7

LINEAR PROGRAMMING (LP) AND APPLICATIONS

LINEAR PROGRAMMING (LP) IS one of the most widely used optimization techniques and perhaps the most effective. The term *linear programming* was coined by George Dantzig in 1947 to refer to problems in which both the objective function and the constraints are linear (Dantzig, 1998; Martin, 1999; Vanderbei, 1999). The word *programming* does not refer to computer programming, but means optimization. This is also true in the phrases "nonlinear programming," "integer programming," and so on. The following are examples of LP that occur in plant management:

1. Assign employees to schedules so that the workforce is adequate each day of the week and worker satisfaction and productivity are as high as possible.
2. Select products to manufacture in the upcoming period, taking best advantage of existing resources and current prices to yield maximum profit.
3. Find a pattern of distribution from plants to warehouses that will minimize costs within the capacity limitations.
4. Submit bids on procurement contracts to take into account profit, competitors' bids, and operating constraints.

When stated mathematically, each of these problems potentially involves many variables, many equations, and many inequalities. A solution must not only satisfy all of the constraints, but also must achieve an extremum of the objective function, such as maximizing profit or minimizing cost. With the aid of modern software you can formulate and solve LP problems with many thousands of variables and constraints.

7.1 GEOMETRY OF LINEAR PROGRAMS

Consider the problem

$$\text{Maximize: } f = x_1 + 3x_2$$

$$\begin{aligned} \text{Subject to:} \quad -x_1 + x_2 &\le 1 \\ x_1 + x_2 &\le 2 \\ x_1 \ge 0, \quad x_2 &\ge 0 \end{aligned} \tag{7.1}$$

The feasible region lies within the unshaded area of Figure 7.1 defined by the intersections of the half spaces satisfying the linear inequalities. The numbered points are called extreme points, corner points, or vertices of this set. If the constraints are linear, only a finite number of vertices exist.

Contours of constant value of the objective function f are defined by the linear equation

$$x_1 + 3x_2 = \text{Constant} = c \tag{7.2}$$

As c varies, the contour is moved parallel to itself. The maximum value of f is the largest c for which the line has at least one point in common with the constraint set.

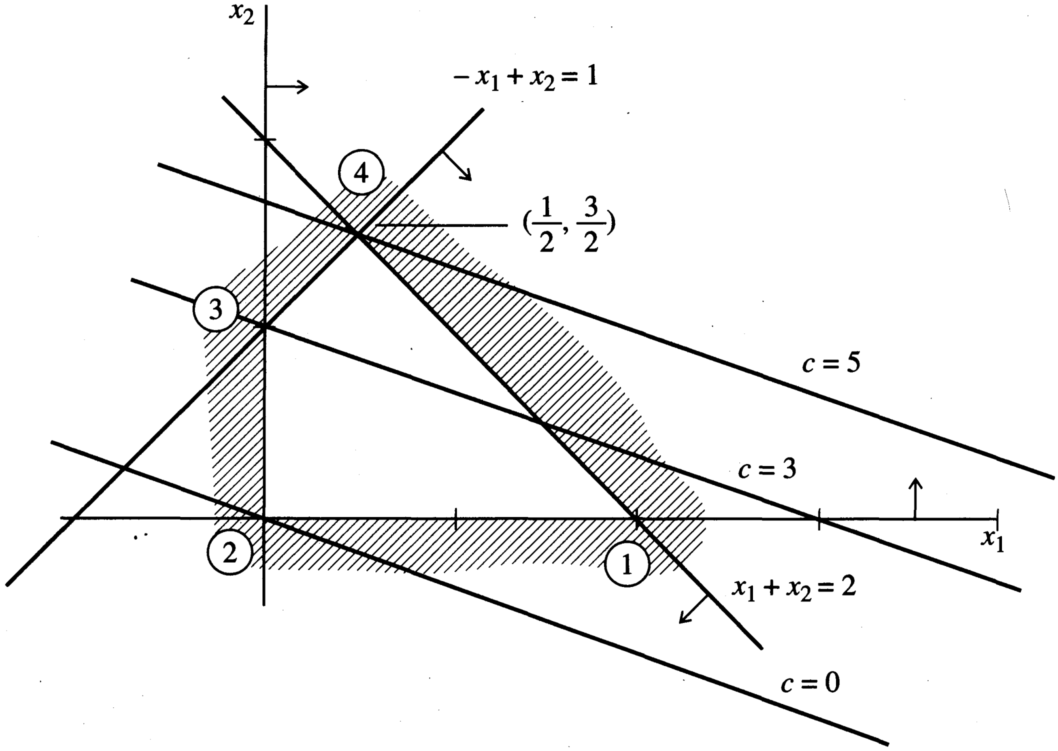

FIGURE 7.1
Geometry of a linear program.

For Figure 7.1, this point occurs for $c = 5$, and the optimal values of $\mathbf{x}$ are $x_1 = 0.5$, $x_2 = 1.5$. Note that the maximum value occurs at a vertex of the constraint set. If the problem seeks to minimize f, the minimum is at the origin, which is again a vertex. If the objective function were $f = 2x_1 + 2x_2$, the line $f =$ Constant would be parallel to one of the constraint boundaries, $x_1 + x_2 = 2$. In this case the maximum occurs at two extreme points, ($x_1 = 0.5, x_2 = 1.5$) and ($x_1 = 2, x_2 = 0$) and, in fact, also occurs at all points on the line segment joining these vertices.

Two additional cases can exist. First, if the constraint $x_1 + x_2 \leq 2$ had been removed, the feasible region would appear as in Figure 7.2, that is, the set would be unbounded. Then max f is also unbounded because f can be made as large as desired subject to the constraints. Second, at the opposite extreme, the constraint set could be empty, as in the case where $x_1 + x_2 \leq 2$ is replaced by $x_1 + x_2 \leq -1$. Thus an LP problem may have (1) no solution, (2) an unbounded solution, (3) a single optimal solution, or (4) an infinite number of optimal solutions. The methods to be developed deal with all these possibilities.

The fact that the extremum of a linear program always occurs at a vertex of the feasible region is the single most important property of linear programs. It is true for any number of variables (i.e., more than two dimensions) and forms the basis for the simplex method for solving linear programs (not to be confused with the simplex method discussed in Section 6.1.4).

Of course, for many variables the geometrical ideas used here cannot be visualized, and therefore the extreme points must be characterized algebraically. This is

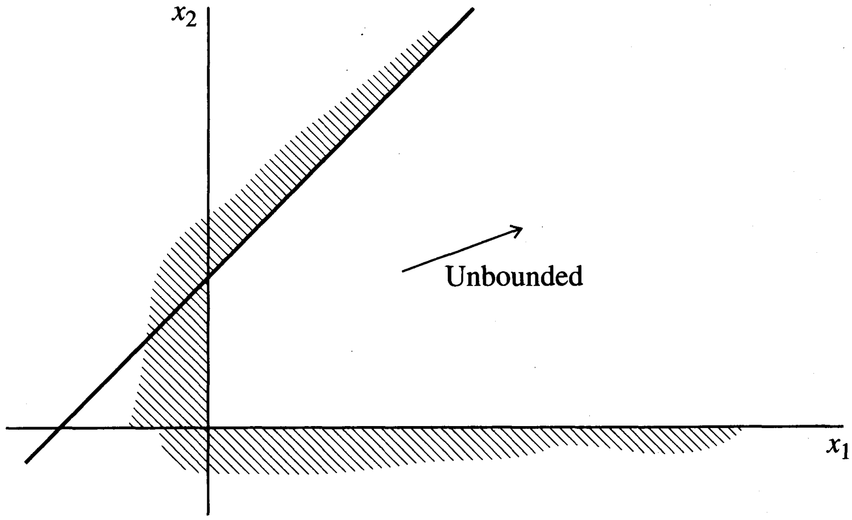

FIGURE 7.2
Unbounded minimum.

done in the next two sections, in which the problem is placed in standard form and the basic results of linear programming are stated.

Standard form for linear programs

An LP problem can always be written in the following form. Choose $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ to minimize

$$f = \sum_{j=1}^{n} c_j x_j \tag{7.3}$$

$$\text{Subject to:} \quad \sum_{j=1}^{n} a_{ij} x_j = b_i, \quad i = 1, 2, \ldots, m \tag{7.4}$$

$$l_j \leq x_j \leq u_j, \quad j = 1, \ldots, n \tag{7.5}$$

where c_j are the n objective function coefficients, a_{ij} and b_i are parameters in the m linear equality constraints, and l_j and u_j are lower and upper bounds with $l_j \leq u_j$. Both l_j and u_j may be positive or negative. In matrix form, this problem is

$$\text{Minimize: } f = \mathbf{cx} \tag{7.6}$$

$$\text{Subject to: } \mathbf{Ax} = \mathbf{b} \tag{7.7}$$

$$\text{and } \mathbf{l} \leq \mathbf{x} \leq \mathbf{u} \tag{7.8}$$

$\mathbf{A}$ is an $m \times n$ matrix whose (i, j) element is the constraint coefficient a_{ij}, and $\mathbf{c}$, $\mathbf{b}$, $\mathbf{l}$, $\mathbf{u}$ are vectors whose components are c_j, b_j, l_j, u_j, respectively. If any of the Equations (7.7) were redundant, that is, linear combinations of the others, they could be deleted without changing any solutions of the system. If there is no solution, or if there is only one solution for Equation (7.7), there can be no optimization. Thus the case of greatest interest is where the system of equations (7.7) has more unknowns than equations and has at least two and potentially an infinite number of solutions. This occurs if and only if

$$n > m$$

and

$$\text{Rank}(\mathbf{A}) = m$$

We assume these conditions are true in what follows. The problem of linear programming is to first detect whether solutions exist, and, if so, to find one yielding the minimum f.

Note that all the constraints in Equation (7.4) are equalities. It is necessary to place the problem in this form to solve it most easily (equations are easier to work with here than inequalities). If the original system is not of this form, it may easily be transformed by use of so-called *slack variables.* If a given constraint is an inequality, for example,

$$\sum_{j=1}^{n} a_{ij}x_j \le b_i$$

then define a slack variable $x_{n+i} \ge 0$ such that

$$\sum_{j=1}^{n} a_{ij}x_j + x_{n+i} = b_i$$

and the inequality becomes an equality. Similarly, if the inequality is

$$\sum_{j=1}^{n} a_{ij}x_j \ge b_i$$

we write

$$\sum_{j=1}^{n} a_{ij}x_j - x_{n+i} = b_i$$

Note that the slacks must be nonnegative to guarantee that the inequalities are satisfied.

EXAMPLE 7.1 STANDARD LP FORM

Transform the following linear program into standard form:

$$\text{Minimize:} \quad f = x_1 + x_2$$

$$\begin{aligned} \text{Subject to:} \quad 2x_1 + 3x_2 &\le 6 \\ x_1 + 7x_2 &\ge 4 \\ x_1 + x_2 &= 3 \\ x_1 \ge 0, \quad x_2 &\text{ unconstrained in sign} \end{aligned}$$

Solution. Define slack variables $x_3 \geq 0, x_4 \geq 0$. Then the problem becomes

$$\begin{aligned} &\text{Minimize: } f = x_1 + x_2 \\ &\text{Subject to: } \quad 2x_1 + 3x_2 + x_3 = 6 \\ &\qquad\qquad\quad x_1 + 7x_2 - x_4 = 4 \\ &\qquad\qquad\quad x_1 + x_2 = 3 \\ &x_1 \geq 0, \quad x_3 \geq 0, \quad x_4 \geq 0 \end{aligned}$$

In the rest of this chapter, we assume that the rows of the constraint matrix **A** are linearly independent, that is, rank (**A**) = m. If a slack variable is inserted in every row, then **A** contains a submatrix that is the identity matrix. In the preceding example, if we insert a slack variable x_5 into the equality:

$$x_1 + x_2 + x_5 = 3$$

$$0 \leq x_5 \leq 0 \quad (\text{i.e.}, x_5 = 0)$$

then the rows of **A** are independent. Modern LP solvers automatically transform problems in this way.

7.2 BASIC LINEAR PROGRAMMING DEFINITIONS AND RESULTS

We now generalize the ideas illustrated earlier from 2 to n dimensions. Proofs of the following theorems may be found in Dantzig (1963). First a number of standard definitions are given.

DEFINITION 1. A *feasible solution* to the linear programming problem is a vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ that satisfies Equations (7.7) and the bounds (7.8).

DEFINITION 2. A *basis matrix* is an $m \times m$ nonsingular matrix formed from some m columns of the constraint matrix **A** (*Note*: Because rank (**A**) = m, **A** contains at least one basis matrix).

DEFINITION 3. A *basic solution* to a linear program is the unique vector determined by choosing a basis matrix, setting each of the $n - m$ variables associated with columns of **A** not in the basis matrix equal to either l_j or u_j, and solving the resulting square, nonsingular system of equations for the remaining m variables.

