T$ or $O \rightarrow A \rightarrow B \rightarrow E \rightarrow D \rightarrow T$, with length = 16.

9.4-1. (a) $\{(O, A); (A, B); (B, C); (B, E); (E, D); (D, T)\}$, with length = 18.

9.5-1. (a)

Arc	(1, 2)	(1, 3)	(1, 4)	(2, 5)	(3, 4)	(3, 5)	(3, 6)	(4, 6)	(5, 7)	(6, 7)
Flow	4	4	1	4	1	0	3	2	4	5

CHAPTER 10

10.2-2. Since activities D, E, J, and K are not immediate predecessors of any other activities, the corresponding nodes have arcs leading directly to the Finish node.

10.3-4. (b) Ken will be able to meet his deadline if no delays occur.
(c) Critical paths: Start $\rightarrow B \rightarrow E \rightarrow J \rightarrow M \rightarrow$ Finish
Start $\rightarrow C \rightarrow G \rightarrow L \rightarrow N \rightarrow$ Finish
Focus attention on activities with 0 slack.
(d) If activity I takes 2 extra weeks, there will be no delay because its slack is 3.

10.3-7. Critical path: Start $\rightarrow A \rightarrow B \rightarrow C \rightarrow E \rightarrow F \rightarrow J \rightarrow K \rightarrow N \rightarrow$ Finish
Total duration = 26 weeks

10.4-1. $\mu = 37$, $\sigma^2 = 9$

10.4-5. (a)

Activity	μ	σ^2
A	12	0
B	23	16
C	15	1
D	27	9
E	18	4
F	6	4

(b) Start $\rightarrow A \rightarrow C \rightarrow E \rightarrow F \rightarrow$ Finish Length = 51 days Mean critical path
Start $\rightarrow B \rightarrow D \rightarrow$ Finish Length = 50 days

(d) $\dfrac{d - \mu_p}{\sqrt{\sigma_p^2}} = \dfrac{57 - 50}{\sqrt{25}} = 1.4 \Rightarrow P(T \leq 57) = 0.9192$ (from the Normal table)

10.5-4. (a) Critical path: Start $\rightarrow A \rightarrow C \rightarrow E \rightarrow$ Finish
Total duration = 12 weeks

(b) New plan:

Activity	Duration	Cost
A	3 weeks	$54,000
B	3 weeks	$65,000
C	3 weeks	$68,666
D	2 weeks	$41,500
E	2 weeks	$80,000

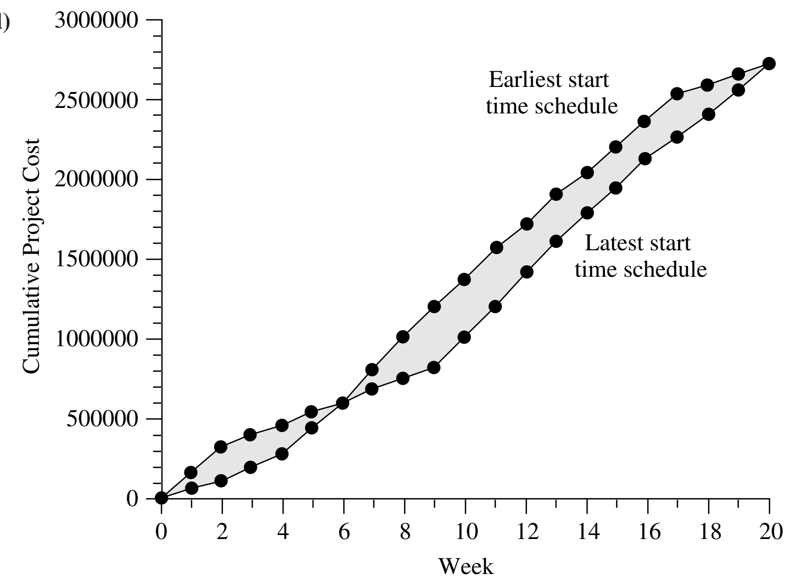
$7,834 is saved by this crashing schedule.

10.5-5. (b)

Activity	Time Normal	Time Crash	Cost Normal	Cost Crash	Maximum Time Reduction	Crash Cost per Week Saved	Start Time	Time Reduction	Finish Time
A	5	3	$20	$30	2	$ 5	0	2	3
B	3	2	$10	$20	1	$10	0	1	2
C	4	2	$16	$24	2	$ 4	3	0	7
D	6	3	$25	$43	3	$ 6	3	0	9
E	5	4	$22	$30	1	$ 8	2	0	7
F	7	4	$30	$48	3	$ 6	2	0	9
G	9	5	$25	$45	4	$ 5	7	1	15
H	8	6	$30	$44	2	$ 7	9	2	15

Finish Time = 15
Total Cost = $217

10.6-2. (d)



CHAPTER 11

11.3-1.

	Store 1	Store 2	Store 3
Allocations	1	2	2
	3	2	0

11.3-8. **(a)**

Phase	(*a*)	(*b*)
1	2*M*	2.945*M*
2	1*M*	1.055*M*
3	1*M*	0
Market share	6%	6.302%

11.3-14. $x_1 = -2 + \sqrt{13} \approx 1.6056$, $x_2 = 5 - \sqrt{13} \approx 1.3944$; $Z = 98.233$.

11.4-3. Produce 2 on first production run; if none acceptable, produce 2 on second run. Expected cost = $575.

CHAPTER 12

12.1-2. **(a)** Minimize $Z = 4.5x_{em} + 7.8x_{ec} + 3.6x_{ed} + 2.9x_{el} + 4.9x_{sm} + 7.2x_{sc} + 4.3x_{sd} + 3.1x_{sl}$,

subject to

$$\begin{aligned} x_{em} + x_{ec} + x_{ed} + x_{el} &= 2 \\ x_{sm} + x_{sc} + x_{sd} + x_{sl} &= 2 \\ x_{em} + x_{sm} &= 1 \\ x_{ec} + x_{sc} &= 1 \\ x_{ed} + x_{sd} &= 1 \\ x_{el} + x_{sl} &= 1 \end{aligned}$$

and

all x_{ij} are binary.

12.3-1. **(b)**

Constraint	Product 1	Product 2	Product 3	Product 4	Totals		Modified Right-Hand Side	Original Right-Hand Side
First	5	3	6	4	6000	≤	6000	6000
Second	4	6	3	5	12000	≤	105999	6000
Marginal revenue	$70	$60	$90	$80	$80000			
Solution	0	2000	0	0				
	≤	≤	≤	≤				
	0	9999	0	0				
Set Up?	0	1	0	0	1	≤	2	
Start-up Cost	$50,000	$40,000	$70,000	$60,000				

Contingency Constraints:

Product 3:	0	≤	1	:Product 1 or 2
Product 4:	0	≤	1	:Product 1 or 2

Which Constraint (0 = First, 1 = Second): 0

12.3-5. (b, d) (long, medium, short) = (14, 0, 16), with profit of \$95.6 million.

12.4-3. (b)

Constraint	Product 1	Product 2	Product 3	Total		Right-Hand Side
Milling	9	3	5	498	≤	500
Lathe	5	4	0	349	≤	350
Grinder	3	0	2	135	≤	150
Sales Potential	0	0	1	0	≤	20
Unit Profit	50	20	25	\$2870		
Solution	45	31	0			
	≤	≤	≤			
	999	999	0			
Produce?	1	1	0	2	≤	2

12.4-6. (a) Let $x_{ij} = \begin{cases} 1 & \text{if arc } i \rightarrow j \text{ is included in shortest path} \\ 0 & \text{otherwise.} \end{cases}$

Mutually exclusive alternatives: For each column of arcs, exactly one arc is included in the shortest path. Contingent decisions: The shortest path leaves node i only if it enters node i.

12.5-1. (a) $(x_1, x_2) = (2, 3)$ is optimal.
(b) None of the feasible rounded solutions are optimal for the integer programming problem.

12.6-1. $(x_1, x_2, x_3, x_4, x_5) = (0, 0, 1, 1, 1)$, with $Z = 6$.

12.6-7. (b)

Task	1	2	3	4	5
Assignee	1	3	2	4	5

12.6-9. $(x_1, x_2, x_3, x_4) = (0, 1, 1, 0)$, with $Z = 36$.

12.7-1. (a, b) $(x_1, x_2) = (2, 1)$ is optimal.

12.8-1. (a) $x_1 = 0, x_3 = 0$

CHAPTER 13

13.2-7. (a) Concave.

13.4-1. Approximate solution = 1.0125.

13.5-4. Exact solution is $(x_1, x_2) = (2, -2)$.

13.5-8. (a) Approximate solution is $(x_1, x_2) = (0.75, 1.5)$.

13.6-3.

$$\begin{aligned} -4x_1^3 - 4x_1 - 2x_2 + 2u_1 + u_2 &= 0 \quad (\text{or} \le 0 \text{ if } x_1 = 0). \\ -2x_1 - 8x_2 + u_1 + 2u_2 &= 0 \quad (\text{or} \le 0 \text{ if } x_2 = 0). \\ -2x_1 - x_2 + 10 &= 0 \quad (\text{or} \le 0 \text{ if } u_1 = 0). \\ -x_1 - 2x_2 + 10 &= 0 \quad (\text{or} \le 0 \text{ if } u_2 = 0). \end{aligned}$$
$$x_1 \ge 0, \quad x_2 \ge 0, \quad u_1 \ge 0, \quad u_2 \ge 0.$$

13.6-8. $(x_1, x_2) = (1, 2)$ cannot be optimal.

13.6-10. **(a)** $(x_1, x_2) = (1 - 3^{-1/2}, 3^{-1/2})$.

13.7-2. **(a)** $(x_1, x_2) = (2, 0)$ is optimal.
(b) Minimize $\quad Z = z_1 + z_2,$

subject to

$$\begin{aligned} 2x_1 \qquad\quad + u_1 - y_1 \qquad\qquad + z_1 \qquad &= 8 \\ 2x_2 + u_1 \qquad - y_2 \qquad\qquad + z_2 &= 4 \\ x_1 + x_2 \qquad\qquad\qquad + v_1 \qquad\qquad &= 2 \end{aligned}$$

$x_1 \geq 0, \quad x_2 \geq 0, \quad u_1 \geq 0, \quad y_1 \geq 0, \quad y_2 \geq 0, \quad v_1 \geq 0, \quad z_1 \geq 0,$
$z_2 \geq 0.$

13.8-3. **(b)** Maximize $\quad Z = 3x_{11} - 3x_{12} - 15x_{13} + 4x_{21} - 4x_{23},$

subject to

$$\begin{aligned} x_{11} + x_{12} + x_{13} + 3x_{21} + 3x_{22} + 3x_{23} &\leq 8 \\ 5x_{11} + 5x_{12} + 5x_{13} + 2x_{21} + 2x_{22} + 2x_{23} &\leq 14 \end{aligned}$$

and

$$0 \leq x_{ij} \leq 1, \qquad \text{for } i = 1, 2, 3;\ j = 1, 2, 3.$$

13.9-1. $(x_1, x_2) = (5, 0)$ is optimal.

13.9-10. **(a)** $(x_1, x_2) = \left(\frac{1}{3}, \frac{2}{3}\right)$.

13.10-5. **(a)** $P(x; r) = -2x_1 - (x_2 - 3)^2 - r\left(\frac{1}{x_1 - 3} + \frac{1}{x_2 - 3}\right)$.

(b) $(x_1, x_2) = \left[3 + \left(\frac{r}{2}\right)^{1/2}, 3 + \left(\frac{r}{2}\right)^{1/3}\right]$.

CHAPTER 14

14.2-2. **(a)** Player 1: strategy 2; player 2: strategy 1.

14.2-7. **(a)** Politician 1: issue 2; politician 2: issue 2.
(b) Politician 1: issue 1; politician 2: issue 2.

14.4-3. **(a)** $(x_1, x_2) = (\frac{2}{5}, \frac{3}{5})$; $(y_1, y_2, y_3) = (\frac{1}{5}, 0, \frac{4}{5})$; $v = \frac{8}{5}$.

14.5-2. **(a)** Maximize $\quad x_4,$

subject to

$$\begin{aligned} 5x_1 + 2x_2 + 3x_3 - x_4 &\geq 0 \\ 4x_2 + 2x_3 - x_4 &\geq 0 \\ 3x_1 + 3x_2 \qquad - x_4 &\geq 0 \\ x_1 + 2x_2 + 4x_3 - x_4 &\geq 0 \\ x_1 + x_2 + x_3 \qquad &= 1 \end{aligned}$$

and

$$x_1 \geq 0, \qquad x_2 \geq 0, \qquad x_3 \geq 0, \qquad x_4 \geq 0.$$

CHAPTER 15

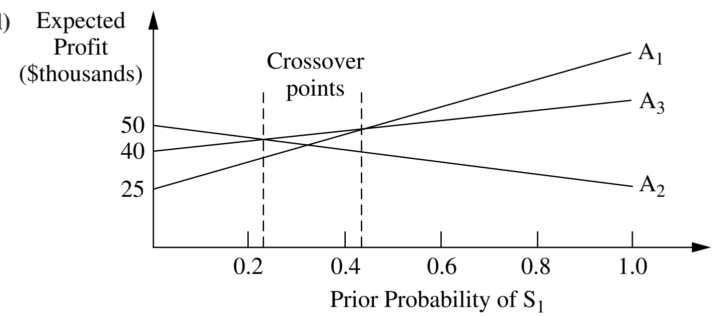
15.2-1. (a)

Alternative	State of Nature: Sell 10,000	Sell 100,000
Build Computers	0	54
Sell Rights	15	15

(c) Let p = prior probability of selling 10,000. They should build when $p \leq 0.722$, and sell when $p > 0.722$.

15.2-3. (c) Warren should make the countercyclical investment.

15.2-5. (d)



A_2 and A_3 cross at approximately $p = 0.25$. A_1 and A_3 cross at approximately $p = 0.43$.

15.2-8. Order 25.

15.3-1. (a) EVPI = EP (with perfect info) − EP (without more info) = 34.5 − 27 = $7.5 million.