DEFINITION 4. A *basic feasible solution* is a basic solution in which all variables satisfy their bounds (7.8).

DEFINITION 5. A *nondegenerate basic feasible solution* is a basic feasible solution in which all basic variables x_j are strictly between their bounds, that is, $l_j < x_j < u_j$.

DEFINITION 6. An *optimal solution* is a feasible solution that also minimizes f in Equation (7.6).

For example, in the system

$$
\begin{aligned}
-x_1 + x_2 + x_3 \quad\quad &= 1 \\
x_1 + x_2 \quad\quad + x_4 &= 2 \\
x_i \geq 0, \quad i = 1, \ldots, 4&
\end{aligned}
\tag{7.9}
$$

obtained from Equation (7.1) by adding slack variables x_3 and x_4, the matrix

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

formed from columns 3 and 4 of the equations in (7.9) is nonsingular and hence is a basis matrix. The corresponding basic solution of (7.9)

$$x_1 = 0, \quad x_2 = 0, \quad x_3 = 1, \quad x_4 = 1$$

is a nondegenerate basic feasible solution. The matrix

$$\mathbf{B}_1 = \begin{bmatrix} -1 & 0 \\ 1 & 1 \end{bmatrix}$$

formed from columns 1 and 4 of Equation (7.9) is also a basis matrix. The corresponding basic solution is obtained by setting $x_2 = x_3 = 0$ and solving

$$
\begin{aligned}
-x_1 \quad\quad &= 1 \\
x_1 + x_4 &= 2
\end{aligned}
$$

yielding $x_1 = -1, x_4 = 3$. This basic solution is not feasible.

The importance of these definitions is brought out by the following results:

RESULT 1. The objective function f assumes its minimum at a vertex of the feasible region. If it assumes its minimum at more than one vertex, then it takes on the same value at every point of the line segment joining any two optimal vertices.

This theorem is a multidimensional generalization of the geometric arguments given previously. By result 1, in searching for a solution, we need only look at vertices. It is thus of interest to know how to characterize vertices in many dimensions algebraically. This information is given by the next result.

RESULT 2. A vector $\mathbf{x} = (x_1, \ldots, x_n)$ is a vertex of the constraint set of an LP problem if and only if $\mathbf{x}$ is a basic feasible solution of the constraints (7.7)–(7.8).

Result 2 is true in two dimensions as can be seen from the example of relations (7.1), whose constraints have been rewritten in equation form in (7.9). The (x_1, x_2) coordinates of the vertex at $x_1 = 0, x_2 = 1$ are given by the (x_1, x_2) coordinates of the basic feasible solution

$$x_1 = 0, \quad x_2 = 1, \quad x_3 = 0, \quad x_4 = 1$$

The optimal vertex corresponds to the basic feasible solution

$$x_1 = 0.5, \quad x_2 = 1.5, \quad x_3 = x_4 = 0$$

An alternative definition of a vertex provides geometric insight and generalizes easily to nonlinear problems. Refer again to Figure 7.1. There are two variables, and each vertex is at the intersection of two *active constraints.* If there were three variables, active constraints would correspond to planes, and vertices would be determined by the intersection of at least three active constraints. For n variables, at least n hyperplanes must interact to define a point. We say "at least," because it is possible that more than n hyperplanes pass through a vertex. One can always draw other redundant constraints through the vertices in Figure 7.1.

We can state these ideas precisely as follows. Consider any optimization problem with n variables, let x be any feasible point, and let $n_{\text{act}}(\mathbf{x})$ be the number of active constraints at $\mathbf{x}$. Recall that a constraint is active at $\mathbf{x}$ if it holds as an equality there. Hence equality constraints are active at any feasible point, but an inequality constraint may be active or inactive. Remember to include simple upper or lower bounds on the variables when counting active constraints. We define the *number of degrees of freedom* (dof) at $\mathbf{x}$ as

$$\text{dof}(\mathbf{x}) = n - n_{\text{act}}(\mathbf{x})$$

DEFINITION: A feasible point $\mathbf{x}$ is called a *vertex* if $\text{dof}(\mathbf{x}) \leq 0$ and the coefficient matrix of the active constraints at $\mathbf{x}$ has rank n. It is a *nondegenerate* vertex if $\text{dof}(\mathbf{x}) = 0$, and a degenerate vertex if $\text{dof}(\mathbf{x}) < 0$, in which case abs[dof($\mathbf{x}$)] is called the *degree of degeneracy* at $\mathbf{x}$.

Comparing this definition with the previous one ($\mathbf{x}$ is a vertex if and only if it is a basic feasible solution), if $\mathbf{x}$ is a basic feasible solution, then $n - m$ nonbasic bounds are active, plus m equalities, so

$$n_{\text{act}}(\mathbf{x}) \geq n - m + m = n$$

and $\text{dof}(\mathbf{x}) \leq 0$. If k basic variables are at their bounds, $n_{\text{act}}(\mathbf{x}) = n + k$, and $\mathbf{x}$ is a degenerate vertex with degree of degeneracy k. It is straightforward to show that the active constraint matrix has rank n. One can reverse the argument, showing the definitions are equivalent.

In nonlinear programming problems, optimal solutions need not occur at vertices and can occur at points with positive degrees of freedom. It is possible to have no active constraints at a solution, for example in unconstrained problems. We consider nonlinear problems with constraints in Chapter 8.

Results 1 and 2 imply that, in searching for an optimal solution, we need only consider vertices, hence only basic feasible solutions. Because a basic feasible solution has m basic variables, an upper bound to the number of basic feasible solutions is the number of ways m variables can be selected from a group of n variables, which is

$$\binom{n}{m} = \frac{n!}{(n-m)!\,m!}$$

For large n and m this is a very large number. Thus, for large problems, it is impossible to evaluate f at all vertices to find the minimum. What is needed is a computational scheme that selects, in an orderly fashion, a sequence of vertices, each one yielding a lower value of f, until finally the minimum is attained. In this way we consider only a small subset of the vertices. The simplex method, devised by G. B. Dantzig, is such a scheme. This procedure finds a vertex and determines whether it is optimal. If not, it finds a neighboring vertex at which the value of f is less than or equal to the previous value. The process is iterated and in a finite number of steps (usually between m and $2m$) the minimum is found. The simplex method also discovers whether the problem has no finite minimal solution (i.e., $\min f = -\infty$) or if it has no feasible solutions (i.e., an empty constraint set). It is a powerful scheme for solving any linear programming problem.

To explain the method, it is necessary to know how to go from one basic feasible solution (BFS) to another, how to identify an optimal BFS, and how to find a better BFS from a BFS that is not optimal. We consider these questions in the following two sections. The notation and approach used is that of Dantzig (1998).

Systems of linear equations and equivalent systems

Consider the system of m linear equations in n unknowns

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ \vdots \qquad\qquad\qquad\qquad & \quad \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m \end{aligned} \tag{7.10}$$

A solution to this system is any set of variables $x_1 \ldots x_n$ that simultaneously satisfies all equations. The set of all solutions to the system is called its solution set. The system may have one, many, or no solutions. If there is no solution, the equations are said to be inconsistent, and their solution set is empty.

Equivalent systems and elementary operations

Two systems of equations are said to be equivalent if they have the same solution sets. Dantzig (1998) proved that the following operations transform a given linear system into an equivalent system:

1. Multiplying any equation E_i by a constant $q \neq 0$
2. Replacing any equation E_t by the equation $E_t + qE_i$, where E_i is any other equation of the system

These operations are called elementary row operations. For example, the linear system of Equations (7.9)

$$\begin{aligned} -x_1 + x_2 + x_3 \qquad &= 1 \\ x_1 + x_2 \qquad + x_4 &= 2 \end{aligned}$$

may be transformed into an equivalent system by multiplying the first equation by -1 and adding it to the second, yielding

$$-x_1 + x_2 + x_3 \qquad = 1$$

$$2x_1 \qquad - x_3 + x_4 = 1$$

Note that the solution $x_1 = 0, x_3 = 0, x_2 = 1, x_4 = 2$ is a solution of both systems. In fact, any solution of one system is a solution of the other.

Pivoting

A particular sequence of elementary row operations finds special application in linear programming. This sequence is called a *pivot operation*, defined as follows.

DEFINITION. A pivot operation consists of m elementary operations that replace a linear system by an equivalent system in which a specified variable has a coefficient of unity in one equation and zero elsewhere. The detailed steps are as follows:

1. Select a term $a_{rs}x_s$, in row (equation) r, column (variable) s, with $a_{rs} \neq 0$ called the pivot term.
2. Replace the rth equation E_r by the rth equation multiplied by $1/a_{rs}$.
3. For each $i = 1, 2, \ldots, m$ except $i = r$, replace the ith equation E_i by $E_i - a_{is}/a_{rs}E_r$, that is, by the sum of E_i and the replaced rth equation multiplied by $-a_{is}$.

EXAMPLE 7.2 USE OF PIVOT OPERATIONS

Consider the system

$$2x_1 + 3x_2 - 4x_3 + x_4 = 1 \qquad (a)$$

$$x_1 - x_2 \qquad + 5x_4 = 6 \qquad (b)$$

$$3x_1 + x_2 + x_3 \qquad = 2 \qquad (c)$$

Transform the set of equations to an equivalent system in which x_1 is eliminated from all but Equation *(a)*, but having a unity coefficient in Equation *(a)*.

Solution. Choose the term $2x_1$ as the pivot term. The first operation is to make the coefficient of this term unity, so we divide Equation *(a)* by 2, yielding the equivalent system

$$x_1 + 1.5x_2 - 2x_3 + 0.5x_4 = 0.5 \qquad (a')$$

$$x_1 - x_2 \qquad + 5x_4 = 6 \qquad (b)$$

$$3x_1 + x_2 + x_3 \qquad = 2 \qquad (c)$$

The next operation eliminates x_1 from Equation *(b)* by multiplying *(a')* by -1 and adding the result to Equation *(b)*, yielding

$$x_1 + 1.5x_2 - 2x_3 + 0.5x_4 = 0.5 \qquad (a')$$

$$-2.5x_2 + 2x_3 + 4.5x_4 = 5.5 \qquad (b')$$

$$3x_1 + x_2 + x_3 = 2 \qquad (c)$$

Finally, we eliminate x_1 from Equation *(c)* by multiplying *(a')* by -3 and adding the result to Equation *(c)*, yielding

$$x_1 + 1.5x_2 - 2x_3 + 0.5x_4 = 0.5 \qquad (a')$$

$$-2.5x_2 + 2x_3 + 4.5x_4 = 5.5 \qquad (b')$$

$$3.5x_2 + 7x_3 - 1.5x_4 = 0.5 \qquad (c')$$

Canonical systems

In the following discussion we assume that, in the system of Equations (7.6)–(7.8), all lower bounds $l_j = 0$, and all upper bounds $u_j = +\infty$, that is, that the bounds become $x \geq 0$. This simplifies the exposition. The simplex method is readily extended to general bounds [see Dantzig (1998)]. Assume that the first m columns of the linear system (7.7) form a basis matrix **B.** Multiplying each column of (7.7) by $\mathbf{B}^{-1}$ yields a transformed (but equivalent) system in which the coefficients of the variables $(x_1, \ldots, x_m)$ are an identity matrix. Such a system is called *canonical* and has the form shown in Table 7.1.

The variables $x_1, \ldots, x_m$ are associated with the columns of **B** and are called basic variables. They are also called dependent, because if values are assigned to the nonbasic, or independent variables, $x_{m+1}, \ldots, x_n$, then $x_1, \ldots, x_m$ can be determined immediately. In particular, if $x_{m+1}, \ldots, x_n$ are all assigned zero values then we obtain the basic solution

$$x_1 = \bar{b}_1, x_2 = \bar{b}_2, \ldots, x_m = \bar{b}_m, x_{m+1} = x_{m+2} = \cdots = x_n = 0$$

TABLE 7.1
Canonical system with basic variables $x_1, x_2, \ldots, x_m$

Dependent (basic) variables	Independent (nonbasic) variables	Constants
x_1	$+\bar{a}_{1,m+1}x_{m+} + \bar{a}_{1,m+2}x_{m+2} + \cdots + \bar{a}_{1n}x_n$	$= \bar{b}_1$
x_2	$+\bar{a}_{2,m+1}x_{m+1} + \bar{a}_{2,m+2}x_{m+2} + \cdots + \bar{a}_{2n}x_n$	$= \bar{b}_2$
$\ddots$	$\vdots \qquad \vdots$	$\vdots$
x_m	$+\bar{a}_{m,m+1}x_{m+1} + \bar{a}_{m,m+2}x_{m+2} + \cdots + \bar{a}_{mn}x_n$	$= \bar{b}_m$

If

$$\bar{b}_i \geq 0, \quad i = 1, \ldots, m$$

then this is a basic feasible solution. If one or more $\bar{b}_i = 0$, the basic feasible solution is degenerate.

Instead of actually computing $\mathbf{B}^{-1}$ and multiplying the linear system (7.7) by it, we can place Equation (7.7) in canonical form by a sequence of m pivot operations. First pivot on the term $a_{11}x_1$ if $a_{11} \neq 0$. If $a_{11} = 0$, there exists an element in its first row that is nonzero, since $\mathbf{B}$ is nonsingular. Rearranging the columns makes this the (1, 1) element and allows the pivot. Repeating this procedure for the terms $a_{22}x_2, \ldots, a_{mm}x_m$ generates the canonical form. Such a form will be used to begin the simplex algorithm.

7.3 SIMPLEX ALGORITHM

The simplex method is a two-phase procedure for finding an optimal solution to LP problems. Phase 1 finds an initial basic feasible solution if one exists or gives the information that one does not exist (in which case the constraints are inconsistent and the problem has no solution). Phase 2 uses this solution as a starting point and either (1) finds a minimizing solution or (2) yields the information that the minimum is unbounded (i.e., $-\infty$). Both phases use the simplex algorithm described here.

In initiating the simplex algorithm, we treat the objective function

$$f = c_1x_1 + c_2x_2 + \cdots + c_nx_n$$

as just another equation, that is,

$$-f + c_1x_1 + c_2x_2 + \cdots + c_nx_n = 0 \tag{7.11}$$

which we include in the set to form an augmented system of equations. The simplex algorithm is always initiated with this augmented system in canonical form. The basic variables are some m of the x's, which we renumber to make the first m, that is, $x_1 \ldots x_m$ and $-f$. The problem can then be stated as follows.

Find values of $x_1 \geq 0, x_2 \geq 0, \ldots, x_n \geq 0$ and min f satisfying

$$\begin{array}{llllllll}
x_1 & & & + \bar{a}_{1,m+1}x_{m+1} & + \cdots + & \bar{a}_{1n}x_n & = & \bar{b}_1 \\
& x_2 & & & & & & \\
& & \ddots & \vdots & & \vdots & & \vdots \\
& & x_m & + \bar{a}_{m,m+1}x_{m+1} & + \cdots + & \bar{a}_{mn}x_n & = & \bar{b}_m \\
& & (-f) & + \bar{c}_{m+1}x_{m+1} & + \quad\; + & \bar{c}_nx_n & = & -\bar{f}
\end{array} \tag{7.12}$$

In this canonical form the basic solution is

$$f = \bar{f}, \bar{x}_1 = \bar{b}_1, \ldots, \bar{x}_m = \bar{b}_m, x_{m+1} = x_{m+2} = \cdots = x_n = 0 \qquad (7.13)$$

We assume that this basic solution is feasible, that is,

$$\bar{b}_1 \geq 0, \bar{b}_2 \geq 0, \ldots, \bar{b}_m \geq 0 \qquad (7.14)$$

The workings of phases 1 and 2 guarantee that this assumption is always satisfied. If Equation (7.14) holds, we say that the linear programming problem is in feasible canonical form.

Test for optimality

If the problem is in feasible canonical form, we have a vertex directly at hand, represented by the basic feasible solution (7.13). But the form provides even more valuable information. By merely glancing at the numbers $\bar{c}_j, j = m + 1, \ldots, n$, you can tell if this extreme point is optimal and, if not, you can move to a better one. Consider first the optimality test, given by the following result.

RESULT 3. A basic feasible solution is a minimal feasible solution with total cost $\bar{z}$ if all constants $\bar{c}_{m+1}, \bar{c}_{m+2}, \ldots, \bar{c}_n$ are nonnegative, that is, if

$$\bar{c}_j \geq 0 \quad j = m + 1, \ldots, n \qquad (7.15)$$

The $\bar{c}_j$ are called *reduced costs*.

The proof of this result involves writing the previous equation as

$$f = \bar{f} + \bar{c}_{m+1}x_{m+1} + \ldots + \bar{c}_n x_n$$

Because the variables $x_{m+1} \ldots x_n$ are presently zero and are constrained to be nonnegative, the only way any one of them can change is for it to become positive. But if $\bar{c}_j \geq 0$ for $j = m + 1, \ldots, n$, then increasing any x_j cannot decrease the objective function f because then $\bar{c}_j x_j \geq 0$. Because no feasible change in the nonbasic variables can cause f to decrease, the present solution must be optimal.

The reduced costs also indicate if there are multiple optima. Let all $\bar{c}_j \geq 0$ and let $\bar{c}_k = 0$ for some nonbasic variable x_k. Then, if the constraints allow that variable to be made positive, no change in f results, and there are multiple optima. It is possible, however, that the variable may not be allowed by the constraints to become positive; this may occur in the case of degenerate solutions. We consider the effects of degeneracy later. A corollary to these results is the following:

RESULT 4. A basic feasible solution is the unique minimal feasible solution if $\bar{c}_j > 0$ for all nonbasic variables.

Of course, if some $\bar{c}_j < 0$ then f can be decreased by increasing the corresponding x_j, so the present solution is probably nonoptimal. Thus we must consider means of improving a nonoptimal solution.

Consider the problem of minimizing f, where

$$
\begin{aligned}
5x_1 - 4x_2 + 13x_3 - 2x_4 + x_5 &= 20 \\
x_1 - x_2 + 5x_3 - x_4 + x_5 &= 8 \\
x_1 + 6x_2 - 7x_3 + x_4 + 5x_5 - f &= 0
\end{aligned}
\tag{7.16}
$$

$$x_j \geq 0, \quad j = 1, 2, \ldots, 5 \tag{7.17}$$

We show how the canonical form can be used to improve a nonoptimal basic feasible solution.

Assume that we know that x_5, x_1, $-f$ can be used as basic variables and that the basic solution will be feasible. We can thus reduce system (7.16) to feasible canonical form by pivoting successively on the terms x_5 (first equation) and x_1 (second equation) ($-f$ already appears in the correct way). This yields

$$
\begin{aligned}
x_5 \qquad - 0.25x_2 + 3x_3 - 0.75x_4 &= 5 \\
x_1 \quad - 0.75x_2 + \textcircled{2x_3} - 0.25x_4 &= 3 \\
-f + 8x_2 - 24x_3 + 5x_4 &= -28
\end{aligned}
\tag{7.18}
$$

The circled term will be explained soon. The basic feasible solution is

$$x_5 = 5, \quad x_1 = 3, \quad x_2 = x_3 = x_4 = 0, \quad f = 28 \tag{7.19}$$

Note that an arbitrary pair of variables does not necessarily yield a basic solution to Equation (7.16) that is feasible. For example, had the variables x_1 and x_2 been chosen as basic variables, the basic solution would have been

$$x_1 = -12, \quad x_2 = -20, \quad x_3 = x_4 = x_5 = 0, \quad f = -132 \tag{7.20}$$

which is not feasible, because x_1 and x_2 are negative.

For the original basic feasible solution, one reduced cost is negative, namely $\bar{c}_3 = -24$. The optimality test of relations (7.15) thus fails. Furthermore, if x_3 is increased from its present value of zero (with all other nonbasic variables remaining zero), f must decrease because, by the third equation of (7.18), f is then related to x_3 by

$$f = 28 - 24x_3 \tag{7.21}$$

How large should x_3 become? It is reasonable to make it as large as possible, because the larger the value of x_3, the smaller the value of f. The constraints place a limit on the maximum value x_3 can attain, however. Note that, if $x_2 = x_4 = 0$, relations (7.18) state that the basic variables x_1, x_5 are related to x_3 by

$$
\begin{aligned}
x_5 &= 5 - 3x_3 \\
x_1 &= 3 - 2x_3
\end{aligned}
\tag{7.22}
$$

Thus as x_3 increases, x_5 and x_1 decrease, and they cannot be allowed to become negative. In fact, as x_3 reaches 1.5, x_1 becomes 0 and as x_3 reaches 1.667, x_5 becomes 0. By that time, however, x_1 is already negative, so the largest value x_3 can attain is

$$x_3 = 1.5 \tag{7.23}$$

Substituting this value into Equations (7.21) and (7.22) yields a new basic feasible solution with lower cost:

$$x_5 = 0.5, x_3 = 1.5, x_1 = x_2 = x_4 = 0, f = -8 \tag{7.24}$$

This solution reduces f from 28 to -8. The immediate objective is to see if it is optimal. This can be done if the system can be placed into feasible canonical form with x_5, x_3, $-f$ as basic variables. That is, x_3 must replace x_1 as a basic variable. One reason that the simplex method is efficient is that this replacement can be accomplished by doing one pivot transformation.

Previously x_1 had a coefficient of unity in the second equation of (7.18) and zero elsewhere. We now wish x_3 to have this property, and this can be accomplished by pivoting on the term $2x_3$, circled in the second equation of (7.18). This causes x_3 to become basic and x_1 to become nonbasic, as is seen here:

$$\begin{aligned} x_5 \quad - 1.5x_1 + \boxed{0.875x_2} - 0.375x_4 &= 0.5 \\ x_3 + 0.5x_1 - 0.375x_2 - 0.125x_4 &= 1.5 \\ -f + 12x_1 - x_2 + 2x_4 &= 8 \end{aligned} \tag{7.25}$$

This gives the basic feasible solution (7.24), as predicted. It also indicates that the present solution although better, is still not optimal, because $\bar{c}_2$, the coefficient of x_2 in the f equation, is -1. Thus we can again obtain a better solution by increasing x_2 while keeping all other nonbasic variables at zero. From Equation (7.25), the current basic variables are then related to x_2 by

$$\begin{aligned} x_5 &= 0.5 - 0.875x_2 \\ x_3 &= 1.5 + 0.375x_2 \\ f &= -8 - x_2 \end{aligned} \tag{7.26}$$

Note that the second equation places no bound on the increase of x_2, but the first equation restricts x_2 to a maximum of 0.5 / 0.875 = 0.571, which reduces x_5 to zero. As before, we obtain a new feasible canonical form by pivoting, this time using $0.875x_2$ in the first equation of (7.25) as the pivot term. This yields the system

$$\begin{aligned} x_2 \quad - 1.714x_1 - 0.429x_4 + 1.142x_5 &= 0.571 \\ x_3 \quad - 0.143x_1 - 10.286x_4 + 0.429x_5 &= 1.714 \\ -f + 10.286x_1 + 1.571x_4 + 1.143x_5 &= 8.571 \end{aligned} \tag{7.27}$$

and the basic feasible solution

$$x_2 = 0.571, x_3 = 1.714, x_1 = x_4 = x_5 = 0, f = -8.571 \tag{7.28}$$

Because all reduced costs for the nonbasic variables are positive, this solution is the unique minimal solution of the problem, by the corollary of the previous section. The optimum has been reached in two iterations.

Degeneracy

In the original system (7.18), if the constant on the right-hand side of the second equation had been zero, that is, if the basic feasible solution had been degenerate, then x_1 would have been related to x_3 by

$$x_1 = -2x_3 \tag{7.29}$$

And any positive change in x_3 would have caused x_1 to become negative. Thus x_3 would be forced to remain zero and f could not decrease. We go through the pivot transformation anyway and attain a new form in which the degeneracy may not be limiting. This can easily occur, for if relation (7.29) had been

$$x_1 = 2x_3$$

then x_3 could be made positive.

Unboundedness

If relations (7.26) had been

$$x_5 = 0.5 + 0.875x_2$$

$$x_3 = 1.5 + 0.375x_2$$

$$f = -8 - x_2$$

then x_2 could be made as large as desired without causing x_5 and x_3 to become negative, and f could be made as small as desired. This indicates an unbounded solution. Note that it occurs whenever all coefficients in a column with negative $\bar{c}_j$ are also negative (or zero).

Improving a nonoptimal basic feasible solution in general

Let us now formalize the procedures of the previous section. If at least one $\bar{c}_j < 0$, then, at least if we assume nondegeneracy (all $\bar{b}_i > 0$), it is always possible to construct, by pivoting, another basic feasible solution with lower cost. If more than one $\bar{c}_j < 0$, the variable x_s to be increased can be the one with the most negative $\bar{c}_j$; that is, the one whose relative cost factor is

$$\bar{c}_s = \min \bar{c}_j < 0 \tag{7.30}$$

Although this may not lead to the greatest decrease in f (because it may not be possible to increase x_s very far), this is intuitively at least a good rule for choosing the variable to become basic. More sophisticated "pricing" schemes have been developed, however, that perform much better and are included in most modern LP solvers [see Bixby, 1992]. An important recent innovation is the development of steepest edge pricing [see Forrest and Goldfarb (1992)].