(d)

Data: State of Nature	Prior Probability	P (Finding | State) Finding: Sell 10,000	Sell 100,000
Sell 10,000	0.5	0.666666667	0.333333333
Sell 100,000	0.5	0.333333333	0.666666667

Posterior Probabilities: Finding	P (Finding)	P (State | Finding) State of Nature: Sell 10,000	Sell 100,000
Sell 10,000	0.5	0.666666667	0.333333333
Sell 100,000	0.5	0.333333333	0.666666667

15.3-3. (b) EVPI = EP (with perfect info) − EP (without more info) = 53 − 35 = $18
(c) Betsy should consider spending up to $18 to obtain more information.

15.3-8. (a) Up to $230,000
(b) Order 25.

15.3-9. (a)

	State of Nature		
Alternative	**Poor Risk**	**Average Risk**	**Good Risk**
Extend Credit	−15,000	10,000	20,000
Don't Extend Credit	0	0	0
Prior Probabilities	0.2	0.5	0.3

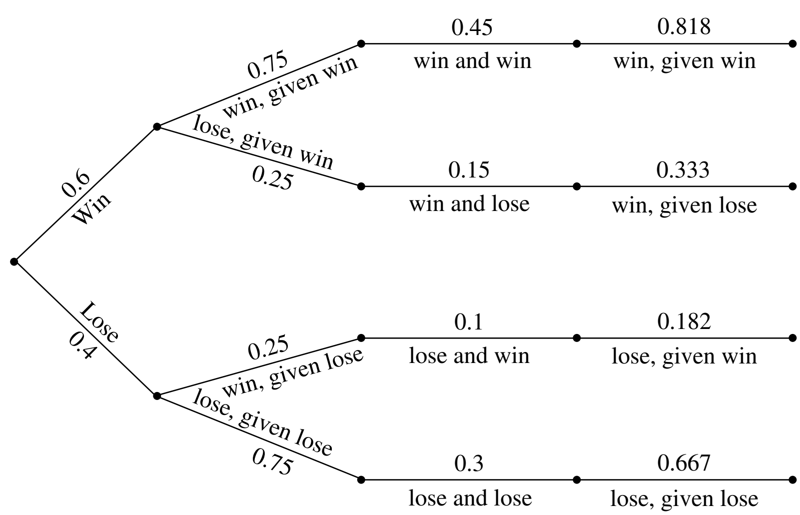
(c) EVPI = EP (with perfect info) − EP (without more info) = 11,000 − 8,000 = \$3,000. This indicates that the credit-rating organization should not be used.

15.3-13. (a) Guess coin 1.
(b) Heads: coin 2; tails: coin 1.

15.4-1. (b) The optimal policy is to do no market research and build the computers.

15.4-4. (c) EVPI = EP (with perfect info) − EP (without more info) = 1.8 − 1 = \$800,000
(d)

Prior Probabilities	Conditional Probabilities	Joint Probabilities	Posterior Probabilities
P (state)	*P* (finding|state)	*P* (state and finding)	*P* (state|finding)



(f, g) Leland University should hire William. If he predicts a winning season then they should hold the campaign. If he predicts a losing season then they should not hold the campaign.

15.4-10. (a) Choose to introduce the new product (expected payoff is \$12.5 million).
(b) EVPI = EP (with perfect info) − EP (without more info) = 20 − 12.5 = \$7.5 million
(c) The optimal policy is not to test but to introduce the new product.

15.5-2. (a) Choose not to buy insurance (expected payoff is \$249,840).
(b) u(insurance) = 499.82
u(no insurance) = 499.8
Optimal policy is to buy insurance.

15.5-4. $u(10) = 9$

CHAPTER 16

16.3-3. **(c)** $\pi_0 = \pi_1 = \pi_2 = \pi_3 = \pi_4 = \frac{1}{5}$.

16.4-1. **(a)** All states belong to the same recurrent class.

16.5-8. **(a)** $\pi_0 = 0.182$, $\pi_1 = 0.285$, $\pi_2 = 0.368$, $\pi_3 = 0.165$.
(b) 6.50

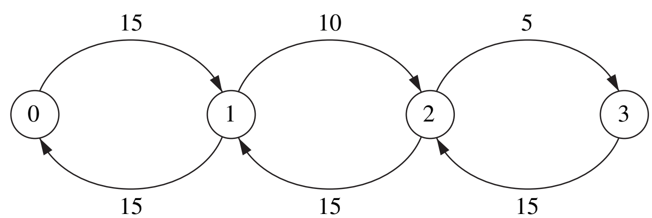
CHAPTER 17

17.2-1. Input source: population having hair; customers: customers needing haircuts; and so forth for the queue, queue discipline, and service mechanism.

17.2-2. **(b)** $L_q = 0.375$
(d) $W - W_q = 24.375$ minutes

17.4-2. **(c)** 0.0527

17.5-5. **(a)** State:



(c) $P_0 = \frac{9}{26}$, $P_1 = \frac{9}{26}$, $P_2 = \frac{3}{13}$, $P_3 = \frac{1}{13}$.
(d) $W = 0.11$ hour.

17.5-9. **(b)** $P_0 = \frac{2}{5}$, $P_n = (\frac{3}{5})(\frac{1}{2})^n$
(c) $L = \frac{6}{5}$, $L_q = \frac{3}{5}$, $W = \frac{1}{25}$, $W_q = \frac{1}{50}$

17.6-1. **(a)** $P_0 + P_1 + P_2 + P_3 + P_4 = 0.96875$ or 97 percent of the time.

17.6-21. **(a)** Combined expected waiting time = 0.211
(c) An expected process time of 3.43 minutes would cause the expected waiting times to be the same for the two procedures.

17.6-29. **(a)** 0.429

17.6-33. **(a)** three machines
(b) three operators

17.7-1. **(a)** W_q (exponential) $= 2W_q$ (constant) $= \frac{8}{5}W_q$ (Erlang).
(b) W_q (new) $= \frac{1}{2}W_q$ (old) and L_q (new) $= L_q$ (old) for all distributions.

17.7-6. **(a, b)** Under the current policy an airplane loses 1 day of flying time as opposed to 3.25 days under the proposed policy.
Under the current policy 1 airplane is losing flying time per day as opposed to 0.8125 airplane.

17.7-10.

Service Distribution	P_0	P_1	P_2	L
Erlang	0.561	0.316	0.123	0.561
Exponential	0.571	0.286	0.143	0.571

17.8-1. (a) This system is an example of a nonpreemptive priority queueing system.

(c) $\dfrac{W_q \text{ for first-class passengers}}{W_q \text{ for coach-class passengers}} = \dfrac{0.033}{0.083} = 0.4$

17.8-4. ***(a)*** $W = \frac{1}{2}$
(b) $W_1 = 0.20,\ W_2 = 0.35,\ W_3 = 1.10$
(c) $W_1 = 0.125,\ W_2 = 0.3125,\ W_3 = 1.250$

CHAPTER 18

18.3-1. (a) $E(WC) = 16$
(b) $E(WC) = 26.5$

18.4-2. 4 cash registers

18.4-5. (a) Model 2 with s fixed at 1
(b) Adopt the proposal.

18.4-10. (d) $E(TC)$ for status quo = \$85 per hour
$E(TC)$ for proposal = \$83 per hour

18.4-13. (a) The customers are trucks to be loaded or unloaded and the servers are crews. The system currently has 1 server.
(e) A one-person team should not be considered since that would lead to a utilization factor of $\rho = 1$, which is not permitted in this model.
(f, g) $E(TC)$ for 4 members = \$82.50 per hour
$E(TC)$ for 3 members = \$65 per hour
$E(TC)$ for 2 members = \$55 per hour
A crew of 2 people will minimize the expected total cost per hour.

18.4-25. One doctor: $E(TC)$ = \$624.80, two doctors: \$92.50; have two doctors.

CHAPTER 19

19.3-1. (a) $t = 1.83,\ Q = 54.77$
(b) $t = 1.91,\ Q = 57.45,\ S = 52.22$

19.3-3. (a)

Data

D =	676	(demand/year)
K =	\$75	(setup cost)
h =	\$600.00	(unit holding cost)
L =	3.5	(lead time in days)
WD =	365	(working days/year)

Decision

Q =	5	(order quantity)

Results

Reorder point =	6.5
Annual setup cost =	\$10,140
Annual holding cost =	\$ 1,500
Total variable cost =	\$11,640

(d) Data

D =	676	(demand/year)
K =	\$75	(setup cost)
h =	\$600	(unit holding cost)
L =	3.5	(lead time in days)
WD =	365	(working days/year)

Results

Reorder point =	6.48
Annual setup cost =	\$3,900
Annual holding cost =	\$3,900
Total variable cost =	\$7,800

Decision

Q =	13	(order quantity)

The results are the same as those obtained in part (*c*).

(f) Number of orders per year = 52
ROP = 6.5 − inventory level when each order is placed

(g) The optimal policy reduces the total variable inventory cost by \$3,840 per year, which is a 33 percent reduction.

19.3-7. (a) h = \$3 per month which is 15 percent of the acquisition cost.

(c) Reorder point is 10.

(d) ROP = 5 hammers, which adds \$20 to his TVC (5 hammers × \$4 holding cost).

19.3-9. $t = 3.26$, $Q = 26{,}046$, $S = 24{,}572$

19.3-15. (a) Optimal $Q = 500$

19.4-4. Produce 3 units in period 1 and 4 units in period 3.

19.5-6. (b) Ground Chuck: $R = 145$.
Chuck Wagon: $R = 829$.

(c) Ground Chuck: safety stock = 45.
Chuck Wagon: safety stock = 329.

(f) Ground Chuck: \$39,378.71.
Chuck Wagon: \$41,958.61.
Jed should choose Ground Chuck as their supplier.

(g) If Jed would like to use the beef within a month of receiving it, then Ground Chuck is the better choice. The order quantity with Ground Chuck is roughly 1 month's supply, whereas with Chuck Wagon the optimal order quantity is roughly 3 month's supply.

19.6-4. (a) Optimal service level = 0.667

(c) $Q^* = 500$

(d) The probability of running short is 0.333.

(e) Optimal service level = 0.833

19.6-8. (a) This problem can be interpreted as an inventory problem with uncertain demand for a perishable product with euro-traveler's checks as the product. Once Stan gets back from his trip the checks are not good anymore, so they are a perishable product. He can re-deposit the amount into his savings account but will incur a fee of lost interest. Stan must decide how many checks to buy without knowing how many he will need.

C_{under} = value of 1 day − cost of 1 day − cost of 1 check = \$49.
C_{over} = cost of check + lost interest = \$3

(b) Purchase 4 additional checks.

(c) Optimal service level = 0.94
Buy 4 additional checks.

19.7-3. If $x \leq 46$, order $46 - x$ units; otherwise, do not order.

19.7-10. **(a)** $G(y) = \frac{3}{10}y + 70e^{-y/25} - \frac{15}{2}$
(b) $(k, Q) = (21, 100)$ policy

CHAPTER 20

20.4-1. **(c)** Forecast = 36

20.4-3. Forecast = 2,091

20.4-7. Forecast (0.1) = 2,072

20.6-2. Forecast = 552

20.6-4. Forecast for next production yield = 62 percent

20.7-1. **(a)** MAD = 15

20.7-4. **(a)** Since sales are relatively stable, the averaging method would be appropriate for forecasting future sales. This method uses a larger sample size than the last-value method, which should make it more accurate. Since the older data are still relevant, they should not be excluded, as would be the case in the moving-average method.
(e) Considering the MAD values, the averaging method is the best one to use.
(f) Unless there is reason to believe that sales will not continue to be relatively stable, the averaging method is likely to be the most accurate in the future as well. However, 12 data points generally are inadequate for drawing definitive conclusions.

20.9-1. **(b)** $y = 410 + 17.6x$
(d) $y = 604$

CHAPTER 21

21.2-1. **(c)** Use slow service when no customers or one customer is present and fast service when two customers are present.

21.2-2. **(a)** The possible states of the car are dented and not dented.
(c) When the car is not dented, park it on the street in one space. When the car is dented, get it repaired.

21.2-5. **(c)** State 0: attempt ace; state 1: attempt lob.

21.3-2. **(a)** Minimize $Z = 4.5y_{02} + 5y_{03} + 50y_{14} + 9y_{15}$,

subject to

$$y_{01} + y_{02} + y_{03} + y_{14} + y_{15} = 1$$
$$y_{01} + y_{02} + y_{03} - \left(\frac{9}{10}y_{01} + \frac{49}{50}y_{02} + y_{03} + y_{14}\right) = 0$$
$$y_{14} + y_{15} - \left(\frac{1}{10}y_{01} + \frac{1}{50}y_{02} + y_{15}\right) = 0$$

and

all $y_{ik} \geq 0$.

21.3-5. **(a)** Minimize $Z = -\frac{1}{8}y_{01} + \frac{7}{24}y_{02} + \frac{1}{2}y_{11} + \frac{5}{12}y_{12}$,

subject to

$$y_{01} + y_{02} - \left(\frac{3}{8}y_{01} + y_{11} + \frac{7}{8}y_{02} + y_{12}\right) = 0$$
$$y_{11} + y_{12} - \left(\frac{5}{8}y_{01} + \frac{1}{8}y_{02}\right) = 0$$
$$y_{01} + y_{02} + y_{11} + y_{12} = 1$$

and

$y_{ik} \geq 0$ for $i = 0, 1$; $k = 1, 2$.

21.4-2. Car not dented: park it on the street in one space. Car dented: repair it.

21.4-5. State 0: attempt ace. State 1: attempt lob.

21.5-1. Reject $600 offer, accept any of the other two.

21.5-2. **(a)** Minimize $Z = 60(y_{01} + y_{11} + y_{21}) - 600y_{02} - 800y_{12} - 1{,}000y_{22}$,

subject to

$$y_{01} + y_{02} - (0.95)\left(\frac{5}{8}\right)(y_{01} + y_{11} + y_{21}) = \frac{5}{8}$$
$$y_{11} + y_{12} - (0.95)\left(\frac{1}{4}\right)(y_{01} + y_{11} + y_{21}) = \frac{1}{4}$$
$$y_{21} + y_{22} - (0.95)\left(\frac{1}{8}\right)(y_{01} + y_{11} + y_{21}) = \frac{1}{8}$$

and

$y_{ik} \geq 0$ for $i = 0, 1, 2$; $k = 1, 2$.

21.5-3. After three iterations, approximation is, in fact, the optimal policy given for Prob. 21.5-1.

21.5-11. In periods 1 to 3: Do nothing when the machine is in state 0 or 1; overhaul when machine is in state 2; and replace when machine is in state 3. In period 4: Do nothing when machine is in state 0, 1, or 2; replace when machine is in state 3.

CHAPTER 22

22.1-1. **(b)** Let the numbers 0.0000 to 0.5999 correspond to strikes and the numbers 0.6000 to 0.9999 correspond to balls. The random observations for pitches are 0.7520 = ball, 0.4184 = strike, 0.4189 = strike, 0.5982 = strike, 0.9559 = ball, and 0.1403 = strike.

22.1-10. **(b)** Use $\lambda = 4$ and $\mu = 5$.

(i) Answers will vary. The option of training the two current mechanics significantly decreases the waiting time for German cars, without a significant impact on the wait for Japanese cars, and does so without the added cost of a third mechanic. Adding a third mechanic lowers the average wait for German cars even more, but comes at an added cost for the third mechanic.

22.3-1. **(a)** 5, 8, 1, 4, 7, 0, 3, 6, 9, 2

22.4-2. **(b)** $F(x) = 0.0965$ when $x = -5.18$
$F(x) = 0.5692$ when $x = 18.46$
$F(x) = 0.6658$ when $x = 23.29$

22.4-5. **(a)** $x = \sqrt{r}$

22.4-8. **(a)** Here is a sample replication.

Summary of Results:

Win? (1 = Yes, 0 = No)	0
Number of Tosses =	3

Simulated Tosses / **Results**

Toss	Die 1	Die 2	Sum	Win?	Lose?	Continue?
1	4	2	6	0	0	Yes
2	3	2	5	0	0	Yes
3	6	1	7	0	1	No
4	5	2	7	NA	NA	No
5	4	4	8	NA	NA	No
6	1	4	5	NA	NA	No
7	2	6	8	NA	NA	No

22.4-13. **(a)** $x = -4 \ln (1 - r)$
(b) $x = -2 \ln [(1 - r_1)(1 - r_2)]$
(c) $x = 4\sum_{i=1}^{6} r_i - 8$

22.7-1. Use the first 10 three-digit decimals from Table 22.3 and generate observations from

$$x_i = \frac{1}{1 - r_i}.$$

Method:	Analytic	Monte Carlo	Stratified sampling	Complementary random numbers
Mean:	∞	4.3969	8.7661	3.812

22.7-4. **(a)** Let the numbers 0.0000 to 0.3999 correspond to a minor repair and 0.4000 to 0.9999 correspond to a major repair. The average repair time is then (1.224 + 0.950 + 1.610)/3 = 1.26 hours.
(c) The average repair time is 1.28 hours.
(e) The average repair time is 1.09 hours.
(f) The method of complementary random numbers in part (*e*) gave the closest estimate. It performs well because using complements helps counteract the more extreme random numbers (such as 0.9503).

22.8-1. **(a)** Est. $\{W_q\} = \frac{7}{3}$ and $P\{1.572 \le W_q \le 3.094\} = 0.90$

AUTHOR INDEX

Abrahamson, P. G., 45n
Ahrens, J. H., 1108n
Ahuja, R. K., 449
Allen, A. O., 926
Anderson, D. R., 89
Andradóttir, S., 1140
Armstrong, J. E., 1038
Arrow, Kenneth J., 980n
Aumann, R. J., 743
Avis, D. 873n
Avriel, M., 705n
Axsätter, S., 958n, 987

Bachem, A., 702n
Badiru, A. B., 511n, 513
Baker, Robert, 347–348
Ball, M., 449
Banks, J., 1140
Barnes, E. R., 321n
Baumol, W. J., 654n
Bayes, Thomas, 754n, 759
Bazaraa, M. S., 171, 220, 284, 391, 449, 706
Beasley, J. E., 631
Beightler, C., 668n
Bellman, R., 980n
Ben-Tuvia, S., 1130n
Bertsekas, D. P., 568, 706, 1077
Billington, C., 985n
Binmore, K., 743
Birge, J. R., 707
Bixby, H., 631
Bland, R., 129n
Bolling, W. Blake, 835n
Box, G. E. P., 1026, 1038
Boykin, R. F., 12n
Braklow, J. W., 19n
Brenner, D. A., 20n
Brodie, R. J., 1038
Browne, J., 926
Bryn, J. E., 63n, 64n
Bunday, B. D., 866n
Bunn, D., 1038
Burke, P. J., 886n
Bushnell, R. C., 654n

Calvert, J. E., 171
Carson, J. S., II, 1140
Cavalier, T. M., 321n
Charnes, A., 669
Clemen, R. T., 781
Cochran, J. K., 1140
Cohen, M., 18n, 984n
Conway, R. W., 867n, 870n
Cooper, R. B., 890
Cooper, W. W., 669
Cottle, R. W., 670n
Covaliu, Z., 779n
Coveyou, R. R., 1105n
Crane, M. A., 1140
Crowder, Harlan E., 623, 631

Dakin, R. J., 616
Dantzig, George B., 2, 109, 172, 220, 284,
391, 449, 670n
DeGans, O. B., 22
Denardo, E. V., 568, 1077
DeWitt, C. W., 20n
Dieter, V., 1108n
Diewert, W. E., 705n
Dikin, I. I., 321n
Doig, A. G., 616
Dreger, J. B., 514
Duffin, R. J., 668n
Dutta, B., 743
Edwards, W., 781
Ehrenfeld, S., 1130n
Eilon, Samuel, 14
Erlang, A. K., 841, 863
Erlenkotter, D., 953n
Eschenbach, T. G., 779n
Esogbue, A. O., 987

Falomir, D., 62n
Feinberg, E. A., 1078
Fenelon, M., 631
Fishburn, P. C., 781, 782
Fishman, G. S., 1140
Fortuin, L., 22
Fossett, L., 873n
Fourer, R., 694
Fracco, A. V., 707
Frank, M., 698n
Freedman, B. A., 321n
Fridenberg, D., 743

Ganeshan, R., 987
Gardner, E. S., Jr., 1038
Gass, S. I., 22, 89
Geoffrion, A. M., 391, 683n, 694
Gill, P. E., 707
Glicksberg, I., 980n
Glover, F., 449
Glynn, P. W., 1132n
Goeller, B. F., 13n
Gomory, Ralph, 630
Graham, W. W., 19n
Greenberg, H. J., 166
Gross, D., 890
Gross, O., 980n
Grotschel, M., 702n
Gu, Z., 631

Hall, R. W., 22, 926
Hammesfahr, R. D. J., 1140
Harrell, C., 1140
Harris, C. M., 890
Hart, S., 743
Hassler, S. M., 19n
Hazelwood, R. N., 866n
Heavey, C., 926
Heyman, D. P., 827, 963n
Hillier, Frederick S., 89, 284, 391, 449, 631, 782, 870n, 873n, 926, 987, 1038, 1140
Hillier, M. S., 89, 284, 391, 449, 631, 782, 926, 926n, 987, 1038, 1140
Hirdijk, A., 872n
Hoff, J. C., 1038
Hoffman, Karla, 623, 631
Holloran, T. J., 63n, 64n
Hooker, J. N., 340
Horst, R., 707
Howard, R. A., 568, 1066n
Huxley, S. J., 9n

Iglehart, D. L., 1132n

Jackson, J. R., 887n
Jarvis, J. J., 171, 220, 284, 391, 449
Jenkins, G. M., 1026, 1038
Johnson, Ellis L., 623, 631

Kamburowski, J., 510n
Kamesam, P. V., 18n, 984n
Kao, E. P. C., 827
Kaplan, E. H., 10n
Karamarkar, Narendra, 163–165, 325, 340–322
Karish, W., 679n
Karlin, S., 828, 849n, 980n
Keefer, D. L., 510n
Kelton, W. D., 1114n, 1140
Kennington, J. L., 386n
Kirkwood, C. W., 778n
Kleiknen, J. P. C., 1140
Kleindorfer, P., 18n, 984n
Kleinrock, L., 890, 926
Klingman, D., 10n, 449
Korte, B., 702n
Kostreva, Michael, 623, 631
Kuhn, W., 679n

Land, A. H., 616
Lasdon, L. S., 20n, 62n
Lau, A. H.-L., 510n, 963n
Lau, H.-S., 510n, 963n
Law, A. M., 1114n, 1140
Lee, A. M., 926
Lee, H. C., 18n, 984n, 985n
Lemoine, A. J., 1140
Lenstra, J. K., 22
Lieberman, Gerald J., 89, 284, 391, 449, 631, 782, 926, 987, 1038, 1140
Little, John D. C., 840
Liu, B., 987
Lo, F., 873n
Lustig, I. J., 340

Mackulak, G. T., 1140
Magazine, M., 987
Magnanti, T. L., 449
Mangasarian, O. T., 683n
Manrodt, K. B., 1037n, 1038
Markowitz, Harry, 658
Marshall, K. T., 779, 782, 1038
Marsten, R. E., 340
Maxwell, W. L., 867n, 870n
McCormick, G. P., 707
McGregor, J., 849n
Meketon, M. S., 321n
Melhem, S. A., 20n
Meyer, R. R., 697n
Meyerson, R. B., 743
Middleton, Michael, 769, 1115
Miser, H. J., 22
Monma, C., 449
Morris, W. T., 22
Murray, W., 707
Murthy, D. N. P., 22, 391
Murty, K. G., 449, 707

Nahmias, S., 987
Nash, S. G., 707
Nelson, B. L., 1140
Nemhauser, G. L., 449, 631, 707
Nemirovskii, A., 707
Nesterov, Y., 707
Newell, G. F., 926
Nilikant, V., 1038
Nocedal, J., 707

O'Keefe, E., 10n
Oliver, R. M., 779n, 782
Oliver, R. T., 1038
Orlin, J. B., 449
Osman, I. H., 631
Owen, G., 743
Özelamar, L., 512n

Padberg, Manfred, 623, 631
Page, N. W., 22, 391
Pang, J. S., 670n
Papadopoulos, H. T., 926
Pardalos, P. M., 707
Parthasarathy, T., 743
Peck, K. E., 19n
Peck, L. G., 866n
Pegden, C. D., 1140
Peterson, E. L., 668n
Peterson, R., 987
Phillips, D. T., 668n
Phillips, N. V., 10n, 449
Pooch, U. W., 1140
Porteus, E. L., 963n
Powell, M. J. D., 702n
Powell, W. B., 19n
Prabhu, N. U., 873n, 890
Pritsker, A. A. B., 1140
Pulat, P. S., 511n, 513
Puterman, M. L., 1078
Pyke, D., 987

Rayward-Smith, V. J., 631
Reeves, C. R., 631
Reiman, M., 873
Reinfeld, N. V., 372n
Resnick, S. I., 827
Rinnooy Kan, A. H. G., 707
Rodin, E. Y., 22, 391
Ross, S., 827
Rothberg, E., 631
Roy, A., 62n
Russell, E. J., 374n
Rust, R. T., 1038

Sadowski, R. P., 1140
Saigal, R., 340
Saltzman, M., 340
Sanders, N. R., 1037n, 1038
Savoy, P. A., 1140
Scarf, H., 980n
Schaible, S., 669, 705n
Schedler, G. S., 1132n
Schnoy, P. P., 779n
Schriber, T. J., 1140
Schriver, A., 220, 631
Schwartz, A., 1078
Scraton, R. E., 866n
Seelen, L. P., 879n
Sen, A., 743
Shanno, D. F., 340
Shannon, R. E., 1140
Sharpe, William, 658
Sherali, H. D., 171, 220, 284, 391, 449, 706
Sherbrooke, C. C., 987
Shetty, C. M., 706
Shubik, M., 743
Sigal, C. E., 1140
Silver, E., 987

Simester, D., 1038
Simon, Herbert A., 14, 22
Smith, D. K., 568
Smith, G. D., 631
Sniedovich, M., 568
Sobel, M. J., 827, 963n
Sonderman, D., 45n
Soyster, A. L., 321n
Steiger, D., 10n
Stewart, W. J., 828
Stidham, S., Jr., 840n, 920n
Stone, R. E., 670n
Subramanian, R., 340
Suhl, Uwe H., 623, 631
Sweeney, D. J., 89

Tavares, L. V., 514
Taylor, H., 828
Taylor, P. E., 9n
Tayur, S., 987
Tekerian, A., 18n, 984n
Terlaky, T., 340
Thapa, M. N., 172, 220, 284, 391, 449
Tijms, H. C., 828, 872n, 879n
Tilanus, C. B., 22
Tirole, J., 743
Todd, M. J., 707
Tucker, A. W., 679n
Tumay, K., 1140

Ulusay, G., 512n

Van Beek, P., 22
Vanderbei, R. J., 172, 220, 284, 321n, 340
Van Dijk, N. M., 890
Van Hoorn, M. H., 879n
Van Roy, Tony J., 623, 631
Van Wassenhove, L., 22
Veinott, A. F., Jr., 1066n
Vogel, W. R., 372n
Voxman, W. L., 171
Vredini, W. A., 510n

Wagner, Harvey M., 533n
Wall, J. A., 1140
Walrand, J., 890
Wang, K. C. P., 1078
Wang, Z., 386n
Waren, A. D., 20n
Weglarz, J., 514
White, D. J., 1078
Whitt, W., 1140
Whittle, P., 1078
Williams, H. P., 22, 90, 631
Williams, T. A., 89
Wirth, R., 10n
Wolfe, P., 684n, 698n
Wolff, R. W., 890
Wolsey, Laurence A., 623, 631
Wong, R. T., 449
Wright, G., 1038
Wright, M. H., 707
Wunderling, R., 631

Young, W., 10n
Yu, O. S., 873n
Yurkiewicz, J., 1038

Zang, I., 705n
Zaniewski, D. J., 1078
Zehner, C. M., 668n
Zhang, Y., 510n
Zheng, Y-S., 958n
Ziemba, W. T., 669