Having decided on the variable x_s to become basic, we increase it from zero, holding all other nonbasic variables zero, and observe the effects on the current basic variables. By Equation (7.12), these are related to x_s by

$$\begin{aligned} x_1 &= b_1 - \bar{a}_{1s}x_s \\ x_2 &= b_2 - \bar{a}_{2s}x_s \\ &\vdots \\ x_m &= \bar{b}_m - \bar{a}_{ms}x_s \\ f &= \bar{f} + \bar{c}_s x_s, \quad \bar{c}_s < 0 \end{aligned} \tag{7.31}$$

Increasing x_s decreases f, and the only factor limiting the decrease is that one of the variables $x_1 \ldots x_m$ can become negative. However, if

$$\bar{a}_{is} \leq 0, \quad i = 1, 2, \ldots, m \tag{7.32}$$

then x_s can be made as large as desired. Thus we have the following result.

RESULT 5 (UNBOUNDEDNESS). If, in the canonical system for some s, all coefficients $\bar{a}_{is}$ are nonpositive and $\bar{c}_s$ is negative, then a class of feasible solutions can be constructed for which the set of f values has no lower bound.

The class of solutions yielding unbounded f is the set

$$x_i = \bar{b}_i - \bar{a}_{is}x_s, \quad i = 1, \ldots, m \tag{7.33}$$

with x_s any positive number and all other $x_i = 0$. If, however, at least one $\bar{a}_{is}$ is positive, then x_s cannot be increased indefinitely because eventually some basic variable becomes first zero, then negative. From Equation (7.31), x_i becomes zero when $\bar{a}_{is} > 0$ and when x_s attains the value

$$x_s = \frac{\bar{b}_i}{\bar{a}_{is}}, \bar{a}_{is} > 0 \tag{7.34}$$

The first x_i to become negative is the x_i that requires the smallest x_s to drive it to zero. This value of x_s is the greatest value for x_s permitted by the nonnegativity constraints and is given by

$$x_s^* = \frac{\bar{b}_r}{\bar{a}_{rs}} = \min_{\bar{a}_{is} > 0} \frac{\bar{b}_i}{\bar{a}_{is}} \tag{7.35}$$

The basic variable x_r then becomes nonbasic, to be replaced by x_s. We saw from the example in Equations (7.16)–(7.28) that a new canonical form with x_s replacing x_r as a basic variable is easily obtained by pivoting on the term $\bar{a}_{rs}x_s$. Note that the previous operations may be viewed as simply locating that pivot term. Finding $\bar{c}_s = \min \bar{c}_j < 0$ indicates that the pivot term was in column s, and finding that the minimum of the ratios $\bar{b}_i/\bar{a}_{is}$ for $\bar{a}_{is} > 0$ occurred for $i = r$ indicates that it was in row r.

As seen in the example, if the basic solution is degenerate, then the x_s^* given by Equation (7.35) may be zero. In particular, if some $\bar{b}_i = 0$ and the corresponding $\bar{a}_{is} > 0$ then, by Equation (7.35), $x_s^* = 0$. In this case the pivot operation is still carried out, but f is unchanged.

Iterative procedure

The procedure of the previous section provides a means of going from one basic feasible solution to one whose f is at least equal to the previous f (as can occur, in the degenerate case) or lower, if there is no degeneracy. This procedure is repeated until (1) the optimality test of relations (7.15) is passed or (2) information is provided that the solution is unbounded, leading to the main convergence result.

RESULT 6. Assuming nondegeneracy at each iteration, the simplex algorithm terminates in a finite number of iterations.

Because the number of basic feasible solutions is finite, the algorithm can fail to terminate only if a basic feasible solution is repeated. Such repetition implies that the same value of f is also repeated. Under nondegeneracy, however, each value of f is lower than the previous, so no repetition can occur, and the algorithm is finite.

Degenerate case

If, at some iteration, the basic feasible solution is degenerate, the possibility exists that f can remain constant for some number of subsequent iterations. It is then possible for a given set of basic variables to be repeated. An endless loop is then set up, the optimum is never attained, and the simplex algorithm is said to have cycled. Examples of cycling have been constructed [see Dantzig (1998), Chapter 10].

Some procedures are guaranteed to avoid cycling (Dantzig, 1998). Modern LP solvers contain very effective antidegeneracy strategies, although most are not guaranteed to avoid cycling. In practice, almost all LPs have degenerate optimal solutions. A high degree of degeneracy (i.e., a high percentage of basic variables at bounds) can slow the simplex method down considerably. Fortunately, an alternative class of LP algorithms, called *barrier methods*, are not affected by degeneracy. We discuss these briefly later in the chapter.

Two phases of the simplex method

The simplex algorithm requires a basic feasible solution as a starting point. Such a starting point is not always easy to find and, in fact, none exists if the constraints are inconsistent. Phase 1 of the simplex method finds an initial basic feasible solution or yields the information that none exists. Phase 2 then proceeds from this starting point to an optimal solution or yields the information that the solution is unbounded. Both phases use the simplex algorithm of the previous section.

Phase 1. Phase 1 starts with some initial basis **B** and an initial basic (possibly infeasible) solution $(\mathbf{x}_B, \mathbf{x}_N)$ satisfying

$$\mathbf{B}\mathbf{x}_B + \mathbf{N}\mathbf{x}_N = \mathbf{b} \tag{7.36}$$

In the previous expression, all components of $\mathbf{x}_N$ are at bounds and **N** is the corresponding matrix of coefficients for $\mathbf{x}_N$. Because **B** is nonsingular

$$\mathbf{x}_B = \mathbf{B}^{-1}(\mathbf{b} - \mathbf{N}\mathbf{x}_N) \tag{7.37}$$

If $\mathbf{x}_B$ is between its bounds, the basic solution is feasible and we begin phase 2, which optimizes the true objective. Otherwise, some components of $\mathbf{x}_B$ violate their bounds. Let L and U be the sets of indices of basic variables that violate their bounds, that is

$$x_j < l_j, \quad j \in L \tag{7.38}$$

and

$$x_j > u_j, \quad j \in U \tag{7.39}$$

Phase 1 minimizes the following linear objective function, the sum of infeasibilities, *sinf*;

$$sinf = \sum_{j \in L} (l_j - x_j) + \sum_{j \in U} (x_j - u_j) \tag{7.40}$$

Note that each term is positive, and that $sinf = 0$ if and only if the basic solution is feasible. When minimizing *sinf*, the standard simplex algorithm is applied, but the rules for choosing the pivot row described earlier must be changed, because some basic variables are now infeasible. During this process, infeasible basic variables can satisfy their bounds and feasible ones can violate their bounds, so the index sets L and U (and hence the function *sinf*) can change at any iteration. If the simplex optimality test is met and $sinf > 0$, then the LP is infeasible. Otherwise, when $sinf = 0$, phase 2 begins using the simplex method discussed earlier.

The initial basis

Often a good initial basis is known. Once an LP model is constructed and validated, it is common to do several series of case studies. In each case study, a set of LP data elements (cost or right-hand side components, bounds, or matrix elements a_{ij}) are assigned a sequence of closely related sets of values. For example, one may vary several costs through a range of values or equipment capacities or customer demands (both of the last two are right-hand sides or bounds). If there are several sets of parameter values, after the first set is solved, the optimal basis is stored and used as the initial basis for the LP problem that uses the second set, and so on. This usually sharply reduces computation time compared with a cold start, where no good initial basis is known. In fact, the simplex method's ability to warm start effectively is one of its major advantages over barrier methods (discussed later).

EXAMPLE 7.3 ITERATIVE SOLUTION OF AN LP PROBLEM

Consider first the problem illustrated geometrically in Figure 7.1 given in relations (7.1), that is

$$\text{Maximize:} \quad f = x_1 + 3x_2$$

$$\text{Subject to:} \quad -x_1 + x_2 + x_3 = 1$$

$$x_1 + x_2 + x_4 = 2$$

$$x_1 \geq 0, \quad i = 1, \ldots, 4 \qquad (a)$$

where x_3, x_4 are slack variables. Solve for the maximum using the simplex method.

Solution. Here no phase 1 is needed because an initial basic feasible solution is obvious. To apply directly the results of the previous sections, we rephrase the problem as

$$\text{Minimize:} \quad -x_1 - 3x_2$$

subject to Equation *(a)*. The initial feasible canonical form is

$$\begin{aligned} -x_1 + \textcircled{x_2} + x_3 \qquad &= 1 \\ x_1 + x_2 \qquad + x_4 &= 2 \\ -x_1 - 3x_2 \qquad - f &= 0 \end{aligned} \qquad (b)$$

The initial basic feasible solution is

$$x_1 = x_2 = 0, \quad x_3 = 1, \quad x_4 = 2, \quad f = 0 \qquad (c)$$

This corresponds to vertex (2) of Figure 7.1.

Iteration 1. Because $\bar{c}_2 = \min(\bar{c}_1, \bar{c}_2) = -3 < 0$, x_2 becomes basic. To see which variable becomes nonbasic, we compute the ratios b_i/a_{i2}; for all i such that $\bar{a}_{i2} > 0$. This gives

$$\frac{\bar{b}_1}{\bar{a}_{12}} = \frac{1}{1} = 1, \qquad \frac{\bar{b}_2}{\bar{a}_{22}} = \frac{2}{1} = 2$$

The minimum of these is $\bar{b}_1/\bar{a}_{12}$; thus the basic variable with unity coefficient in row 1, x_3, leaves the basis. The pivot term is $a_{12}x_2$ that is, the x_2 term circled in Equation *(b)*. Pivoting on this term yields

$$\begin{aligned} -x_1 + x_2 + x_3 \qquad &= 1 \\ \textcircled{2x_1} \qquad - x_3 + x_4 &= 1 \\ -4x_1 \qquad + 3x_3 \qquad - f &= 3 \end{aligned} \qquad (d)$$

Iteration 2. The new basic feasible solution is

$$x_1 = x_3 = 0, \quad x_2 = x_4 = 1, \quad f = -3$$

Note that f is reduced. The solution corresponds to vertex (3) of Figure 7.1. Because $\bar{c}_1 = -4 = \min_j \bar{c}_j$, x_1 becomes basic. The only ratio $\bar{b}_i/\bar{a}_{i1}$ having $\bar{a}_{i1} > 0$ is that for $i = 2$; thus x_4 becomes nonbasic and the circled pivot term is $\bar{a}_{21}x_1 = 2x_1$. Pivoting yields

$$\begin{aligned} x_2 + 0.5x_3 + 0.5x_4 \quad &= 1.5 \\ x_1 \quad - 0.5x_3 + 0.5x_4 \quad &= 0.5 \\ x_3 + 2x_4 - f &= 5 \end{aligned} \tag{e}$$

with basic feasible solution

$$x_1 = 0.5, x_2 = 1.5, x_3 = x_4 = 0, f = -5$$

which corresponds to vertex (4) of Figure 7.1. This is optimal, since all $\bar{c}_j > 0$. The path taken by the method is vertices (2), (3), (4).

7.4 BARRIER METHODS

Barrier methods for linear programming were first proposed in the 1980s and are now included in most commercial LP software systems. Their underlying principles and the way they operate are very different from the simplex method. They generate a sequence of points that may not satisfy all the constraints until the method converges and none of the points need be extreme points. This allows them to cut across the feasible region rather than moving from one extreme point to another, as the simplex method does. Hence they usually take far fewer iterations than the simplex method, but each iteration takes more time. See Martin (1999), Vanderbei (1999), or Wright (1999) for complete explanations. Current implementations of barrier methods are competitive with the best simplex codes, are often faster on very large problems, and often do very well in problems where the simplex method is slowed by degeneracy.

7.5 SENSITIVITY ANALYSIS

In addition to providing optimal **x** values, both simplex and barrier solvers provide values of dual variables or Lagrange multipliers for each constraint. We discuss Lagrange multipliers at some length in Chapter 8, and the conclusions reached there, valid for nonlinear problems, must hold for linear programs as well. In Chapter 8 we show that the dual variable for a constraint is equal to the derivative of the optimal objective value with respect to the constraint limit or right-hand side. We illustrate this with examples in Section 7.8.

7.6 LINEAR MIXED INTEGER PROGRAMS

A mixed integer linear program (MILP) is an LP in which one or more of the decision variables must be integers. A common subset of MILPs are binary, in which the integer variables can be either 0 or 1, indicating that something is either done or not done. For example, the binary variable $x_j = 1(0)$ can mean that a facility is (is not) placed at location j, or project j is (is not) selected. For such yes–no variables, fractional values have no significance. Almost all LP solvers now include the capability to solve MILPs, and this dramatically increases their usefulness. The computational difficulty of solving MILPs is determined mainly by the number of integer variables, and only in a secondary way by the number of continuous variables or constraints. Currently the best MILP solvers can handle hundreds of integer variables in reasonable time, sometimes more, depending on the problem structure and data. We discuss MILP's further in Chapter 9 [see also Martin (1999) and Wolsey (1998)].