SUBJECT INDEX

ABC control method, 983
Absorbing state, 811, 820–822
Acceptance-rejection method, 1109–1110
Accessible state, 810
Action, choosing, 751
Activity durations
crashing decisions, 494–502
critical path, 475–476
individual activities, 477–482
means and variances of, 509–510
precedence diagramming method, 511
probability of meeting deadlines, 491, 510–511, 1098
time-cost trade-offs, 492–502
uncertain, 485–492
Activity-on-arc, 471
Activity-on-node, 471
Actual average cost per unit time, 815
Additive congruential method, 1105
Additivity assumption, 37n, 40–42
distribution of effort problem, 549
Adjacent basic feasible solution, 117
Adjacent corner-point feasible solution, 110–111, 113, 193–195
Advisory capacity, 8
Airline industry, 584–585
Airplanes, 834
Air pollution control, 50–52, 302–304
Air Products and Chemical Inc., 583
Albuquerque Microelectronics Operation, 1011
Algebraic language, 114–115
Algebraic procedure, 109
Algorithmic research procedure, 665, 702–703
Algorithms, 5, 14
convex programming, 697–698
general-purpose, 386
interior-point, 320–332
periodic review, 953–956
Algorithms—*Cont.*
policy improvement, 1064–1068
sequential approximation, 697–698
sequential unconstrained, 697
shortest-path, 411–413
streamlined, 350
Allowable range to stay feasible, 159, 255, 265–266, 267
Allowable range to stay optimal, 158, 170, 255, 266, 270–271, 277–278
American Airlines, 584, 963, 1010, 1011, 1029
Amoco Oil Company, 780
AMPL, 73
Animation capabilities, 1113
Antithetic variates, 1129n
Aperiodic states, 812
Application-oriented simulations, 1112–1113
Arc capacity, 410
Arcs, 407–408
decision tree, 765
ARIMA method, 1026, 1037
Arithmetic multiplication, 1170
Artificial problem, 132
Artificial variables, 132, 134, 242n, 310
Artificial-variable technique
definition, 132
equality constraints, 132–136
functional constraints in ≥ form, 137–140
minimization, 140
negative right-hand sides, 136–137
no feasible solution, 148–149
radiation therapy example, 137–138
two-phase method, 142–148
variable allowed to be negative
with a bound, 150
with no bound, 151–152
Asset divestitures, 584
Assignees, 381
Assignment problem, 350, 381–390, 601
assigning products to plants, 386–389
case, 402–404
and minimum cost flow problem, 435
model and solution procedures, 383–390
prototype example, 382–383
Asymptotic confidence interval, 1135
@RISK software, 1115–1116, 1117–1118, 1121, 1124–1125
AT&T, 4, 163, 320, 581, 925, 1098
Augmented corner-point feasible solution, 116, 198–199
Augmented corner-point solution, 116, 198–199
Augmented form of the model, 118, 203
Augmented form of the problem, 115
Augmented solution, 115–116
Augmenting path, 423, 426–427
Augmenting path algorithm, 422, 423–424
Ault Foods Limited, 582
Auto assembly, case, 103–104
Autocorrelation, 1027
Automatic problem preprocessing, 624–628
Auxiliary binary variables, 585, 586, 592–594, 595–598
Auxiliary variables, 334
Average computer time per iteration, 166
Average cost criterion, 1069–1070
Averaging forecasting method, 1017

Backlogging, 939
Backorders, 943
Backward induction procedure, 769
Backward pass, 481
Balance equation, 826, 850
Balking, 836
Barrier function, 697, 703
Basic arcs, 439–440

Basic feasible solution, 116, 198–199, 601
adjacent, 117
augmented, 116
constructing, 368
degenerate, 200
and feasible spanning tree, 439–440
initial, 118–119, 121, 132, 134, 148–149
leaving basic variable, 444–448
with northwest corner rule, 369–370
optimal, 131
optimality test for, 122–123
in revised simplex method, 204–206
solving for the new, 121–122, 126
transportation problem, 378
Basic solutions, 116, 198–199
classification of, 246
Basic variables, 118, 124, 199
changes in coefficients of, 274–278
vector of, 204
Basis, the, 116
Basis for a set of vectors, 1172
Basis matrix, 205
initial, 211
Batches, 1131
Batista, Fulgencio, 347–349
Bayes' decision rule, 760–761, 769, 770–771, 773
definition, 754–758
sensitivity analysis with, 755–758
Bayes' theorem, 759
Bell Laboratories, 163, 320
Best bound, 609
Beta distribution, 486
BF solution; *see* Basic feasible solution
Big M method, 134–136, 139–143, 148–149, 338, 361, 388
Bill of materials, 950
Binary integer programming, 577
applications
airline industry, 584–585
capital budgeting, 580–581
dispatching shipments, 582–583
production and distribution, 581–582
scheduling asset divestiture, 584
scheduling interrelated activities, 583–584
site selection, 581
branch-and-bound technique, 604–616
bounding, 606–607
branching, 605–606
completing the example, 609–613
fathoming, 607–609
other options with, 613–616
summary of, 608–609
cases, 648–653
Binary integer programming—*Cont.*
prototype example, 578–579
recent developments in problems solving
automatic problem preprocessing, 624–628
background, 622–624
generating cutting planes, 628–630
solving problems, 600–604
Binary representation, 590–591
Binary variables, 383, 577
integer programming formulation, 585–600
Binding constraints, 155
Binomial distribution, 563
BIP; *see* Binary integer programming
B_i parameter, 262–268
Birth, 848
Birth and death process
assumptions, 848
balance equation, 850
definition, 848
description, 848–852
M/M/s model, 852–860
County Hospital example, 858–860
finite calling population variation, 864–866
finite queue variation, 861–863
multiple-server case, 856–858, 863
single-server case, 854–856, 861–863
model with state-dependent service rate and/or arrival rate, 866–871
multiple-server case, 865–866
queueing models based on, 852–870
rate in = rate out principle, 850–852
single-server case, 865
steady-state condition, 849
transient condition, 849
Black-Scholes option pricing model, 1155
Blending program, 56–57
Bolzano search plan, 671
Bottom-up approach forecasting, 1014
Boundary of the feasible region, 190–191, 193–194, 704
Boundary repulsion term, 703
Bounded feasible region, 593
Bounding, 605, 606–607
binary integer programming, 605, 606–607, 608
mixed integer programming, 618
Bounding step, 618
Box-Jenkins method, 1026–1028, 1037
B_1 parameters, 315–317
Branch-and-bound algorithm, 608–609, 616–622
Branch-and-bound technique
in binary integer programming, 604–606
integer programming, 604–606
Branch-and-cut algorithm, 623
Branches, 765
Branching, 605–606
binary integer programming, 606–606, 608–609, 613
mixed integer programming, 618
Branching variable, 606, 613, 618
British Columbia Hydro and Power Authority, 780
Bureau of Labor Statistics, 3
Businesses
organizational complexity, 1–2
parties affected by, 9

Cafeteria costs, case, 104–106
Calculus, 556
Call centers, 1012
Call center staffing, case, 106–108
Calling population, 835, 864–866
Capital budgeting problem, 580–581
Carrying costs, 935
Castro, Fidel, 347
Causal forecasting, 1028–1029
Centering scheme, 325–326
Certainty assumption, 43, 156
distribution of effort problem, 548
Chain reaction, 37, 378
Chance fork, 765
Chapman-Kolmogorov equations, 808–810
China, 4, 583–584
Chi-square distribution, 1108–1109
Citgo Petroleum Corporation, 4, 10, 18, 73, 405, 430
C_j parameters, 313–315
Classical optimization methods
constrained optimization with equality constraints, 1166–1167
derivative of a definite integral, 1167–1168
unconstrained optimization function
several variables, 1166
single variable, 1165–1166
Class property, 811
Coefficient reduction, 626n
Coin-flipping game, 1087–1092
Column rank of a matrix, 1172–1173
Column vector, 1171
Commercial service, 840–841
Communicating states, 810
Complementarity constraint, 670, 685–686
Complementarity problem, 669–670
Complementary basic solutions, 242–245, 310
relationships between, 245–247
Complementary decision variables, 318

Complementary optimal basic solution, 245–246, 253–254, 312n
Complementary optimal solutions, 237
Complementary optimal solution **y**, 237
Complementary random numbers, 1129
Complementary slackness, 243–244
Complementary slackness property, 237
Complementary solutions, 236–237, 238
Complementary solution **y**, 236
Complementary variables, 685–686
Computational difficulty, 601
Computationally infeasible model, 591
Computer-based system, 19
Computer Club Warehouse, 1019–1020, 1023–1025
Computerized inventory systems, 956–957, 984–985
Computer revolution, 2
Computers, and simplex method, 160–163
Concave downward, 662n
Concave functions, 316, 662–664
  global maximum, 1166
  of several variables, 1161–1163
  of single variable, 1159–1161
Concave upward, 662n
Condition for feasibility, 246
Condition for optimality, 246
Confidence coefficient, 1135
Confidence interval, 1033–1034, 1035, 1090, 1135
Congressional Budget Office, 1012
Congruential random number generation, 1103–1105
Connected network, 410
Connected nodes, 410
Conservation of flow, 410
Constant lead time, 942
Constant-level model forecasting
  averaging method, 1017
  exponential smoothing, 1017–1018
  exponential smoothing for linear trend, 1021–1025
  last-value method, 1016
  moving-average method, 1017
  and time series forecasting, 1018–1019
Constant-level time series, 1015, 1016
Constrained optimization, 679–683, 1166–1167
Constrained problems, 665
Constraint boundary, 109–110, 191
Constraint boundary equation, 190–191
  indicating variables for, 199
Constraints, 11, 33
  tightening, 620–622
Construction company problem, 1118–1122
Consumer market survey, 1014
Contingent decisions, 578, 579, 589, 601
Continuous demand, 966
Continuous random variable, 966
Continuous review, 940–941
  deterministic model, 941–951
  stochastic model, 956–961
Continuous simulation, 1086
Continuous time Markov chains, 822–827, 848, 864, 879, 890
Continuous time stochastic process, 802
Continuous time transition probability function, 823, 824
Contribution to profit, 690
Convex combination, 130
Convex functions, 662–664, 975
  global minimum, 1165–1166
  of several variables, 1161–1163
  of single variable, 1159–1161
Convexity, 1159–1164
  convex sets, 1163–1164
  functions of a single variable, 1159–1161
  functions of several variables, 1161–1163
Convexity test, 1159–1160
Convex programming, 667, 697–702
  algorithms, 697–698
  Frank-Wolfe algorithm, 698–702
  gradient algorithms, 697
  problem, 664
  quadratic approximations, 702
  sequential-approximation algorithm, 697–698
  sequential linear approximation algorithm, 698–702
  sequential unconstrained algorithms, 697
  software options, 702
Convex set, 198, 663, 1163–1164
Cooperative game, 742
Corel Quattro Pro, 162
Corner-point feasible solution, 35, 110–114, 138, 146–148, 166, 190, 601
  adjacent, 193–195
  augmented, 198–199
  and optimal solution, 35–36
  properties of, 195–198
  in simplex method, 191–193
Corner-point infeasible solution, 110, 192–193, 194
Corner-point solutions, 110, 138
  augmented, 198–199
Correlation, 12
Cost assumption, 355
Cost of ordering, 938–939; *see also* Ordering cost
Cost of waiting, 912
Cost overrun, 508
Cost per unit shipped, 656
Costs and benefits, 8
Cost underrun, 508
CPF; *see* Corner-point feasible solution
CPLEX, 6, 78, 159, 161–163, 167, 310, 623, 689–690, 1157–1158
CPM; *see* Critical path method
CPM method of time-cost trade-offs, 493
Crash cost, 494
Crashing activities, 493, 494–496
Crashing techniques, 160n
Crashing the project, 493
Crash point, 493
Crew scheduling problem, 585
Critical path, 475–476, 482, 484–485
Critical path method
  cases, 524–532
  computer use, 509
  dealing with overlapping activities, 511
  evaluation of, 508–512
  future of, 512
  means and variances of activity durations, 509–510
  probability of meeting deadline, 510–511
  for project costs, 502–508
  project scheduling, 475–485
  resource allocation, 511–512
  uses, 468
  value of, 508–509
Critical points, 1165
Crossover point, 657
Cross-product terms, 37n, 40
Crude Monte Carlo techniques, 1126–1127
Crystal Ball, 1115
Cuba, 347–349
Curse of dimensionality, 560
Customers, 9, 430
Cut, 426–427, 628
Cutting planes, 624, 628–630
Cut value, 427
Cycle, 409–410, 1132
Cycle length, 1104

Databases, 19
Data gathering, 7–10, 66, 1111–1112
Data verification, 66
Deadlines, 1098
  construction company problem, 1118–1122
  meeting, 491, 510–511
Death, 848
Decision analysis, 749–801
  cases, 795–800
  choosing an action, 751
  compared to game theory, 728
  conclusions, 781
  decision making with experimentation, 750
    Bayes' theorem, 759–760

Decision analysis—*Cont.*
decision making with experimentation—*Cont.*
posterior probabilities, 758–762
probability tree diagram, 760
prototype example, 758
value of experimentation, 762–764
decision making without experimentation, 750, 751–758
Bayes' decision rule, 754–758
formulation of prototype example, 752
maximin payoff criterion, 752–753
maximum likelihood criterion, 753–754
decision trees, 764–770
backward induction procedure, 768
constructing, 765–767
performing the analysis, 767–769
software for, 769–770
in utility theory, 776–778
environment of uncertainty, 749
graphical techniques, 779
payoff, 751
payoff table, 751
posterior probabilities, 767
practical applications, 778–781
prior distribution, 752
prior probabilities, 752, 767
prototype example, 750
and queueing theory, 909–912
state of nature, 751
utility theory, 770–778
Decision conferencing, 779
Decision fork, 765
Decision models, queueing theory
unknown μ and s, 917–920
unknown D and s, 920–923
unknown s, 917
Decision support system, 19, 20
Decision tree, 562–563
Decision variables, 11, 13, 26, 32, 42, 75, 592–594
Decreasing marginal utility for money, 772
Defining equations, 192, 194, 199
Degenerate basic feasible solution, 200
Degenerate basic variables, 381
Degenerate service-time distribution, 872
Degenerate variables, 129
Degree of association model, 1029
Degrees of freedom, 116
Delphi method, 1014
Delta Air Lines, 4, 585
Demand, 354–355, 936, 957
continuous, 966
dependent/independent, 949
exponential distribution, 974–975
known, 951, 958
unsatisfied, 964
Demand capacity, 364
Demand node, 410, 430–431
Department of Labor, 1012
Dependent demand, 949
Dependent variables, 1030
Descendants, 607
Destinations, 354
Deterministic dynamic programming, 541–562, 953–954
distributing scientists to research teams, 549–552
distribution of effort problem, 547–549
distribution of medical teams, 542–547
scheduling employment levels, 552–559
Wyndor Glass problem, 559–562
Deterministic inventory models
continuous review, 941–951
basic model, 942–943
broader perspective on, 949–951
observations about, 948–949
with planned shortages, 943–946
with quantity discounts, 946–947
templates, 947–948
periodic review, 951–956
algorithm, 953–954
application of algorithm, 955–956
example, 952–953
Deterministic policy, 1057, 1059
Diagonal matrix, 327
Difference, 370
Digital Equipment Corporation, 4, 582
Directed arc, 408, 413, 422–423
Directed network, 408
Directed path, 409
Discounted cost criterion, 1069–1077
linear programming formulation, 1073–1075
method of successive approximations, 1075–1077
policy improvement algorithm, 1070–1073
Discount factor, 940, 980
Discount rate, 940
Discrete-event simulation, 1086
Discrete time Markov chains, 1053, 1054
Discrete time stochastic process, 802
Discretized uniform distribution, 1102
Distribution network, example, 59–61
Distribution of effort problem, 547–549
Distribution proportion, 1128
Distribution systems, 1099
Distribution Unlimited Company, 59–61
Divisibility assumption, 42, 548
Documenting, 21
Dominated strategies, 729–731
Donor cells, 377
Dual, 230
Dual feasible solution, 246–247, 309–310
Duality theory, 230–254, 238, 682–683
adapting to other primal forms, 247–252
economic interpretations
dual problem, 239–241
simplex method, 241–242
essence of, 231–238
applications, 238
origin of dual problem, 232–236
summary of relationships, 236–238
primal-dual relationships, 238, 242–247
and sensitivity analysis, 230, 252–254
Dual problem, 231–232, 311
complementary basic solutions, 243
constructing, 247–249
economic interpretations, 239–241
origin of, 232–236
in simplex method, 238
SOB method, 249–252
Dual simplex method, 153, 309–312
in parametric linear programming, 316
summary of, 310–312
Dual variables, 366
Dummy assignees, 382
Dummy demand node, 431
Dummy destination, 355, 359–362
Dummy machine, 382
Dummy sink, 422
Dummy source, 355, 362–365, 364, 422
Dummy task, 382, 388
Du Pont Corporation, 780
Dynamic programming, 522–568
characteristics, 538–541
deterministic, 541–562
principle of optimality, 540, 549
probabilistic, 562–568
prototype example, 633–638
stagecoach problem, 633–638

Earliest finish time, 477
Earliest start time, 477
Earliest start time rule, 479
Earliest time schedule, 482
Econometric models, 1011–1012
Economic order quantity formula, 943, 951
Economic order quantity model, 941–951
assumptions, 942–943
broader perspective on, 949–951
observations about, 948–949
with planned shortages, 943–946, 957–958
with quantity discounts, 946–947
reorder point, 942
templates, 947–948
Economic trends, 1011–1012

Edge of the feasible region, 110, 113–119, 193, 194
Efficient frontier, 659
Either-or-constraints, 586–587
Elementary algebraic operations, 121–122
Elementary row operations, 126, 146
Employees, 9
Employment level scheduling, 552–559
Energy Electric System, 781
Entering basic variable, 120, 311
network simplex method, 441–444
tie-breaking for, 128–129
Enumeration procedure, 604
Enumeration tree, 606
Environmental Protection Agency, 50n
EOQ; *see* Economic order quantity model
Equality constraint, 66, 114–115, 705, 1166–1167
in artificial-variable technique, 132–136
Equilibrium solution, 732
Equivalence property, 886
Ergodic Markov chain, 813
Erlang distribution, 837, 838, 855, 873–875, 878–879, 890, 1107–1108
Erlang's loss system, 863
Error tolerance, 704
Excel Solver, 5–6, 31, 67–72, 157–159, 162, 163, 266, 272, 358–365, 389–390, 413–414, 428–429, 434–435, 579–580, 689–690, 769–770, 947–948, 1112, 1115–1116, 1156–1157
Excel VBA macro language, 73
Excess demand capacity, 364
Excess supply capacity, 364
Expected average cost per unit time, 1055, 1077
for complex cost functions, 816–818
in Markov chains, 814–816
Expected interarrival time, 839
Expected monetary value criterion, 754n
Expected payoff, decision trees, 767–769
Expected recurrence time, 820
Expected return, 658
Expected total discounted cost, 1069, 1077
Expected utility, 773
Expected value
and experimentation, 764
of perfect information, 762–763
probability theory definition, 734
Experimentation
decision making with, 758–764
decision making without, 751–758
expected value of, 764
value of, 762–764
Exponential demand distribution, 974–975
Exponential distribution, 825, 836, 858, 890, 1107–1108
calling population, 866
with a parameter, 842
role in queueing theory, 841–848
Exponential growth, 600
Exponential smoothing forecasting procedure, 1017–1018
for linear trend model, 1021–1025
Exponential time algorithm, 165–166
Exponential utility function, 776
Extreme point, 1164

Fair game, 731
Fanning-out procedure, 426
Farm management, case, 304–307
Fathom, 607
Fathoming, 605
binary integer programming, 607–609, 614–618
integer programming, 614–616
mixed integer programming, 618
Feasibility, condition for, 246
Feasibility test, 311
in sensitivity analysis, 261
Feasible region, 28–29, 34, 46, 110, 663
Feasible solution, 34
Feasible solutions property, 355, 362, 432
Feasible spanning tree, 439–440
Federal National Mortgage Association, 781
Field test, 1113
Financial risk analysis, 1099–1100, 1122–1126
Finite calling population variation, 908
of M/M/s model, 864–866
Finite queues, 836
Finite queue variation of M/M/s model, 861–863
Finite state space, 802
First local maximum, 678
First passage times, 818–820
First-priority rules, 336, 337
First-stage objective function, 337
Fixed lead time, 940
Fixed-time incrementing, 1093–1095, 1132
Fixing variables, 624–625
Fleet assignment problem, 584
Flow diagram, 1111
Flow in, 414
Flow out, 414
Flows, 410
Forecasting, 64, 1009–1038
applications
economic trends, 1011–1012
production yields, 10111
sales forecasting, 1010
spare parts needs, 1010–1011
Forecasting—*Cont.*
applications—*Cont.*
staffing needs, 1012–1013
Box-Jenkins method, 1026–1028
case, 1048–1052
causal, 1028–1029
conclusions, 1038
constant-level time series, 1016–1018
exponential smoothing for nonlinear trend model, 1021–1025
judgmental methods, 1009, 1037
consumer market survey, 1014
Delphi method, 1014
jury of executive opinion, 1014
manager's opinion, 1013
sales for composite, 1014
linear regression, 1028–1036
measures of uncertainty, 1035–1036
more than one time period ahead, 1025
in practice, 1036–1038
regression analysis, 1009
with seasonal effects, 1018–1021
statistical methods, 1009, 1037
applications, 1013
time series, 1009, 1014–1016
Forecasting errors, 1025–1026
Forks, 765
Forward pass, 480–482
Fractional programming, 668–669
Frank-Wolfe algorithm, 698–702
Franz Edelman Awards for Management Science Achievement, 581, 923
Free goods, 155
Frequency distribution, 1122
Frontline systems, 162
Fully crashed activities, 493, 495
Functional constraints, 33, 75–76, 77–78, 161, 318
Functional constraints in ≥ form, 137–140
Functional relationship, 1029–1030
Function value, 42
Fundamental insight, 208, 212–220
adapting to other model forms, 218–219
applications, 219–220
duality theory, 232–234
mathematical summary, 216–218
in sensitivity analysis, 255–256
transportation simplex method, 365–366
verbal description, 212
Fundamental theorem, network simplex method, 440

Game theory
compared to decision analysis, 728
cooperative game, 742

Game theory—*Cont.*
definition, 726
dominated strategies, 729–731
extensions, 741–742
fair game, 731
graphical solution procedure, 735–738
infinite games, 742
minimax criterion, 732
minimax theorem, 734
mixed strategies, 733–735
noncooperative game, 742
nonzero-sum game, 742
n-person game, 742
primary objective, 727
solved by linear programming, 738–741
solving simple games, 728–733
stable solution, 732
strategies, 726–728
two-person, zero-sum game, 726–729, 742
unstable solution, 733
value of the game, 731
Gamma distribution, 855n, 873
GAMS, 73
Gantt charts, 472
Gasoline blending, 56–57
Gaussian elimination, 118, 122, 135, 141, 146, 213, 260–261, 262, 274, 278, 283
General Agreement on Tariffs and Trade, 402n
General distribution, 838
Generalized Erlangian distributions, 879
Generalized positive polynomials, 668
Generalized reduced gradient method, 697
General Motors, 780
General-purpose algorithm, 386
General-purpose programming language, 1112–1113
General-purpose simulation language, 1112
Genetic algorithms, 604
Geometric concepts, 109
Geometric programming, 668
Global maximization, 668
Global maximum, 662, 704, 1165
Global minimum, 662, 1165
Global supply chain, 582
Goal programming, 309
nonpreemptive, 333–335
preemptive, 333, 335–339
and solution procedures, 332–333
types of goals, 332–333
Government, 9
Government space project, 549–552
Gradient, 674
of objective function, 322
projected, 323–325
relevance of, 321–323
Gradient algorithms, 697, 702
Gradient search procedure, 674–678
Graphical method, 30, 46
decision analysis, 779
game theory, 735–738
nonlinear programming, 659–664
Grass-roots forecasting approach, 1014
Group programming, case, 347–349

Health care simulation, 1100
Heuristic algorithms, 605
Heuristics, 624n
Hewlett-Packard, 4, 936, 985–986
Hidden costs, 387
Hidroeléctrica Español, 1010
Histogram, 1118, 1126
Holding cost, 937, 939, 964
Holding time, 837
Homart Development Company, 584
Hyperexponential distribution, 878–879
Hyperplane, 191, 194, 196

IBM, 4, 18, 19, 21, 936, 984–985, 1010, 1105
Optimization Subroutine Library, 623
IBM PC Company, 1098
ICI Americas, 781
Identity matrix, 211, 214, 1171
ILOG, Inc., 161, 162
Immediate predecessors, 469–470, 471
Implementation of models, 20–21
Increasing marginal utility for money, 772
Incremental analysis, 220, 259–260
Incumbent, 607
Independent demand, 949
Independent Living Center, 780
Independent Poisson processes, 887–888
Independent variable, 1030
Indicating variables, 199
Inequality constraints, 114–115
Infeasible solution, 34
Infinite games, 742
Infinite queue, 836
Infinite queues in series, 886–887
Inflection point, 1165
Influence diagram, 779
Initial basic feasible solution, 118–119, 132, 134, 148–149
constructing, 369–372
Initial basis matrix, 211
Initialization, 111–112, 118, 125, 144, 310, 368–374
Initial state probabilities, 1074
Initial transportation simplex tableau, 374, 376
Input constraints, 32
Input source, queueing theory, 835
Integer linear programming, 576
Integer programming, 64, 162, 576–631
BIP applications
airline industry, 584–585
capital budgeting, 580–581
dispatching shipments, 582–583
production and distribution, 581–582
scheduling asset divestiture, 584
scheduling interrelated activities, 583–584
site selection, 581
branch-and-bound technique
bounding, 613
branching, 613
fathoming, 614–616
California Manufacturing Company, 577–580
cases, 642–653
formulation with binary variables, 585–600
with auxiliary binary variables, 593–598
binary representation of general integer variables, 590–591
covering all characteristics, 598–600
either-or constraints, 586–587
examples, 591–600
fixed-charge problem, 589–590
functions with *n* possible values, 588–589
K out of N constraints, 587
violating proportionality, 594–598
when decision variables are continuous, 592–594
prototype example, 577–580
software options, 579–580
solving problems, 600–604
special type of problems, 601
Integer programming models, 42
Integer-restricted variables, 616–618
Integer solutions property, 357, 381, 384, 433, 1063
Interactive sessions, 21
Interactive system, 63
Interarrival time, 836, 838
expected, 839
probability distribution, 841
Interfaces, 3–5, 354, 430, 431, 580–585, 779–781, 923, 924, 925, 963, 1013, 1099, 1100
Interior-point algorithm, 25, 164, 320–332, 703
centering scheme, 325–326
gradient
projected, 323–325
relevance, 321–323
illustration of, 327–332
summary of, 328
Interior-point approach, 163–168
compared to simplex, 165–166
complementary role with simplex, 167–168
key solution concept, 164–165

Interior points, 164
Intermediate storage facilities, 430
Internal service, 841
International Federation of Operational Research Societies, 3
International Mathematics and Statistics Library, 1105
International Paper Company, 430
Interrelated activities scheduling, 583–584
Interrelated decisions, 543
Intersection of constraint boundaries, 194
Inventories
  backlogging, 939
  backorders, 943
  carrying cost, 935
  computerized systems, 956–957, 984–985
  continuous review, 940–941
  discount factor, 940
  discount rate, 940
  holding cost, 937
  no backlogging, 939
  ordering cost, 937
  periodic review, 940–941
  replenishment, 938
  revenue, 939–940
  safety stock, 959
  salvage cost, 940
  salvage value, 940
  setup cost, 937
  shortage cost, 937
  and simulation, 1098
  spare parts needs, 1010–1011
  stockout, 943
  surplus stock, 1004–1008
  two-bin system, 956
  unit production cost, 937
Inventory control, 1000–1002
Inventory holding cost rate, 948
Inventory policy, 936, 957
  just-in-time system, 935, 950–951
  optimal, 953
  simulation, 1116–1118
Inventory theory, 935–987
  cases, 1000–1008
  components of models, 938–941
  conclusions, 987
  and costs, 937–939
  deterministic models
    continuous review, 941–951
    periodic review, 951–956
  economic order quantity model, 941–951
  examples, 936–938
  multiechelon systems, 984–985
  multiproduct systems, 983
Inventory theory—*Cont.*
  stochastic models
    continuous review, 956–961
    periodic review, 975–983
    single-period for perishable products, 961–975
  stochastic versus deterministic models, 940
  supply chain management, 985–986
Inverse of a matrix, 1173
Inverse transformation method, 1106–1107
IP; *see* Integer programming
Irreducible ergodic Markov chain, 813
Irreducible Markov chains, 811
Iterations, 111–112, 113, 120–122, 125–126, 161, 166, 311, 376–379, 1116
Iterative algorithm, 113, 164