7.7 LP SOFTWARE

LP software includes two related but fundamentally different kinds of programs. The first is solver software, which takes data specifying an LP or MILP as input, solves it, and returns the results. Solver software may contain one or more algorithms (simplex and interior point LP solvers and branch-and-bound methods for MILPs, which call an LP solver many times). Some LP solvers also include facilities for solving some types of nonlinear problems, usually quadratic programming problems (quadratic objective function, linear constraints; see Section 8.3), or separable nonlinear problems, in which the objective or some constraint functions are a sum of nonlinear functions, each of a single variable, such as

$$f(x) = x_1^2 + e^{x2} + 7/x_3 - 2x_4$$

Modeling systems

A second feature of LP programs is the inclusion of modeling systems, which provide an environment for formulating, solving, reporting on, analyzing, and managing LP and MILP models. Modeling systems have links to several LP, MILP, and NLP solvers and allow users to change solvers by changing a single statement. Modeling systems are all designed around a language for formulating optimization models, and most are capable of formulating and solving both linear and nonlinear problems. *Algebraic modeling systems* represent optimization problems using algebraic notation and a powerful indexing capability. This allows sets of similar constraints to be represented by a single modeling statement, regardless of the number of constraints in the set. For more information on algebraic modeling languages, see Section 8.9.3.

Another type of widely used modeling system is the *spreadsheet solver*. Microsoft Excel contains a module called the Excel Solver, which allows the user to enter the decision variables, constraints, and objective of an optimization problem into the cells of a spreadsheet and then invoke an LP, MILP, or NLP solver. Other spreadsheets contain similar solvers. For examples using the Excel Solver, see Section 7.8, and Chapters 8 and 9.

The power of linear programming solvers

Modern LP solvers can solve very large LPs very quickly and reliably on a PC or workstation. LP size is measured by several parameters: (1) the number of variables n, (2) the number of constraints m, and (3) the number of nonzero entries nz in the constraint matrix $\mathbf{A}$. The best measure is the number of nonzero elements nz because it directly determines the required storage and has a greater effect on computation time than n or m. For almost all LPs encountered in practice, nz is much less than mn, because each constraint involves only a few of the variables x. The *problem density* $100(nz/mn)$ is usually less than 1%, and it almost always decreases as m and n increase. Problems with small densities are called *sparse*, and real world LPs are always sparse. Roughly speaking, a problem with under 1000 nonzeros is small, between 1000 and 50,000 is medium-size, and over 50,000 is large. A small problem probably has m and n in the hundreds, a medium-size problem in the low to mid thousands, and a large problem above 10,000.

Currently, a good LP solver running on a fast (> 500 mHz) PC with substantial memory, solves a small LP in less than a second, a medium-size LP in minutes to tens of minutes, and a large LP in an hour or so. These codes hardly ever fail, even if the LP is badly formulated or scaled. They include preprocessing procedures that detect and remove redundant constraints, fixed variables, variables that must be at bounds in any optimal solution, and so on. Preprocessors produce an equivalent LP, usually of reduced size. A postprocessor then determines values of any removed variables and Lagrange multipliers for removed constraints. Automatic scaling of variables and constraints is also an option. Armed with such tools, an analyst can solve virtually any LP that can be formulated.

Solving MILPs is much harder. Focusing on MILPs with only binary variables, problems with under 20 binary variables are small, 20 to 100 is medium-size, and over 100 is large. Large MILPs may require many hours to solve, but the time depends greatly on the problem structure and the availability of a good starting point. We discuss MILP and MINLP formulations in Chapter 9.

Imbedded Linear Programming solvers

In addition to their use as stand-alone systems, LPs are often included within larger systems intended for decision support. In this role, the LP solver is usually hidden from the user, who sees only a set of critical problem input parameters and a set of suitably formatted solution reports. Many such systems are available for supply chain management—for example, planning raw material acquisitions and deliveries, production and inventories, and product distribution. In fact, the process industries—oil, chemicals, pharmaceuticals—have been among the earliest users. Almost every refinery in the developed world plans production using linear programming.

When embedded in decision support systems (usually in a Windows environment), LP solvers typically receive input data from a program written in C or Visual Basic and are often in the form of dynamic link libraries (DLLs). Most of today's LP solvers are available as DLLs.

Available Linear Programming software

Many LP software vendors advertise in the monthly journal *OR/MS Today*, published by INFORMS. For a survey of LP software, see Fourer (1997, 1999) in that journal. All vendors now have Websites, and the following table provides a list of LP software packages along with their Web addresses.

Company name	Solver name	Web addresses/E-mail address
CPLEX Division of ILOG	CPLEX	www.cplex.com
IBM	Optimization Software Library (OSL)	www.research.ibm.com/osl/
LINDO Systems Inc.	LINDO	www.lindo.com
Dash Associates	XPRESS-MP	www.dashopt.com
Sunset Software Technology	AXA	Sunsetw@ix.netcom.com
Advanced Mathematical Software	LAMPS	info@amsoft.demon.co.uk

7.8 A TRANSPORTATION PROBLEM USING THE EXCEL SOLVER SPREADSHEET FORMULATION

Figure 7.3 displays a Microsoft Excel spreadsheet containing the formulas and data for an LP transportation problem. This spreadsheet is one of six optimization examples included with Microsoft Excel '97. With a standard installation of Microsoft Office, the Excel workbook containing all six examples is in the file

MicrosoftOffice/office/examples/solver/solvsamp.xls

We encourage the reader to start Excel on his or her computer, find and open this file, and examine and solve this spreadsheet as the rest of this section is read. The 15 decision variables are the number of units of a single product to ship from three plants to five warehouses. Initial values of these variables (all ones) are in the range C8:G10. The constraints are (1) the amount shipped from each plant cannot exceed the available supply, given in range B16:B18, (2) the amount shipped to each warehouse must meet or exceed demand there, given in range C14:G14, and (3) all amounts shipped must be nonnegative. Cells C16:G18 contain the per unit costs of shipping the product along each of the 15 possible routes. The total cost of shipping into each warehouse is in the range C20:G20, computed by multiplying the amounts shipped by their per unit costs and summing. Total shipping cost in cell B20 is to be minimized. Before reading further, attempt to find an optimal solution to this problem by trying your own choices for the decision variables.

	A	B	C	D	E	F	G	H	I	J	K
1	**Example 2: Transportation Problem.**										
2	Minimize the costs of shipping goods from production plants to warehouses near metropolitan demand										
3	centers, while not exceeding the supply available from each plant and meeting the demand from each										
4	metropolitan area.										
6			*Number to ship from plant x to warehouse y (at intersection):*								
7	*Plants:*	*Total*	*San Fran*	*Denver*	*Chicago*	*Dallas*	*New York*				
8	S. Carolina	5	1	1	1	1	1				
9	Tennessee	5	1	1	1	1	1				
10	Arizona	5	1	1	1	1	1				
11			–	–	–	–	–				
12	Totals:		3	3	3	3	3				
13											
14	*Demands by Whse* ⟶		180	80	200	160	220				
15	*Plants:*	*Supply*	*Shipping costs from plant x to warehouse y (at intersection):*								
16	S. Carolina	310	10	8	6	5	4				
17	Tennessee	260	6	5	4	3	6				
18	Arizona	280	3	4	5	5	9				
19											
20	*Shipping:*	$83	$19	$17	$15	$13	$19				
21											
22	The problem presented in this model involves the shipment of goods from three plants to five regional										
23	warehouses. Goods can be shipped from any plant to any warehouse, but it obviously costs more to										
24	ship goods over long distances than over short distances. The problem is to determine the amounts										
25	to ship from each plant to each warehouse at minimum shipping cost in order to meet the regional										
26	demand, while not exceeding the plant supplies.										
27											
28	Problem Specifications										
30	Target cell		B20		Goal is to minimize total shipping cost.						
32	Changing cells		C8:G10		Amount to ship from each plant to each						
33					warehouse.						
35	Constraints		B8:B10<=B16:B18		Total shipped must be less than or equal to						
36					supply at plant.						
38			C12:G12>=C14:G14		Totals shipped to warehouses must be greater						
39					than or equal to demand at warehouses.						
41			C8:G10>=0		Number to ship must be greater than or equal						
42					to 0.						
43											
44	You can solve this problem faster by selecting the **Assume linear model** check box in the **Solver**										
45	**Options** dialog box before clicking **Solve.** A problem of this type has an optimum solution at which										
46	amounts to ship are integers, if all of the supply and demand constraints are integers.										

FIGURE 7.3
A transportation problem in a Microsoft Excel spreadsheet format. Permission by Microsoft.

Algebraic formulation

Let x_{ij} be the number of units of the product shipped from plant i to warehouse j. Then the supply constraints are

$$\sum_{j=1}^{5} x_{ij} \leq \text{avail}_i, \quad i = 1,2,3 \tag{7.41}$$

The demand constraints are

$$\sum_{i=1}^{3} x_{ij} \geq \text{demand}_j, \quad j = 1, \ldots, 5 \tag{7.42}$$

and the nonnegativities:

$$x_{ij} \geq 0, \quad \text{all}\, i, j \tag{7.43}$$

The objective is to minimize

$$\text{Cost} = \sum_{j=1}^{5} \sum_{i=1}^{3} c_{ij} x_{ij} \tag{7.44}$$

Solver parameters dialog

To define this problem for the Excel Solver, the cells containing the decision variables, the constraints, and the objective must be specified. This is done by choosing the Solver command from the Tools menu, which causes the Solver parameters dialog shown in Figure 7.4 to appear. The "Target Cell" is the cell containing the objective function. Clicking the "Help" button explains all the steps needed to enter the "changing" (i.e., decision) variables and the constraints. We encourage you to "Reset all," and fill in this dialog from scratch.

Solver options dialog

Selecting the "Options" button in the Solver Parameters dialog brings up the Solver Options dialog box shown in Figure 7.5. The current Solver version does not determine automatically if the problem is linear or nonlinear. To inform Solver that

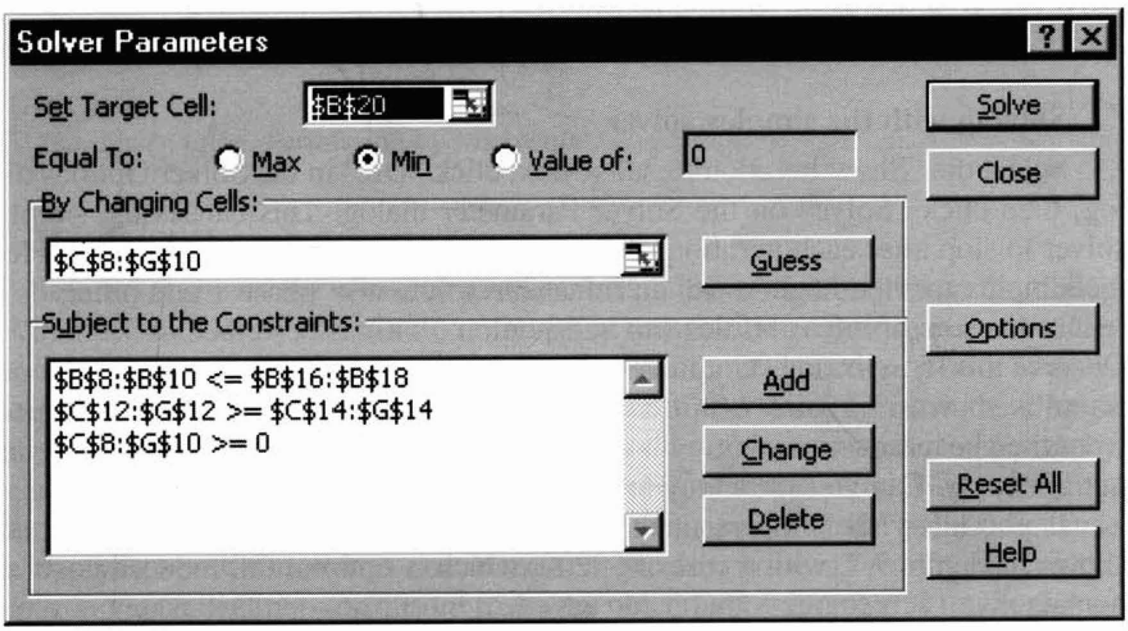

FIGURE 7.4
Solver parameters dialog box. Permission by Microsoft.