Jackson networks, 887–889
Jet engine problem, 359–362
Joint probability, 886, 889
Judgmental forecasting methods, 1009, 1037
  consumer market survey, 1014
  Delphi method, 1014
  jury of executive opinion, 1014
  manager's opinion, 1013
  sales for composite, 1014
Jury of executive opinion, 1014
Just-in-time inventory system, 935, 950–951

Karush-Kuhn-Tucker conditions, 705
  for constrained optimization, 679–683
  for quadratic programming, 685–686
KayCorp, 925
Key solution concept, 164–165
Known constant, 43
Known demand, 936, 951, 958

L. L. Bean, 1013
Lag, 1027
Lagrange multipliers, 680, 1167
Lagrangian function, 702, 1167
Lagrangian relaxation, 613
Last chance schedule, 481–482
Last-value forecasting method, 1016
Las Vegas problem, 565–567
Latest finish time, 480–482
Latest finish time rule, 480
Latest start time, 480–482
Lead time, 940, 942, 957
Learmouth-Lewis generator, 1105
Learning curve effect, 656
Leaving basic variable, 121, 150, 311
  network simplex method, 444–448
  tie-breaking for, 129
Length of a cycle, 1132
Length of a path, 475
LINDO, 6, 31, 78–79, 159, 162–163, 266, 272, 579–580, 689–690, 702, 1158
  introduction to, 169–171
LINDO System, Inc., 74
Linear approximation method, 697–698
Linear complementarity problem, 670
Linear fractional programming, 669
Linear function, 24, 913
Linear goal programming; *see* Goal programming
Linearly constrained optimization, 665
Linearly dependent vector, 1172
Linearly independent vector, 1172
Linear programming, 2, 24–89; *see also* Parametric linear programming; Simplex method; Upper bound technique
  additional examples
    controlling pollution, 50–52
    distribution network, 59–61
    personnel scheduling, 57–59
    radiation therapy, 44–46
    reclaiming solid wastes, 53–57
    regional planning, 46–49
  assumptions
    additivity, 40–42
    certainty, 42, 156
    divisibility, 42–43
    in perspective, 43–44
    proportionality, 36–40
  cases, 61–67
    auto assembly, 103–104
    cafeteria costs, 104–106
    call center staffing, 106–108
    personnel scheduling, 63–65
    petroleum industry, 65–67
    product mix, 62–63
  common applications, 24
  compared to integer programming, 600–604
  for crashing decisions, 496–501
  duality concept, 230
  interior-point algorithm, 163–168
  Markov decision processes, 1059–1064
    discounted cost criterion, 1073–1075
    formulation of, 1061–1063
    randomized policies, 1060–1061
    solving example by, 1063
  maximum flow problem, 422
  modeling languages, 73–74
  network optimization models, 405–406
  parametric, 159–160
  postoptimality analysis, 152
  primal-dual table, 231–233

Linear programming—*Cont.*
problem formulation, 45–46, 48–49, 52–56, 58–59, 60–61
conclusions, 30
graphical solution, 27–30
and OR Courseware, 30–31
problem formulation, 26–27
and separable programming, 692–696
of simplex method, 160–163
with slack variables, 198–199
software, 161–163
to solve game theory, 738–741
on spreadsheets, 67–72
terminology for, 31–33
Linear programming constraints, 685–686
Linear programming model, 11, 25, 31–36
conversion to standard form, 248
LINGO formulation, 82–86
minimum cost flow problem, 433
MPL formulation, 76–78
other legitimate forms, 33
standard form, 32–33
terminology for solutions, 33–36
very large, 72–79
Linear regression, 1028–1036, 1037
confidence interval, 1033–1034, 1035
degree of association model, 1029
dependent variable, 1030
functional relationship, 1029–1030
independent variable, 1030
method of least squares, 1030–1033
prediction interval, 1034–1036
regression line, 1032
simultaneous tolerance levels, 1035–1036
Linear trend, 1022
Linear trend time series, 1015
exponential smoothing method, 1021–1025
Line segment, 193, 1161–1162
LINGO, 6, 31, 73, 74, 78–89, 163, 689–690, 702, 1158
Links, 408
Little's formula, 840, 881, 889
Local maximization, 668
Local maximum, 704
Local minimum, 1165
Long-run profit maximization, 8
Lost revenue from unsatisfied demand, 964
Lotus 1-2-3, 10, 162
Lower, one-sided goal, 332
LP relaxation, 601–603, 606–607, 609, 616, 630

Machines, 834
Mainframe computers, 161
Management information system, 9, 19
Management science, 2
Managerial decisions, 230
Managerial reports, 19
Manager's opinion forecasting, 1013
Manufacturing jobs, 834
Manufacturing systems, 1099
Marginal cost, 656–657, 915
Marginal cost analysis, 495
Marginal profitability, 690
Marginal return, 38–39
Marketing costs, 39
Markov chains, 802–827, 1053, 1054
absorbing states, 820–822
Chapman-Kolmogorov equations, 808–810
classification of, 810–812
absorbing state, 811–812
periodicity properties, 812
recurrent and transition states, 811–812
continuous time, 864, 879
in birth and death process, 848
example, 826–827
formulation, 822–823
key random variables, 823–825
steady-state probabilities, 825–827
first passage times, 818–820
formulating example, 805–807
gambling example, 807–808
inventory example, 803
irreducible, 811
long-run properties
expected average cost per unit time, 814–816
expected average cost per unit time for complex cost functions, 816–818
steady-state probabilities, 812–814
Markovian property, 803, 805, 823
n-step transition probabilities, 804
random walk, 821
stationary transition probabilities, 803–804
stochastic processes, 802
stock examples, 807
and transition matrix, 804–805
transition probabilities, 803–804
Markov decision processes, 1053–1077
conclusions, 1077
deterministic policy, 1057
discounted cost criterion, 1069–1077
expected average cost per unit time, 1055
linear programming and optimal policies, 1058–1064
formulation of, 1061–1063
randomized policies, 1060–1061
solving example by, 1063
model, 1056–1059
policy improvement algorithm, 1064–1069
Markov decision processes—*Cont.*
prototype example, 1053–1056
cost data, 1056
discounted cost method, 1072–1073
and linear programming, 1061, 1063
method of successive approximations, 1076–1077
solved by linear programming, 1074–1074
solved by policy improvement algorithm, 1066–1069
solving by exhaustive enumeration, 1057–1059
stationary policy, 1057
Markovian distribution, 838
Markovian property, 540, 803, 805, 823, 1132
Marshall's, Inc., 431
Master production schedule, 950
Material requirements planning, 949–950
Mathematical models
advantages, 11–12
applying, 18–20
assignment problem, 383
deriving solutions from, 14–16
formulating, 10–13
implementation, 20–21
inventory system, 936
in linear programming, 24–25
modeling languages, 73–74
pitfalls, 12
testing, 16–18
transportation problem, 360–362, 363–364
Matrices, 1169–1173
basis, 205
column rank, 1172–1173
diagonal, 327
identity, 211, 214, 1171
initial basis, 211
inverse of, 1173
multiplying, 1169–1170
nonsingular, 1173
null, 1171
operations of, 1169–1171
positive semidefinite, 684
projection, 323
row rank, 1172–1173
singular, 1173
submatrices, 1171
subtraction, 1169
transition, 804–805, 809–810, 812
transpose operation, 1170
vectors, 1171–1172
Matrix form, 160
Matrix form of equations, 206–208
Matrix multiplication, 1170
Max-flow min-cut theorem, 426–427

Maximal Software, Inc., 73
Maximin payoff criterion, 752–753
Maximum flow problem, 406, 428–429, 601
  algorithm, 422–424
  applications, 421–422
  augmenting path algorithm, 423–424
  finding augmenting path, 426–427
  and minimum cost flow problem, 436–437
  using Excel, 428–429
Maximum likelihood criterion, 753–754
Mean, 658
Mean absolute deviation, 1026
Mean critical path, 489
Mean leaving rate, 850
Mean square error, 1026
Measures of forecast uncertainty, 1035–1036
Medical team distribution, 542–547
Merit Brass Company, 1010
Metaheuristics, 605
Method of Lagrange multipliers, 1167
Method of least squares, 1030–1033
Method of successive approximations, 1070, 1075–1077
Microcomputers, 431
Microsoft Corporation, 162
Microsoft Excel, 5–6
Microsoft Project, 468, 472–474, 487, 1158
Midpoint rule, 671
Minimax criterion, 732, 734–735
Minimax theorem, 734
Minimization technique, 703–706
Minimum cost flow problem, 61, 66, 350, 391, 405, 406, 601
  applications, 429–431
  and assignment problem, 435
  description of, 429
  distribution network problem, 433–434
  example, 433–434
  final comments on, 437
  formulation, 431–434
  special cases
    maximum flow problem, 436–437
    shortest-path method, 436
    transportation problem, 435
    transshipment problem, 435–436
  using Excel, 434–435
Minimum cover, 629
Minimum ratio test, 120–121, 125
Minimum spanning tree problem, 410, 415–420
  algorithm, 417–418
  applications, 416–417
  applying algorithm, 418–420
Mixed congruential method, 1103–1105
Mixed integer programming, 40, 576, 589, 590, 594, 601–602
Mixed integer programming—*Cont.*
  branch-and-bound algorithm, 616–622
  example, 619–622
  summary of, 618
M/M/1 queueing theory model, 1092–1096
M/M/s queueing theory model, 852–860
Model, 355
Model enrichment, 12
Model for evaluating technology alternatives, 1100
Modeling languages, 73–74, 78–89, 162
Modeling system, 6
Model validation, 3, 12, 17–18, 65, 161
Modified simplex method for quadratic programming, 684, 686–689
Money, utility functions, 771–773
Monsanto Corporation, 4, 12–13
Monte Carlo techniques, 1126–1127
Most favorable value, 34
Most likely estimate, 486
Moving-average forecasting, 1017
MPL/CPLEX, 31, 163, 579–580, 689–690, 1157–1158
MPL software, 6, 73, 76–78, 163, 702
Multiechelon inventory systems, 984–985
Multiperiod inventory models
  with batch orders and no setup cost, 982
  with setup cost, 981–982
  without setup cost, 980
  variation, 980–981
Multiple optimal solutions, 35
Multiple-server case, 863, 956–858
Multiple servers, 853
Multiplicative congruential method, 1105
Multiproduct inventory systems, 983
Multivariable unconstrained optimization, 673–679
Mutually exclusive alternatives, 578, 579, 583, 588, 601

Naive forecasting model, 1016
National Forest Administration, 780
National Weather Service, 780
Natural language financial planning, 63
Nearly optimal solution, 615–616
Negative right-hand sides, 136–137
Net flow, 414
Net flow constraints, 60–61
Net present value, 577, 1123–1126
Network design, 420
Network optimization models, 405–449
  cases, 458–468
  conclusions, 448–449
  maximum flow problem, 420–429
  minimum cost flow problem, 429–438
Network optimization models—*Cont.*
  minimum spanning tree problem, 410, 415–420
  prototype example, 406–407
  shortest-path problem, 411–415
  terminology for, 407–410
Network representation, 351–354, 384–385
Networks, 350, 539
  components of, 408
  definition, 407
  to display projects, 470–474
Network simplex method, 391, 406, 410, 413, 429, 435, 438–449
  correspondence between BF solution and feasible spanning tree, 439–440
  entering basic variable, 441–444
  fundamental theorem, 440
  leaving basic variable and next BF solution, 444–448
  optimality test, 447–448
  and upper bound technique, 438–439
New England Electric System, 780
New Haven Health Department, 4, 10
Newsboy problem, 962, 1002–1004, 1116–1118
Newsvendor problem, 962n
New York City, 924–925, 1098
Next-event incrementing, 1095–1096, 1132
No backlogging, 939
Node constraints, 432, 434, 439
Nodes, 407–408
  decision tree, 765
No feasible solution, 34, 148–149, 607
No leaving basic variable, 129–130
Nonbasic arcs, 439–440
Nonbasic variables, 116, 118, 120, 124, 199, 204, 245
  changes in coefficients of, 252, 269–273
Nonconvex programming, 668
  search procedures, 702–703
  sequential unconstrained minimization technique, 703–706
Noncooperative game, 742
Nonlinear costs, 975
Nonlinear programming, 40, 322, 653–706
  cases, 720–735
  convexity in, 1159
  convex programming, 697–702
  graphical illustration, 659–664
  Karush-Kuhn-Tucker conditions for constrained optimization, 679–683
  multivariable unconstrained optimization, 673–679
  nonconvex programming, 702–706
  one-variable unconstrained optimization, 670–673

Nonlinear programming—*Cont.*
quadratic programming, 683–690
sample applications
portfolio selection, 658–659
product-mix problem, 655–656
transportation problem, 656–568
separable programming, 690–697
types of problems, 664–670
complementarity problem, 669–670
convex programming, 667
fractional programming, 668–669
geometric programming, 668
linearly constrained optimization, 665
nonconvex programming, 668
quadratic programming, 665–667
separable programming, 667
unconstrained optimization, 665, 667
Nonnegative artificial variable, 134
Nonnegative variables, 120
Nonnegativity constraints, 33, 60–61, 114–115, 116, 318, 667
Nonpreemptive goal programming, 333–335
Nonpreemptive priorities, 880, 881–882
Nonsingular matrix, 1173
Nonzero-sum game, 742
No optimal solutions, 35
Nori and Leets Company, 50–52, 160, 590
Normal cost, 494
Normal distribution, 490, 1108–1109
Northwest corner rule, 369–370
compared to other criteria, 373
n-person game, 742
NPV; *see* Net present value
n-step transition probabilities, 804, 808–810
Null matrix, 1171
Null vector, 203, 1172
Numerical instability, 705

Objective function, 11, 13, 33, 75
deterministic dynamic programming, 541–542
gradient, 322
simultaneous changes in coefficients, 273
Objectives, 8–10
Odds and evens game, 726
Oglethorpe Power Corporation, 781
Ohio Edison Company, 780
OMEGA, 20
One-dimensional search procedure, 670–673, 698
100 percent rule
for changes in objective function coefficients, 273
for changes in right-hand sides, 267–268
One-variable unconstrained optimization, 670–673
Open DataBase Connectivity, 87, 89
Operations research
algorithms, 5–6
applications, 4
data gathering, 7–10
impact of, 3–5
mathematical models, 10–20
nature of, 2–3
OR Courseware, 5–6
origins of, 1–2
phases of study, 7
problem definition, 7–10
role of simulation, 1085–1086
OPL Studio, 162
Optimal basic feasible solution, 131
Optimality
condition for, 246
necessary and sufficient conditions, 679
principle of, 540, 549
Optimality test, 111–112, 609
BF solution, 122–123
duality theory, 235
mixed integer programming, 618
network simplex method, 447–448
in sensitivity analysis, 261
simplex method, 125
transportation simplex method, 375–376
Optimal mixed strategy, 737
Optimal policy, 539–540
for inventory, 953, 969–971
Markov decision processes, 1059–1064
policy improvement algorithm, 1070–1073
Optimal policy decision, 540
Optimal production schedule, 953, 956
Optimal service level, 967
Optimal solution, 3, 14, 30, 34, 539
and CPF solutions, 195–198
for dual problem, 235
multiple, 130–132
related to corner-point feasible solution, 35–36
separable programming, 696
simplex method, 122–123, 127–128
OptiMax 2000, 73, 163
Optimistic estimate, 486
Optimization methods; *see* Classical optimization methods
Optimization Subroutine Library, 623
Optimizer, 19, 21
Optimizer system, 984–985
OR Courseware, 5–6, 30–31, 123, 262, 468, 1156–1158
Ordering cost, 937, 964
Order quantity, 958
Order-quantity policy, 957
Organizational complexity, 1–2
Origin, 113
Original form of the model, 118
OR/MS Today, 1098
OR Tutor, 5, 28, 30, 115, 123, 163, 262
Overall measure of performance, 12, 31
Overhead costs, 495
Owners, 9

P and T Company problem, 351–355
Panels of experts, 1014
Parameters of a model, 11, 13, 32, 255, 355
Parameter table, 355–356
Parametric linear programming, 159–160, 309, 312–317
in sensitivity analysis, 280–284
systematic changes in b_1 parameters, 315–317
systematic changes in C_j parameters, 313–315
Parametric programming, 659
Partial autocorrelation, 1027
Partially crashing an activity, 493–494
Parts Inventory Management System, 19
Patent protection, 402n
Path, 408
Payoff, 751
Payoff table
in decision analysis, 751
solving simple games with, 728–733
and strategies, 726–727, 728–729
Penalty function method, 697
Penalty points, 334
Penalty weights, 333
Perfect information, 762–763
Periodicity properties, 812
Periodic review, 940–941
Periodic review inventory models, 951–956
Period of a state, 812
Perishable products, 961–975
Personnel scheduling, 57–59, 63–65
PERT; *see* Program evaluation and review technique
PERT three-estimate approach, 486–487, 1098, 1118, 1120–1121
Pessimistic estimate, 486
Petroleum industry, case, 65–67
Phase-type distributions, 879
Phillips Petroleum, 781
Piecewise linear function, 316, 609–691, 656, 696–697
PIMS; *see* Parts Inventory Management System
Pivot column, 125
Pivot number, 126
Pivot row, 126

Planning, 24
Point estimate, 1135
Poisson distribution, 803, 805, 845–846
Poisson input process, 846, 858, 871, 872, 876–878, 880, 886
Poisson process, 836, 846, 886
  independent, 887–888
Policy decision, 539
Policy improvement algorithm, 1070–1073
Pollaczek-Khintchine formula, 872, 873, 875
Polygon, 193
Polyhedron, 193–194
Polynomial time algorithm, 165–166
Ponderosa Industrial, 62–63
Portfolio selection with risky securities, 658–659
Positive semidefinite matrix, 684
Posterior probabilities, 758–762, 767
Postoptimality analysis, 14, 15, 67, 109
  and interior-point approach, 166, 167
  for linear programming, 152
  parametric linear programming, 159–160
  reoptimization, 152–153
  sensitivity analysis, 156–159
  shadow prices, 153–156
Postoptimality tasks, 219
Posynomials, 668
Precedence diagramming method, 511
Prediction interval, 1034–1036
Preemptive goal programming, 333, 335–339
  more than two priority levels, 339
  sequential procedure, 336–338
  streamlined procedure, 338–339
Preemptive priorities, 880, 882–885
Preimplementation test, 18
Premium Solver, 69, 162, 163
Price demand curve, 655
Price elasticity, 655
Primal, 230
Primal-dual relationships, 230
  complementary basic solutions, 242–245, 312n
  between complementary basic solutions, 245–247
  complementary optimal basic solution, 245–246, 253–254
  complementary optimal solutions, 237
  complementary slackness property, 243–244
  complementary solutions property, 236–237, 238
  corresponding forms, 250
  duality theorem, 238
  strong duality property, 236, 238
  summary of, 236–238
  symmetry property, 237, 248
  weak duality property, 236, 238
Primal-dual table, 231–233
Primal feasible solution, 246–247, 309–310
Primal problem, 231–232, 683
  adapting to other primal forms, 247–252
  complementary basic solutions, 243
Principle of optimality, 540, 549, 1075
Prior distribution, 752
Priority classes, 880
Priority-discipline queueing models, 879–885
  County Hospital example, 883–885
  description, 879–881
  nonpreemptive priorities, 880
  preemptive priorities, 880
  results for nonpreemptive priorities, 881–882
  results for preemptive priorities, 882–883
  single-server variations, 882
Prior probabilities, 752, 767
Probabilistic dynamic programming, 562–568, 1070
  decision tree, 562–563
  reject allowances problem, 563–565
  winning in Las Vegas, 565–567
Probability density function, 842
Probability distribution, 562, 1060, 1102
  generation of random numbers from, 1105–1110
  for project duration, 488–490, 501
Probability of absorption, 820–821
Probability theory, definition of expected value, 734
Probability tree diagram, 760
Problem definition, 7–10
Processing facilities, 430
Procter and Gamble, 4, 354
Product form solution, 886
Production and distribution network, 581–582
Production cost, 939
Production rates, 25–26
Production schedule
  master, 950
  optimal, 953, 956
Production yield forecasts, 1011
Product mix, 26
  case, 62–63
  in nonlinear program, 655–656
  problem, 74
Products
  dependent-demand, 949
  perishable, 961–975
  stable, 961, 975–983
Product splitting, 387
Profit curves, 609–691, 694
Profit function, 38, 39, 656
Profit maximization, 8, 25–26
Profit per batch produced, 26, 34
Program evaluation and review technique, 468–513, 1111
  cases, 524–532
  computer use, 509
  dealing with overlapping activities, 511
  evaluation of, 508–512
  future of, 512
  means and variances of activity durations, 509–510
  probability of meeting deadline, 510–511
  for project costs, 502–508
  project scheduling, 475–485
  resource allocation, 511–512
  in simulation model, 1118–1122
  uses, 468
  value of, 508–509
Programming, 24
Programming languages, 1112–1113
Project costs
  controlling, 502, 506–508
  scheduling, 502–506
Project deadline, 1098, 1118–1122
Project duration, 475
  probability distribution, 488–490, 501
  simplifying approximations, 487–490
Projected gradient, 323–325
Projection matrix, 323
Project management
  cases, 524–532
  earliest start time, 477–482
  earliest finish time, 477–482
  evaluation of PERT/CPM, 508–512
  with MS Project, 472–474
  network visual display, 470–474
  PERT/CPM for, 468–469
  prototype example, 469–470
  scheduling and controlling costs, 502–508
  scheduling with PERT/CPM, 475–485
  time-cost trade-offs, 491, 492–502
  uncertain activity durations, 485–492
    meeting deadline, 491
    PERT three-estimate approach, 486–487
    simplifying approximations, 487–490
Project management software, 468, 509
Project network, 471–472
Project scheduling
  critical path, 475–476
  identifying slack, 482–485
  individual activities, 477–482
  review of, 485
Proof by contradiction, 195–196
Proper form of the Gaussian elimination, 118
Proportionality
  analysis, 62
  assumption, 36–40, 548

Proportionality—*Cont.*
violating, 594–598
Pseudo-random numbers, 1103
Pure strategies, 733

Quadratic approximation method, 697–698
Quadratic programming, 162, 665–667, 683–690
Korush-Kuhn-Tucker conditions, 685–686
modified simplex method, 684, 686–689
software options, 689–690
Quality constraint, in primal problem, 248–249
Quality Stores, 583
Quantity discounts, 946–947
Quasi-Newton method, 702
Queue discipline, 835, 836
Queueing models, 834–835
Queueing simulation, 1096
Queueing Simulator, 1131
Queueing systems, 834, 835
design and operation of, 1097–1098
design decisions, 909
Queueing theory, 64, 834–890
applications
award-winning, 923–926
conclusions, 926
decision making, 909–911
decision models, 917–923
examples, 907–909
waiting-cost functions, 912–917
basic structure of models
basic queueing process, 835
calling population, 835–836
elementary queueing process, 837–839
input source, 835–836
interarrival time, 836
queue, 836
queue discipline, 836
relationships between *L*, *W*, L_q, and W_q, 840
service mechanism, 837
terminology and notation, 839–840
birth and death process models, 849–866
case, 905–906, 932–934
conclusions on, 889–890
examples of real systems
commercial service, 840–841
internal service, 841
social service, 841
transportation service, 841
hyperexponential distribution, 878–879
models without Poisson input, 876–878
nonexponential distribution models
Erlang distribution and, 873–875
M/D/s model, 872–873
$M/E_k/s$ model, 873–876
Queueing theory—*Cont.*
nonexponential distribution models—*Cont.*
M/G/1 model, 871–872
phase-type distributions, 879
Poisson input process, 846
Poisson process, 846
Pollaczek-Khintchine formula, 872, 873, 875
priority-discipline queueing models, 879–885
product form solution, 886
prototype example, 835
queue discipline, 835
queueing networks, 885–889
equivalence property, 886
infinite queues in series, 886–887
Jackson networks, 887–889
role of exponential distribution, 841–848
service mechanism, 835
and simulation, 1131–1138
simulation example, 1092–1096
time advance methods
fixed-time incrementing, 1093–1095
next-event incrementing, 1095–1096
Queue length, 839
Queues, 834, 836

Radiation therapy, example, 44–46, 137–148
Rand Corporation, 1101
Randomized policies, 1060–1061
Random number integer, 1102
Random numbers
characteristics, 1102–1103
congruential methods for generating
additive method, 1105
mixed congruential method, 1103–1105
multiplicative method, 1105
and cycle length, 1104
generation of, 1101–1105, 1102
variance-reducing techniques, 1129
Random numbers table, 1101
Random observations, generation of
acceptance-rejection method, 1109–1110
chi-square distribution, 1108–1109
Erlang distributions, 1107–1108
exponential distribution, 1107–1108
inverse transformation method, 1106–1107
normal distribution, 1108–1109
simple discrete distribution, 1105–1106
Random variables, 43, 823–825
Rank of a matrix, 1172–1173
Rank of a set of vectors, 1172
Rate in = rate out principle, 850–852
Ratio formula, 1135
Real arcs, 439
Recipient cells, 377
Recurrence time, 818–820
Recurrent state, 811–812
Recurring branching variables, 617
Recursive equation, 1064
Recursive relationship, 540, 549, 561
Reduced costs, 170, 272
Redundant constraint, 59, 624–626
Refinery LP system, 65
Regeneration point, 1132, 1134
Regenerative method
application of formula, 1136–1138
innovative approach, 1131–1135
of statistical analysis, 1131
statistical formula, 1135–1136
Regional planning, example, 46–49
Regression analysis, 1009, 1028–1036
Regression line, 1032
Reject allowance, 1011
Reject allowances problem, 563–565
Relaxation
Lagrangian, 613
LP, 601–603, 606
Reoptimization, 609
in sensitivity analysis, 261
technique, 152–153
Reorder point, 948, 957, 958–960
Replicability, 21
Requirements assumption, 355
Residual capacities, 422
Residual network, 422
Resource allocation, 511–512
Restricted-entry rule, 687–688
Retrospective test, 17
Revenue, 939–940
Revenue management, 963
Reverse arcs, 438, 439
Revised simplex method, 160, 190, 202–212
general observations, 211–212
matrix form of current set of equations, 206–208
overall procedure, 208–211
solving for a basic feasible solution, 204–206
summary of, 208–211
Reynolds Metals Company, 583, 1099
Right-hand sides, simultaneous changes in, 267–278
Rijkswaterstaat, 4, 13, 16, 18, 21
Risk analysis, 1123
Risk-averse individual, 772, 776
Risk-neutral individuals, 772
Risk profile, 1100, 1123, 1126
Risk seekers, 772
RiskSim, 1115
Risk tolerance, 776
Rotatables Allocation and Planning System, 1011

Rounding up, 59
Row rank of a matrix, 1172–1173
Row vector, 1171
(R,Q) inventory policy, 957, 982, 985
Russell's approximation method, 371, 373
  compared to other criteria, 374