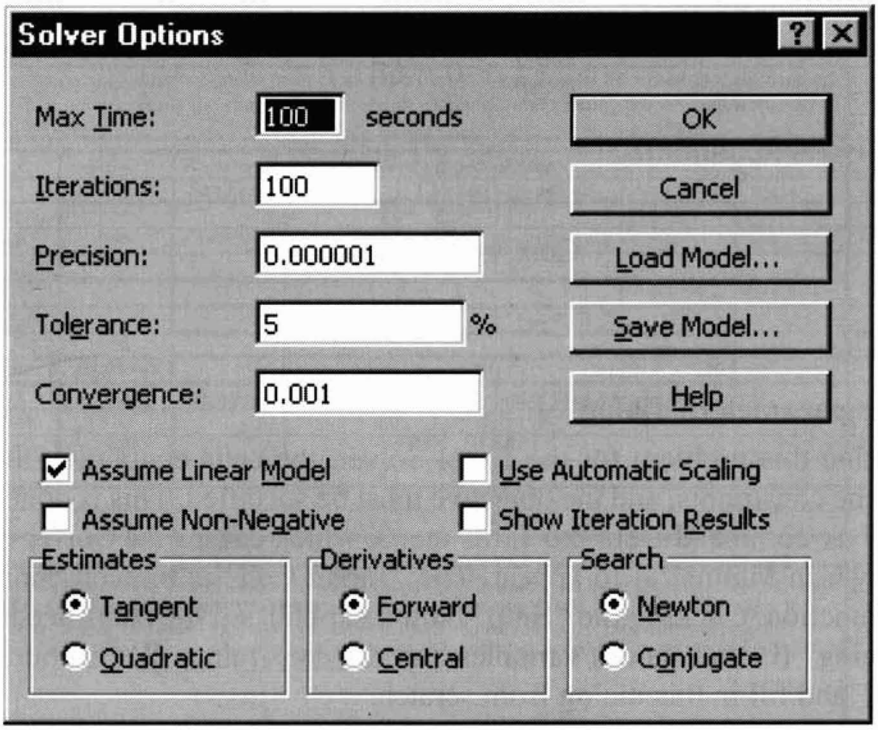

FIGURE 7.5
Solver options dialog box. Permission by Microsoft.

the problem is an LP, select the "Assume Linear Model" box. This causes the simplex solver to be used. It is both faster and more accurate for LPs than the generalized reduced gradient (GRG) nonlinear solver, which is the default choice. The GRG solver is discussed in Chapter 8.

Solving with the simplex solver

Select the "Show Iteration Results" box, click "OK" in the Solver Options dialog, then click "Solve" on the Solver Parameter dialog. This causes the simplex solver to stop after each iteration. Because an initial feasible basis is not provided, the simplex method begins with an infeasible solution in phase 1 and proceeds to reduce the sum of infeasibilities *sinf* in Equation (7.40) as described in Section 7.3. Observe this by selecting "Continue" after each iteration. The first feasible solution found is shown in Figure 7.6. It has a cost of \$3210, with most shipments made from the cheapest source, but with other sources used when the cheapest one runs out of supply. Can you see a way to improve this solution?

If you allow the simplex method to continue, it finds the improved solution shown in Figure 7.7, with a cost of \$3200, which is optimal (all reduced costs are nonnegative). It recognizes that it can save \$20 by shifting ten Dallas units from S. Carolina to Tennessee, if it frees up ten units of supply at Tennessee by supplying Chicago from Arizona (which costs only \$10 more). Supplies at Arizona and Tennessee are completely used, but South Carolina has ten units of excess supply.

	A	B	C	D	E	F	G	H
1	**Example 2: Transportation Problem.**							
2	Minimize the costs of shipping goods from production plants to warehouses near metropolitan demand centers, while not exceeding the supply available from each plant and meeting the demand from each metropolitan area.							
6			*Number to ship from plant x to warehouse y (at intersection):*					
7	*Plants:*	*Total*	*San Fran*	*Denver*	*Chicago*	*Dallas*	*New York*	
8	S. Carolina	310	0	0	0	90	220	
9	Tennessee	260	0	0	190	70	0	
10	Arizona	270	180	80	10	0	0	
11			–	–	–	–	–	
12	Totals:		180	80	200	160	220	
13								
14	*Demands by Whse* ⟶		180	80	200	160	220	
15	*Plants:*	*Supply*	*Shipping costs from plant x to warehouse y (at intersection):*					
16	S. Carolina	310	10	8	6	5	4	
17	Tennessee	260	6	5	4	3	6	
18	Arizona	280	3	4	5	5	9	
19								
20	*Shipping:*	$3,210	$540	$320	$810	$660	$880	
21								

The problem presented in this model involves the shipment of goods from three plants to five regional warehouses. Goods can be shipped from any plant to any warehouse, but it obviously costs more to ship goods over long distances than over short distances. The problem is to determine the amounts to ship from each plant to each warehouse at minimum shipping cost in order to meet the regional demand, while not exceeding the plant supplies.

Problem Specifications

Target cell	B20	Goal is to minimize total shipping cost.
Changing cells	C8:G10	Amount to ship from each plant to each warehouse.
Constraints	B8:B10<=B16:B18	Total shipped must be less than or equal to supply at plant.
	C12:G12>=C14:G14	Totals shipped to warehouses must be greater than or equal to demand at warehouses.
	C8:G10>=0	Number to ship must be greater than or equal to 0.

You can solve this problem faster by selecting the **Assume linear model** check box in the **Solver Options** dialog box before clicking **Solve**. A problem of this type has an optimum solution at which amounts to ship are integers, if all of the supply and demand constraints are integers.

FIGURE 7.6
First feasible solution. Permission by Microsoft.

The sensitivity report

Figure 7.8 shows the sensitivity report, which can be selected from the dialog box that appears when the solution algorithm finishes. The most important information is the "Shadow Price" column in the "constraints" section. These shadow prices (also called dual variables or Lagrange multipliers) are equal to the change in the optimal objective value if the right-hand side of the constraint increases by one unit, with all other right-hand side values remaining the same. Hence the first three multipliers show the effect of increasing the supplies at the plants. Because the supply in South Carolina is not all used, its shadow price is zero. Increasing the supply in Tennessee by one unit improves the objective by 2, twice as much as Arizona. To verify this, increase the Tennessee supply to 261, resolve, and observe that the new objective value is $3198. The last five shadow prices show the effects of increasing the demands. The "Allowable Increase" is the amount the right-hand

	A	B	C	D	E	F	G	H	I	J	K
1	**Example 2: Transportation Problem.**										
2	Minimize the costs of shipping goods from production plants to warehouses near metropolitan demand										
3	centers, while not exceeding the supply available from each plant and meeting the demand from each										
4	metropolitan area.										
6			*Number to ship from plant x to warehouse y (at intersection):*								
7	*Plants:*	*Total*	*San Fran*	*Denver*	*Chicago*	*Dallas*	*New York*				
8	S. Carolina	300	0	0	0	80	220				
9	Tennessee	260	0	0	180	80	0				
10	Arizona	280	180	80	20	0	0				
11			–	–	–	–	–				
12	Totals:		180	80	200	160	220				
13											
14	*Demands by Whse* →		180	80	200	160	220				
15	*Plants:*	*Supply*	*Shipping costs from plant x to warehouse y (at intersection):*								
16	S. Carolina	310	10	8	6	5	4				
17	Tennessee	260	6	5	4	3	6				
18	Arizona	280	3	4	5	5	9				
19											
20	*Shipping:*	$3,200	$540	$320	$820	$640	$880				
21											
22	The problem presented in this model involves the shipment of goods from three plants to five regional										
23	warehouses. Goods can be shipped from any plant to any warehouse, but it obviously costs more to										
24	ship goods over long distances than over short distances. The problem is to determine the amounts										
25	to ship from each plant to each warehouse at minimum shipping cost in order to meet the regional										
26	demand, while not exceeding the plant supplies.										
27											
28	Problem Specifications										
30	Target cell		B20		Goal is to minimize total shipping cost.						
32	Changing cells		C8:G10		Amount to ship from each plant to each						
33					warehouse.						
35	Constraints		B8:B10<=B16:B18		Total shipped must be less than or equal to						
36					supply at plant.						
38			C12:G12>=C14:G14		Totals shipped to warehouses must be greater						
39					than or equal to demand at warehouses.						
41			C8:G10>=0		Number to ship must be greater than or equal						
42					to 0.						
43											
44	You can solve this problem faster by selecting the **Assume linear model** check box in the **Solver**										
45	**Options** dialog box before clicking **Solve.** A problem of this type has an optimum solution at which										
46	amounts to ship are integers, if all of the supply and demand constraints are integers.										

FIGURE 7.7
Optimal solution. Permission by Microsoft.

side can increase before the shadow price changes, and similarly for the "Allowable Decrease." Beyond these ranges, some shipment that is now zero becomes positive while some positive one becomes zero. Try right-hand side changes within and slightly beyond one of the ranges to verify this.

The "Adjustable Cells" section contains sensitivity information on changes in the objective coefficients. The reduced costs are the qualities $\bar{c}_j$ discussed in Section 7.3. These are all nonnegative, as they must be in an optimal solution—see result 3. Note that the $\bar{c}_j$ for the South Carolina–Chicago shipment is zero, indicating that this problem has multiple optima (because this optimal solution is nondegenerate, i.e., all basic variables are positive). The following table shows a set of shipping unit amounts that yields no net cost change.

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
C8	S. Carolina San Fran	0	6	10	1E+30	6
D8	S. Carolina Denver	0	3	8	1E+30	3
E8	S. Carolina Chicago	0	0	6	1E+30	0
F8	S. Carolina Dallas	80	0	5	0	1
G8	S. Carolina New York	220	0	4	4	4
C9	Tennessee San Fran	0	4	6	1E+30	4
D9	Tennessee Denver	0	2	5	1E+30	2
E9	Tennessee Chicago	180	0	4	0	1
F9	Tennessee Dallas	80	0	3	1	0
G9	Tennessee New York	0	4	6	1E+30	4
C10	Arizona San Fran	180	0	3	4	4
D10	Arizona Denver	80	0	4	2	5
E10	Arizona Chicago	20	0	5	1	2
F10	Arizona Dallas	0	1	5	1E+30	1
G10	Arizona New York	0	6	9	1E+30	6

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
B8	S. Carolina Supply	300	0	310	1E+30	10
B9	Tennessee Supply	260	-2	260	80	10
B10	Arizona Supply	280	-1	280	80	10
C12	San Fran Demand	180	4	180	10	80
D12	Denver Demand	80	5	80	10	80
E12	Chicago Demand	200	6	200	10	80
F12	Dallas Demand	160	5	160	10	80
G12	New York Demand	220	4	220	10	220

FIGURE 7.8
Sensitivity report.

Shipment	**Change**	**Cost change**
South Carolina–Chicago	+1	+6
Tennessee–Chicago	−1	−4
Tennessee–Dallas	+1	+3
S. Carolina–Dallas	−1	−5
Total		0

These changes leave the amounts shipped out from the plants and into the warehouses unchanged.

7.9 NETWORK FLOW AND ASSIGNMENT PROBLEMS

This transportation problem is an example of an important class of LPs called *network flow problems*: Find a set of values for the flow of a single commodity on the arcs of a graph (or network) that satisfies both flow conservation constraints at each node (i.e., flow in equals flow out) and upper and lower limits on each flow, and maximize or minimize a linear objective (say, total cost). There are specified supplies of the commodity at some nodes and demands at others. Such problems have the important special property that, if all supplies, demands, and flow bounds are integers, then an optimal solution exists in which all flows are integers. In addition, special versions of the simplex method have been developed to solve network flow problems with hundreds of thousands of nodes and arcs very quickly, at least ten times faster than a general LP of comparable size. See Glover et al. (1992) for further information.

The integer solution property is particularly important in assignment problems. These are transportation problems (like the problem just described) with n supply nodes and n demand nodes, where each supply and demand is equal to 1.0, and all constraints are equalities. Then the model in Equations (7.41) through (7.44) has the following interpretation: Each supply node corresponds to a "job," and each demand node to a "person." The problem is to assign each "job" to a "person" so that some measure of benefit or cost is optimized. The variables x_{ij} are 1 if "job" i is assigned to "person" j, and zero otherwise.

As an example, suppose we want to assign streams to heat exchangers and the cost (in some measure) of doing so is listed in the following matrix:

$$
\begin{array}{r c}
 & \text{Exchanger number} \\
 & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\
\begin{array}{r} A \\ \textit{Stream}\ B \\ C \\ D \end{array} &
\begin{bmatrix} 94 & 1 & 54 & 68 \\ 74 & 10 & 88 & 82 \\ 73 & 88 & 8 & 76 \\ 11 & 74 & 81 & 21 \end{bmatrix}
\end{array}
$$

Each element in the matrix represents the cost of transferring stream i to exchanger j. How can the cost be minimized if each stream goes to only one exchanger?

First let us write the problem statement. The total number of streams n is 4. Let c_{ij} be an element of the cost matrix, which is the cost of assigning stream i to exchanger j. Then we have the following assignment problem:

$$\text{Minimize:} \quad f(\mathbf{x}) = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}x_{ij}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad j = l, \ldots, n \tag{7.45}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad i = l, \ldots, n$$

$$x_{ij} \geq 0 \quad i,j = l, \ldots, n$$

The constraints (7.46) ensure that each stream is assigned to some exchanger, and Equation (7.45) ensures that each exchanger is assigned one stream. Because the supplies and demands are integers, this problem has an optimal integer solution, with each x_{ij} equal to 0 or 1. The reader is invited to solve this problem using the Excel Solver (or any other LP solver) and find an optimal assignment.

REFERENCES

Bixby, R. E. "Implementing the Simplex Method: The Initial Basis." *ORSA J Comput* **4**(3): 267–284 (1992).

Dantzig, G. B. *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ (1998).

Forrest, J. J.; and D. Goldfarb. "Steepest-edge Simplex Algorithms for Linear Programming." *Math Prog* **57:** 341–374 (1992).

Fourer, R. "Linear Programming." *OR/MS Today* **24:** 54–67 (1997).

Fourer, R. "Software for Optimization." *OR/MS Today* **26:** 40–44 (1999).

Glover, F.; D. Klingman; and N. Phillips. *Network Models in Optimization and Their Applications in Practice*. Wiley, New York (1992).

Martin, R. K. *Large Scale Linear and Integer Optimization*. Kluwer Academic Publishers, Norwell, MA (1999).

Vanderbei, R. J. *Linear Programming Foundations and Extensions*. Kluwer Academic Publishers, Norwell, MA (1999).

Wolsey, L. A. *Integer Programming*. Wiley, New York (1998).

Wright, S. J. *Primal-Dual Interior-Point Methods*. SIAM, Philadelphia, PA (1999).

SUPPLEMENTARY REFERENCES

Darst, R. B. *Introduction to Linear Programming*. Marcel Dekker, New York (1990).

Gill, P. E.; W. Murray; M. A. Saunders; J. A. Tomlin; and M. H. Wright. "On Projected Newton Barrier Methods for Linear Programming and an Equivalence to Karmarkar's Projective Method." *Math Program* **36:** 183–191 (1986).

Gass, S. I. *An Illustrated Guide to Linear Programming*. Dover, Mineola, NY (1990).

Johnson, J. D.; and C. Q. Williamson. "In-Line Gasoline Blending at Suntide Refinery." *IEEE Trans. Industry and General Applications,* **159:** 167–179 (March/April, 1967).

Karmarkar, N. "A New Polynomial-Time Algorithm for Linear Programming." *Combinatoria* **4:** 373–383 (1984).

Murtagh, B. A. *Advanced Linear Programming*. McGraw-Hill, New York (1983).

Murty, K. G. *Linear Programming*. Wiley, New York (1983).

Perry C.; and R. Crellin. "The Precise Meaning of a Shadow Price." *Interfaces* **12:** 61–68 (1982).

Shamir, R. "The Efficiency of the Simplex Method: A Survey." *Man. Sci.* **33:** 301–310 (1987).
Schrage, L. *Optimization Modeling with LINDO*. Duxbury Press, Pacific Grove, CA (1997).
Schrijver, A. *Theory of Linear and Integer Programming*. Wiley, New York (1986).
Snee, R. D. "Developing Blending Models for Gasoline and Other Mixtures." *Technometrics* **23:** 119–127 (1981).
Sourander, M. L.; M. Kolari; J. C. Cugini; J. B. Poje; and D. C. White. "Control and Optimization of Olefin-Cracking Heaters." *Hydrocarbon Process.* pp. 63–68 (June, 1984).
Ye, Y. *Interior Point Algorithms: Theory and Analysis*. Wiley, New York (1997).

PROBLEMS

7.1 A refinery has available two crude oils that have the yields shown in the following table. Because of equipment and storage limitations, production of gasoline, kerosene, and fuel oil must be limited as also shown in this table. There are no plant limitations on the production of other products such as gas oils.

The profit on processing crude #1 is \$1.00/bbl and on crude #2 it is \$0.70/bbl. Find the approximate optimum daily feed rates of the two crudes to this plant via a graphical method.

	Volume percent yields		Maximum allowable product rate (bbl/day)
	Crude #1	**Crude #2**	
Gasoline	70	31	6,000
Kerosene	6	9	2,400
Fuel oil	24	~~60~~	12,000

7.2 A confectioner manufactures two kinds of candy bars: Ergies (packed with energy for the kiddies) and Nergies (the "lo-cal" nugget for weight watchers without willpower). Ergies sell at a profit of 50¢ per box, and Nergies have a profit of 60¢ per box. The candy is processed in three main operations: blending, cooking, and packaging. The following table records the average time in minutes required by each box of candy, for each of the three activities.

	Blending	**Cooking**	**Packing**
Ergies	1	5	3
Nergies	2	4	1

During each production run, the blending equipment is available for a maximum of 14 machine hours, the cooking equipment for at most 40 machine hours, and the packaging equipment for at most 15 machine hours. If each machine can be allocated to the making of either type of candy at all times that it is available for production, determine how many boxes of each kind of candy the confectioner should make to realize the maximum profit. Use a graphical technique for the two variables.

7.3 Feed to three units is split into three streams: $F_A, F_B,$ and F_C. Two products are produced: P_1 and P_2 (see following figure), and the yield in weight percent by unit is

Yield (weight %)	Unit A	Unit B	Unit C
P_1	40	30	50
P_2	60	70	50

Each stream has values in $/lb as follows:

Stream	F	P_1	P_2
Value ($/lb)	.40	.60	.30

Because of capacity limitations, certain constraints exist in the stream flows:

1. The total input feed must not exceed 10,000 lb/day.
2. The feed to each of the units A, B, and C must not exceed 5000 lb/day.
3. No more than 4000 lb/day of P_1 can be used, and no more than $7000 lb/day of P_2 can be used.

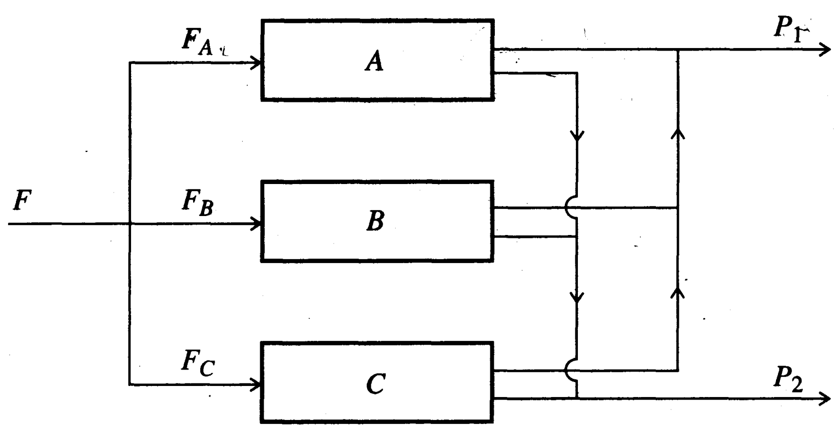

FIGURE P7.3

In order to determine the values of $F_A, F_B,$ and F_C that maximize the daily profit, prepare a mathematical statement of this problem as a linear programming problem. Do *not* solve it.

7.4 Prepare a graph of the constraints and objective function, and solve the following linear programming problem

$$\begin{aligned} \text{Maximize:} \quad & x_1 + 2x_2 \\ \text{Subject to:} \quad & -x_1 + 3x_2 < 10 \\ & x_1 + x_2 \leq 6 \\ & x_1 - x_2 \leq 2 \\ & x_1 + 3x_2 \geq 6 \\ & 2x_1 + x_2 \geq 4 \\ & x_1 \geq 0 \quad x_2 \geq 0 \end{aligned}$$

7.5 A chemical manufacturing firm has discontinued production of a certain unprofitable product line. This has created considerable excess production capacity on the three existing batch production facilities. Management is considering devoting this excess capacity to one or more of three new products: Call them products 1, 2, and 3. The available capacity on the existing units that might limit output is summarized in the following table:

Unit	Available time (h/week)
A	20
B	10
C	5

Each of the three new products requires the following processing time for completion:

Productivity (h/batch)			
Unit	**Product 1**	**Product 2**	**Product 3**
A	0.8	0.2	0.3
B	0.4	0.3	
C	0.2		0.1

The sales department indicates that the sales potential for products 1 and 2 exceeds the maximum production rate and that the sales potential for product 3 is 20 batches per week. The profit per batch is $20, $6, and $8, respectively, on products 1, 2, and 3.

Formulate a linear programming model for determining how much of each product the firm should produce to maximize profit.

7.6 An oil refinery has to blend gasoline. Suppose that the refinery wishes to blend four petroleum constituents into three grades of gasoline: *A*, *B*, and *C*. Determine the mix of the four constituents that will maximize profit.

The availability and costs of the four constituents are given in the following table:

Constituent*	Maximum quantity available (bbl/day)	Cost per barrel ($)
1	3000	13.00
2	2000	15.30
3	4000	14.60
4	1000	14.90

*1 = butane
2 = straight-run
3 = thermally cracked
4 = catalytic cracked

To maintain the required quality for each grade of gasoline, it is necessary to specify certain maximum or minimum percentages of the constituents in each blend. These are shown in the following table, along with the selling price for each grade.

Grade	Specification	Selling price per barrel ($)
A	Not more than 15% of 1 Not less than 40% of 2 Not more than 50% of 3	16.20
B	Not more than 10% of 1 Not less than 10% of 2	15.75
C	Not more than 20% of 1	15.30

Assume that all other cash flows are fixed so that the "profit" to be maximized is total sales income minus the total cost of the constituents. Set up a linear programming model for determining the amount and blend of each grade of gasoline.

7.7 A refinery produces, on average, 1000 gallon/hour of virgin pitch in its crude distillation operation. This pitch may be blended with flux stock to make commercial fuel oil, or it can be sent in whole or in part to a visbreaker unit as shown in Figure P7.7. The visbreaker produces an 80 percent yield of tar that can also be blended with flux stock to make commercial fuel oil. The visbreaking operation is economically break-even if the pitch and the tar are given no value, that is, the value of the overhead product equals the cost of the operation. The commercial fuel oil brings a realization of 5¢/gal, but the flux stock has a cracking value of 8¢/gal. This information together with the viscosity and gravity blending numbers and product specifications, appears in the following table. It is desired to operate for maximum profit.

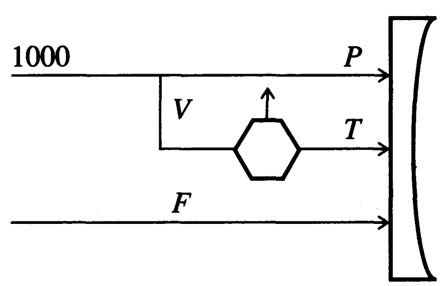

FIGURE P7.7

Fuel oil blending problem

	Quantity available (gal/h)	Value (¢/gal)	Viscosity Bl. No.	Gravity Bl. No.
Pitch	$P = 1000 - V$	0	5	8
Visbreaker feed	V	0	—	—
Tar	$T = 0.8V$	0	11	7
Flux	$F =$ any	8	37	24
Fuel oil	$P + T + F$	5	21 min	12 min

Abbreviation: Bl. No. = blending number.

Formulate the preceding problem as a linear programming problem. How many variables are there? How many inequality constraints? How many equality constraints? How many bounds on the variables?

7.8 Examine the following problem:

$$\text{Minimize: } f = 3x_1 + x_2 + x_3$$
$$\text{Subject to: } \quad x_1 - 2x_2 + x_3 \leq 11$$
$$-4x_1 + x_2 + 2x_3 \geq 3$$
$$2x_1 - x_3 = -1$$
$$x_1, x_2, x_3 \geq 0$$

Is there a basic feasible solution to the problem? Answer yes or no, and explain.

7.9 An LP problem has been converted to standard canonical form by the addition of slack variables and has a basic feasible solution (with $x_1 = x_2 = 0$) as shown in the following set of equations:

$$-2x_1 + 2x_2 + x_3 = 3$$
$$5x_1 + 2x_2 + x_4 = 11$$
$$x_1 + x_2 + x_5 = 4$$
$$4x_1 + 2x_2 + f = 0$$

Answer the following questions:
(a) Which variable should be increased first?
(b) Which row and which column designate the pivot point?
(c) What is the limiting value of the variable you designated part in (a)?

7.10 For the problem given in 7.9, find the next basis. Show the steps you take to calculate the improved solution, and indicate what the basic variables and nonbasic variables are in the new set of equations. (Just a single step from one vertex to the next is asked for in this problem.)

7.11 Examine the following problem

$$\text{Minimize: } f = 3x_1 + x_2 + x_3$$
$$\text{Subject to: } \quad x_1 - 2x_2 + x_3 \leq 11$$
$$-4x_1 + x_2 + 2x_3 \geq 3$$
$$2x_1 - x_3 = -1$$
$$x_1, x_2, x_3 \geq 0$$

Is there a basic feasible solution to the problem? Answer yes or no, and explain.