SABRE reservation system, 584
Saddle point, 732
Safety stock, 959
Sales force composite forecasting, 1014
Sales forecasting, 1010
Salvage cost, 940
Salvage value, 940, 963
Sample covariance, 1135
Sample size of one, 1016
Sample standard deviation, 1090
Sample variances, 1135
Sampling proportion, 1128
Sampling weight, 1128
San Francisco Police Department, 4, 9, 576
Santa Clara University, 780
Satisficing, 14
Save-It Company, 53–57
Scarce goods, 155
Scenarios, 16
School assignment, case, 307–308
Scientific inventory management, 935–936
Scientist distribution problem, 549–552
SDM system, 65–66
Seasonal effects time series, 1016, 1018–1021
  example, 1019–1020
  general procedure, 1020–1021
  seasonally adjusted, 1020
Seasonal factor, 1019
Seed, 1103
Seervada Park problem, 412–413, 418–420, 424–428, 436
Sensible-odd-bizarre method; *see* SOB method
Sensitive parameters, 15, 156, 244
Sensitivity analysis, 11, 15, 16, 43, 62, 778–779
  applying
    changes in b_i parameter, 262–268
    changes in coefficients of basic variables, 274–278
    changes in coefficients of nonbasic variables, 269–273
    introduction of a new variable, 273–274, 278–279
    simultaneous changes in right-hand sides, 267–278
  Bayes' decision rule, 755–758
  cases
    air pollution control, 302–304
    school assignments, 307–308
Sensitivity analysis—*Cont.*
  and duality theory, 230
  essence of, 254–262
  need for, 255
  parametric programming, 280–284
  role of duality theory
    changes in coefficients of nonbasic variables, 252
    introduction of a new variable, 253–254
    other applications, 254
  in simplex method, 156–159
  summary of procedures, 261–262
  using Excel Solver, 157–159
Sensitivity report, 157
Separable function, 667
Separable programming, 667
  extensions, 696–697
  key property of, 693–694, 696
  reformulation as linear programming problem, 692–696
Sequence of distinct acts, 408
Sequence of interrelated decisions, 539
Sequential-approximation algorithms, 697–698
Sequential procedure for preemptive goal programming, 336–338
Sequential quadratic approximation methods, 702
Sequential unconstrained algorithms, 697
Sequential unconstrained minimization technique, 703–706
Serial correlation, 1105
Servers, 837, 838
  multiple, 853
Service completions, 1093
Service facilities, 837
Service industries, 1012, 1100
Service jobs, 834
Service level, 966–967
Service mechanism, 835, 837
Service times, 837, 838
  probability distribution, 841
Set covering problems, 599–600, 601
Set partitioning problems, 600
Setup cost, 937, 957, 972–975
7-Eleven stores, 65
Shadow prices, 153–156, 219, 683
Shape parameter, 873
Shipment dispatching, 582–583
Shortage cost, 937, 939, 964
Shortages, planned, 943–946
Shortest-path problem, 406, 534n, 601
  algorithm for, 411–412
  applying the algorithm, 412–413
  and minimum cost flow problem, 436
Shortest-path problem—*Cont.*
  other applications, 415
  using Excel, 413–414
Simple discrete distribution, 1105–1106
Simplex method, 2, 25, 30, 109–172, 1062–1063; *see also* Dual simplex method; Network simplex method; Revised simplex method; Transportation simplex method
  adaptation to other model forms, 132–152
  algebra of
    determining direction of movement, 120
    determining where to stop, 120–121
    initialization, 118
    iteration 2 and optimal solution, 122–123
    optimality of, 118–119
    optimality test, 122
    solving for the new BF solution, 121–122
  Big M method, 134–136
  cases
    AmeriBank, 185–187
    fabrics/fall fashions, 182–185
    school assignment, 188–189
  compared to dual simplex method, 309–310
  computer implementation, 160–163
  dual problem in, 238
  economic interpretations, 241–242
  essence of, 109–114
    key solution concept, 112–114
    solving the example, 111–112
  foundations of, 190–202
    adjacent CPF solutions, 193–195
    extensions to the augmented form, 198–202
    properties of CPF solutions, 195–198
    terminology, 190–193
  fundamental insight, 212–220
  geometric and algebraic interpretations, 119
  interior-point algorithm, 163–168
  postoptimality analysis
    parametric linear programming, 159–160
    reoptimization, 152–153
    sensitivity analysis, 156–159
    shadow prices, 153–156
  revised, 160
  setting up, 114–118
  tabular form, 123–128
    initialization, 125
    iteration, 125–126
    optimality test, 125
    resulting optimal solution, 127–128
  tie-breaking in
    entering basic variable, 128–129
    leaving basic variable, 129
    multiple optimal solutions, 130–132
    no leaving variable, 129–130

Simplex tableau, 124, 126, 127, 141
Simulated annealing, 605
Simulation, 1084–1140
applications/examples
coin-flipping game, 1087–1091
construction company problem, 1118–1122
distribution systems, 1099
financial risk analysis, 1099–1100, 1122–1126
health care, 1100
inventory management, 1098, 1116–1118
manufacturing systems, 1099
meeting deadlines, 1098
M/M/1 queueing system, 1092–1096
queueing systems, 1097–1098
service industries, 1100
cases, 1151–1155
conclusions, 1138–1140
cycles, 1132
essence of, 1084–1097
continuous simulation, 1086
discrete event, 1086
role in operations research, 1085–1086
generation of random numbers, 1101–1105
generation of random observations, 1105–1110
length of a cycle, 1132
outline of major study, 1110–1115
checking accuracy, 1112
data collection, 1111–1112
formulation of problem, 1111
model formulation, 1111–1112
plan simulations, 1114
presenting recommendations, 1115
run and analysis, 1114–1115
selection of software, 1112–1113
testing model validity, 1113–1114
queueing simulation, 1096
regeneration point, 1132
regenerative method of statistical analysis, 1131–1138
on spreadsheets, 1115–1126
technique and uses, 1084
time advance methods
fixed-time incrementing, 1093–1095
next-event incrementing, 1095–1096
variance-reducing techniques, 1126–1130
Simulation clock, 1091–1092
Simulation model, 1085
Simultaneous solution, 192
Simultaneous solution of constraint boundary equations, 195
Simultaneous tolerance levels, 1035–1036
Single-period probabilistic model, 962n
Single-period stochastic model, 962n
Single-server case, 854–856, 861–862, 865
Single-server queueing system, 1133–1134
Singular matrix, 1173
Sink, 421
Site selection problem, 581
Skewed distribution, 1090
Slack
for an activity, 484
and critical path, 484–485
in functional constraints, 559
identifying, 482–485
zero, 484
Slack variables, 114–115, 198–199, 212
Slope-intercept form, 29
Slope of the profit function, 38
Smoothing constant, 1017
SOB method, 249–252
Social service, 841
Solid waste disposal, example, 53–57
Solid waste management, 430
Solutions of the model, 33–36
Solution tree, 606, 608
Solved nodes, 411–412
Solvers, 78
Sources, 354, 421, 430
South African National Defense Force, 4, 581
Southern Confederation of Kibbutzim, 46–49
Southern Electric, 780
Southland Corporation, 65, 67
Soviet Union, 347
Spanning tree, 410, 434
feasible, 439–440
Spanning tree solution, 440
Spare parts needs, 1010–1011
Special restriction, 692–693
Spreadsheets, 15
for linear programming, 67–72
with LINGO, 86–87
for simulation, 1112–1113, 1115–1126
software, 161–163
transportation problems, 358–365
Square matrix, 1171
(s,S) inventory policy, 973–974
Stable products, 961, 975–983
Stable solution, 732
Staffing needs forecasting, 1012–1013
Stages, 539
Stakeholders, 9
Start-time constraints, 499
Startup costs, 51
State of nature, 751
States, 539, 802
State system, 839
State transition formula, 1092
State vector, 560
Stationary policy, 1057, 1059
Stationary probabilities, 813
Stationary transition probabilities, 803–804, 823
Statistical analysis, 1131–1138
Statistical forecasting methods, 1009, 1037
applications, 1013
Statistical formula, 1135–1136
Steady-state condition, 839, 849
Steady-state equation, 813–814, 825–826
Steady-state probabilities, 825–826, 1055
Steady-state properties, 812–813
Stochastic inventory models, 936
continuous review, 956–961
choosing order quantity Q, 958
choosing reorder point R, 958–960
example, 961
model assumptions, 957–958
service level measures, 958–959
nonlinear costs, 975
periodic review, 975–983
multiperiod models, 980–982
two-period model with no setup cost, 976–980
single-period for perishable products, 961–975
analysis, 965–967
application, 967–968
assumptions, 965
example, 963–965
exponential demand distribution, 974–975
with initial stock level, 968–972
nonlinear costs, 971–972
with no setup cost, 1116
with setup cost, 972–975
types of products, 962–963
Stochastic processes, 802
Stockholders, 9
Stockout, 943
Storage cost, 939
Strategies
dominated, 729–731
game theory, 726–728
minimax criterion, 732, 734–735
mixed, 733–735
optimal mixed, 737
and payoff table, 726–727, 728–729
pure, 733
stable solution, 732
Stratified sampling, 1127–1129
Streamlined algorithms, 350
Streamlined procedure for preemptive goal programming, 338–339

Strictly concave function
of several variables, 1162–1163
of single variable, 1159–1161
Strictly convex function, 975
of several variables, 1162–1163
of single variable, 1159–1161
Strong duality property, 236, 238, 683, 740
Structural constraints, 33
Submatrices, 1171
Sum of squares, 1031
Superoptimal basic solution, 254
Superoptimal solution, 261
Suppliers, 9
Supply, 354
Supply, distribution, and marketing planning, case, 65–67
Supply chain, 985, 1098
Supply chain management, 985–986
Supply node, 410
Surplus stock, 1004–1008
Surplus variables, 139, 234
Symmetry property, 237, 248
SYSNET, 19
Systems Optimization Laboratory, Stanford University, 706

Table lookup approach, 1107
Tables, 1174–1175
Tabular form of simplex method, 123–128
Tabu research, 604
Taco Bell, 4
Tasks, 381
Team approach, 3, 8
Telecommunications transmission systems, 834
Testing, 12
Texaco, 4, 20
Texas Stadium, 583
Tie breaking, 128–132
Tightening constraints, 626–622
Time advance methods
fixed-time incrementing, 1093–1095, 1132
next-event incrementing, 1095–1096, 1132
Time-cost graph, 493
Time-cost trade-offs, 491, 492–502
conclusions on, 501–502
crashing activities, 494–496
for individual activities, 493–494
linear programming for, 496–501
Time series, 1009, 1014–1016
Box-Jenkins model, 1026–1028
case, 1048–1052
constant-level model, 1016–1018
exponential smoothing for linear trend, 1021–1025
Time series—*Cont.*
more than one time period ahead, 1025
seasonal effects, 1018–1021
Time value of money, 1069–1070
Tomco Oil Company, 780
Tornado diagrams, 779
Torricelli Act, 347
Total function value, 42
Total variable cost, 948
Toyota Motor Company, 950
Transient condition, 839, 849
Transient state, 811–812
Transition intensities, 824
Transition matrix, 804–805, 809–810, 812
Transition probabilities, 803–804, 823, 1053, 1056
Transition rates, 824–825, 826–828
Transportation problem, 350–381, 601
award-winning application, 354
case, 401–402
jet engine production, 359–362
and minimum cost flow problem, 435
model, 354–357
network representation, 351–354
in nonlinear programming, 656–658
prototype example, 351–364
terminology for, 354–355
using Excel, 358–365
water supply problem, 362–365, 370–373
Transportation service, 841
Transportation simplex method, 350–351, 365
initialization, 368–374
northwest corner rule, 369–370
Russell's approximation, 371, 373
Vogel's approximation, 370–371
iteration for, 376–379
optimality test, 375–376
setting up, 365–367
summary, 379–381
Transportation simplex tableau, 367–368, 374, 376, 378
Transpose operation, 1170
Transshipment nodes, 410, 421
Transshipment problem, 391, 435–436
Tree, 410
TreePlan, 769–770, 776
Trend factor, 1022
Trend smoothing constant, 1023
Triangular distribution, 1120–1121
Turkish Petroleum Refineries Corporation, 581
Two-bin inventory system, 956
Two-period stochastic inventory model, 976–980
Two-person, zero-sum game, 742
and decision analysis, 751
Two-person, zero-sum game—*Cont.*
formulation of, 726–729
simple game formulation, 728–729
Two-phase method, 142–148, 686–687
Two-segment piecewise linear functions, 697
Two-sided goal, 333
Two-variable problem, 675, 676–677

Unbounded Z, 35, 129–130
Uncapacitated minimum cost flow problems, 439
Uncertainty
choice in face of, 751
and decision analysis, 749
Unconditional state probabilities, 810
Unconstrained optimization, 665
and convex programming, 667
multivariable, 673–679
one-variable, 670–673
several variables, 1166
single variable, 1165–1166
Undirected arc, 408, 413, 422–423
Undirected network, 408
Undirected path, 409
Unemployment insurance econometric forecasting model, 1012
Uniform random number, 1092, 1102
Union Airways, 57–59
United Airlines, 4, 63–65, 67, 924, 1012
United States Department of Defense, 780
United States Postal Service, 780, 1100
Unit holding cost, 963
Unit production cost, 937
Unsatisfied demand, 964
Unsatisfied demand cost, 939
Unsolved nodes, 411–412
Unstable solution, 733
Unused demand capacity, 364
Upper, one-sided goal, 333
Upper bound, 196–197
Upper bound constraints, 60–61, 317–318
Upper bound technique, 78, 309
in linear programming, 317–320
in network simplex method, 438–439
User acceptance test, 21, 985
User team, 18, 21
Utility functions for money, 771–773
Utility theory
applying, 773–775
approach for estimating payoff, 775–776
exponential utility function, 776
fundamental property, 772
using decision analysis, 776–778
utility functions for money, 771–773
Utilization factor, 839

Value
of experimentation, 762–763
of perfect information, 762–764
Value determination, 1065
Value of the game, 731
Variable metric method, 702
Variance, 658
Variance-reducing techniques, 1126–1130
complementary random numbers, 1129
conclusions on, 1130
Monte Carlo techniques, 1126–1127
stratified sampling, 1127–1129
Vector of basic variables, 204
Vectors, 1171–1172
Vehicles, 834
Vogel's approximation method, 370, 371
compared to other criteria, 373–374
Volume discounts, 656

Waiting-cost functions, 912–917
g(N) form, 912–914
h(°W) form, 914–915
linear, 913
Waiting-in-line situation; *see* Queueing theory
Waiting time in the queue, 855
Warm-up period, 1095
Wasted iteration, 386
Water resource distribution, 362–365, 370–373
Weak duality property, 236, 238
Weighted average, 130
Weighted set covering problem, 599n
What-if questions, 20, 62–63
Winter Simulation Conference, 1101
World Bank, 584
Worldwide Inventory Network Optimizer, 986
Worst-case performance, 165–166
Worst case scenario, 487
Wyndor Glass Company problem, 25–44, 68, 109–112, 118, 119, 124–135, 150–151, 156–158, 164–165, 200–202, 205–206, 256–261, 262–284, 559–562, 659–664, 694–696

Xerox Corporation, 923–924

Yellow Freight System, Inc., 4, 19, 21
Yes-or-no decisions, 579–585, 586–587, 589

Zero slack, 484
Zero-sum games; *see* Two-person, zero-sum game