7.12 You are asked to solve the following problem:

$$\text{Maximize: } f = 5x_1 + 2x_2 + 3x_3$$
$$\text{Subject to: } \quad x_1 + 2x_2 + 2x_3 + x_4 = 8$$
$$3x_1 + 4x_2 + x_3 - x_5 = 7$$
$$x_1, \ldots, x_5 \geq 0$$

Explain in detail what you would do to obtain the first feasible solution to this problem. Show all equations. You do not have to calculate the feasible solution—just explain in detail how you would calculate it.

7.13 You are given the following LP equation sets:

(a)

$$\begin{aligned} 3x_1 - x_2 + x_3 &= -6 \\ 4x_1 - 3x_2 + x_4 &= -4 \\ x_1 + 3x_2 + f &= 0 \end{aligned}$$

Why is this formulation problematic?

(b)

$$\begin{aligned} x_1 - 2x_2 + x_3 &= 7 \\ x_1 - 3x_2 + x_4 &= 4 \\ x_1 + 3x_2 + f &= 0 \end{aligned}$$

Is the problem that leads to the preceding formulation solvable? How do you interpret this problem geometrically?

(c)

$$\begin{aligned} 4x_1 + 2x_2 + x_3 &= 6 \\ 6x_1 + 3x_2 + x_4 &= 9 \\ x_1 + 3x_2 + f &= 0 \end{aligned}$$

Apply the simplex rules to minimize f for the formulation. Is the solution unique?

(d)

$$\begin{aligned} 4x_1 + 2x_2 + x_3 &= 7 \\ 6x_1 + 3x_2 + x_4 &= 5 \\ -x_1 + f &= 0 \end{aligned}$$

Can you find the minimum of f? Why or why not?

7.14 Solve the following LP:

$$\text{Minimize: } f = x_1 + x_2$$

$$\begin{aligned} \text{Subject to: } x_1 + 3x_2 &\le 12 \\ x_1 - x_2 &\le 1 \\ 2x_1 - x_2 &\le 4 \\ 2x_1 + x_2 &\le 8 \\ x_1 \ge 0 \quad x_2 &\ge 0 \end{aligned}$$

Does the solution via the simplex method exhibit cycling?

7.15 In Problem 7.1 what are the shadow prices for incremental production of gasoline, kerosene, and fuel oil? Suppose the profit coefficient for crude #1 is increased by 10 percent and crude #2 by 5 percent. Which change has a larger influence on the objective function?

7.16 For Problem 7.9, find the next basis. Show the steps for calculating the new table, and indicate the basic and nonbasic variables in the new table. (Just a single step from one vertex to the next is asked for in this problem.)

7.17 Solve the following linear programming problem:

$$\text{Maximize: } f = x_1 + 3x_2 - x_3$$
$$\text{Subject to: } x_1 + 2x_2 + x_3 = 4$$
$$2x_1 + x_2 \le 5$$

7.18 Solve the following problem:

$$\text{Maximize: } f = 7x_1 + 12x_2 + 3x_3$$
$$\text{Subject to: } 2x_1 + 2x_2 + x_3 \le 16$$
$$4x_1 + 8x_2 + x_3 \le 40$$
$$x_1, x_2, x_3 \ge 0$$

7.19 Solve the following problem:

$$\text{Maximize: } f = 6x_1 + 5x_2$$
$$\text{Subject to: } 2x_1 + 5x_2 \le 20$$
$$-5x_1 - x_2 \le -5$$
$$-3x_1 - 11x_2 \le -33$$

The following figure shows the constraints. If slack variables x_3, x_4 and x_5 are added respectively to the inequality constraints, you can see from the diagram that the origin is not a feasible point, that is, you cannot start the simplex method by letting $x_1 = x_2 = 0$ because then $x_3 = 20$, $x_4 = -5$, and $x_5 = -33$, a violation of the assumption in linear programming that $x_i \ge 0$. What should you do to apply the simplex method to the problem other than start a phase I procedure of introducing artificial variables?

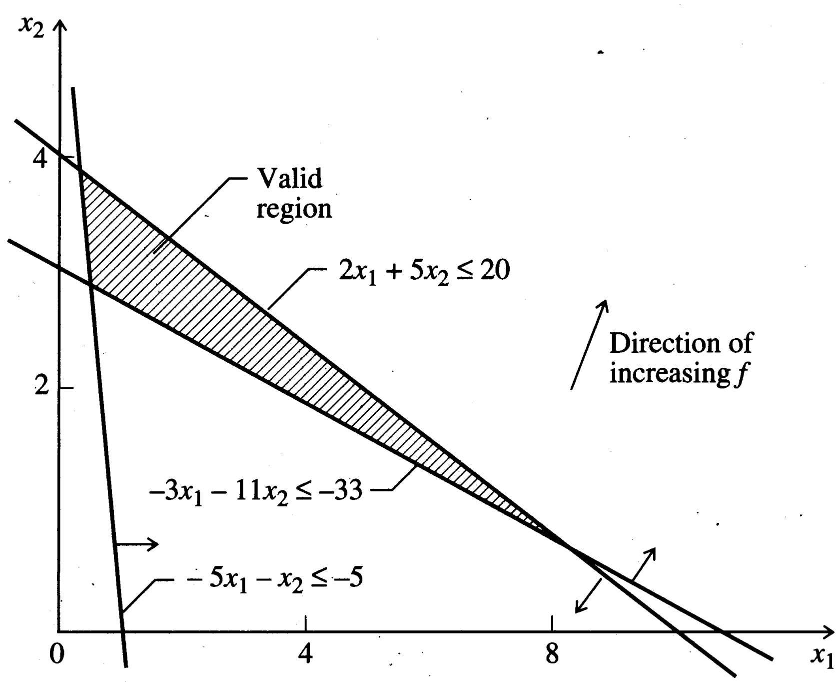

FIGURE P7.19

7.20 Are the following questions true or false and explain why:
(a) In applying the simplex method of linear programming, the solution found, if one is found, is the global solution to the problem.
(b) The solution to a linear programming problem is a unique solution.
(c) The solution to a linear programming problem that includes only inequality constraints (no equality constraints) never occurs in the interior of the feasible region.

7.21 A company has two alkylate plants, A_1 and A_2, from which a given product is distributed to customers C_1, C_2, and C_3. The transportation costs are given as follows:

Refinery	A_1	A_1	A_1	A_2	A_2	A_2
Customer	C_1	C_2	C_3	C_1	C_2	C_3
Cost ($/ton)	25	60	75	20	50	85

The maximum refinery production rates and minimum customer demand rates are fixed and known to be as follows:

Customer or refinery	A_1	A_2	C_1	C_2	C_3
Rate (tons/day)	1.6	0.8	0.9	0.7	0.3

The cost of production for A_1 is \$30/ton for production levels less than 0.5 ton/day: for production levels greater than 0.5 ton/day, the production cost is \$40/ton. A_2's production cost is uniform at \$35/ton.

Find the optimum distribution policy to minimize the company's total costs.

7.22 Alkylate, cat cracked gasoline, and straight run gasoline are blended to make aviation gasolines A and B and two grades of motor gasoline. The specifications on motor gasoline are not as rigid as for aviation gas. Physical property and production data for the inlet streams are as follows:

Stream	RVP	ON(0)	ON(4)	Available (bbl/day)
Alkylate	5	94	108	4000
Cat cracked gasoline	8	84	94	2500
Straight run gasoline	4	74	86	4000

Abbreviations:
RVP = Reid vapor pressure (measure of volatility);
ON = octane number; in parentheses, number of mL/gal of tetraethyl lead (TEL).

For the blended products:

Product	RVP	TEL level	ON	Profit ($/bbl)
Aviation gasoline *A*	≤ 7	0	≥ 80	5.00
Aviation gasoline *B*	≤ 7	4	≥ 91	5.50
Leaded motor gasoline	—	4	≥ 87	4.50
Unleaded motor gasoline	—	0	≥ 91	4.50

Set up this problem as an LP problem, and solve using a standard LP computer code.

7.23 A chemical plant makes three products and uses three raw materials in limited supply as shown in Figure P7.23. Each of the three products is produced in a separate process (1, 2, 3) according to the schematic shown in the figure.

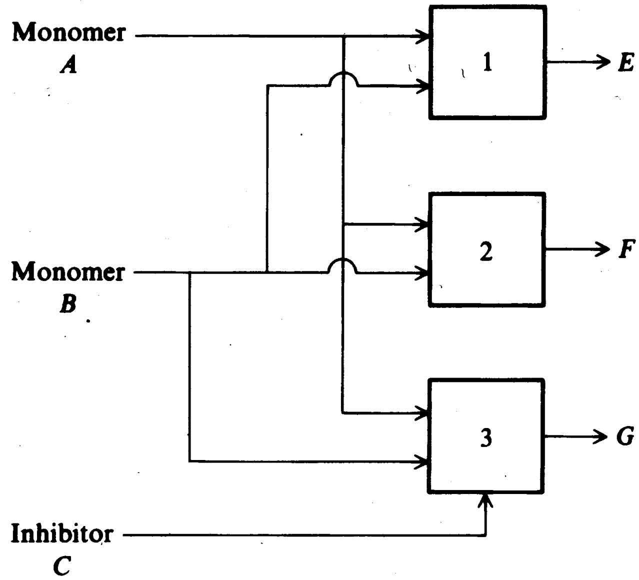

FIGURE P7.23

The available A, B, and C do not have to be totally consumed.

Process data:

Raw material	Maximum available (lb/day)	Cost ($/100 lb)
A	4000	1.50
B	3000	2.00
C	2500	2.50

Process	Product	Reactants needed (lb/lb product)	Operating cost ($)	Selling price of product ($)
1	E	$\frac{2}{3}A, \frac{1}{3}B$	1.00/100 lb A (consumed in 1)	4.00/100 lb E
2	F	$\frac{2}{3}A, \frac{1}{3}B$	0.50/100 lb A (consumed in 2)	3.30/100 lb F
3	G	$\frac{1}{2}A, \frac{1}{6}B, \frac{1}{3}C$	1.00/100 lb G (produced in 3)	3.80/100 lb G

Set up the linear profit function and linear constraints to find the optimum product distribution, and apply the simplex technique to obtain numerical answers.

7.24 Ten grades of crude are available in the quantities shown in the table ranging from 10,000 to 30,000 barrels per day each, with an aggregate availability of 200,000 barrels per day. Refineries X, Y, and Z have incremental operations with stated requirements totaling 180,000 barrels per day. Of the available crude, 20,000 barrels per day is not used. One of the refineries can operate at two incremental operations, X_1 and X_2, which represent different efficiency levels. The net profit or loss for each crude in each refinery operation is given in the table in cents per barrel. It is assumed that the crude evaluations reflect the resulting product distribution from these incremental operations. (In practice, however, if further debits are encountered in the solution because of lack of product quality or for transportation of surplus products, suitable corrections can be made in the crude evaluations and the problem reworked until a realistic solution is obtained.)

Maximize the profit per day by allocating the ten crudes among the three refineries with X being able to operate at two levels, so specify X_1 and X_2 as well as Y and Z.

Crude evaluation, availability and requirement

Crude	*a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	Required (M bpd)
	(Profit or loss of each refinery cpb)										
Refinery											
X_1	−6	3	17	10	63	34	15	22	−2	15	**30**
X_2	−11	−7	−16	9	49	16	4	10	−8	8	**40**
Y	−7	3	16	13	60	25	12	19	4	13	**50**
Z	−1	0	13	3	48	15	7	17	9	3	**60**
Available (M bpd)	**30**	**30**	**20**	**20**	**10**	**20**	**20**	**10**	**30**	**10**	**200**

Abbreviations: M = 1000; bpd = barrels per day; cpb = cents per barrel.

7.25 Consider a typical linear programming example in which N grades of paper are produced on a paper machine. Due to raw materials restrictions not more than a_i tons of grade i can be produced in a week. Let

x_i = numbers of tons of grade i produced during the week
b_i = number of hours required to produce a ton of grade i
p_i = profit made per ton of grade i

Because 160 production hours are available each week, the problem is to find non-negative values of x_i, $i = 1, \ldots, N$, and the integer value N that satisfy

$$x_i \leq a_i \tag{1}$$

$$\sum_{i=1}^{N} b_i x_i \leq c \tag{2}$$

and that maximize the profit function

$$f(x_1, \ldots, x_N) = \sum_{i=1}^{N} p_i x_i \tag{3}$$

Data:

	a_i	b_i	p_i
1	400	0.2	20
2	300	0.4	50
3	200	0.2	20
4	100	0.2	10
5	50	0.2	10

$c = 160$