7

Other Algorithms for Linear Programming

MPL

MPL

The key to the extremely widespread use of linear programming is the availability of an exceptionally efficient algorithm—the simplex method—that will routinely solve the large-size problems that typically arise in practice. However, the simplex method is only part of the arsenal of algorithms regularly used by linear programming practitioners. We now turn to these other algorithms.

This chapter focuses first on three particularly important algorithms that are, in fact, *variants* of the simplex method. In particular, the next three sections present the *dual simplex method* (a modification particularly useful for sensitivity analysis), *parametric linear programming* (an extension for systematic sensitivity analysis), and the *upper bound technique* (a streamlined version of the simplex method for dealing with variables having upper bounds).

Section 4.9 introduced another algorithmic approach to linear programming—a type of algorithm that moves through the interior of the feasible region. We describe this *interior-point approach* further in Sec. 7.4.

We next introduce *linear goal programming* where, rather than having a *single objective* (maximize or minimize Z) as for linear programming, the problem instead has *several goals* toward which we must strive simultaneously. Certain formulation techniques enable converting a linear goal programming problem back into a linear programming problem so that solution procedures based on the simplex method can still be used. Section 7.5 describes these techniques and procedures.

7.1 THE DUAL SIMPLEX METHOD

The *dual simplex method* is based on the duality theory presented in the first part of Chap. 6. To describe the basic idea behind this method, it is helpful to use some terminology introduced in Tables 6.10 and 6.11 of Sec. 6.3 for describing any pair of complementary basic solutions in the primal and dual problems. In particular, recall that both solutions are said to be *primal feasible* if the primal basic solution is feasible, whereas they are called *dual feasible* if the complementary dual basic solution is feasible for the dual problem. Also recall (as indicated on the right side of Table 6.11) that each complementary basic solution is optimal for its problem only if it is *both* primal feasible and dual feasible.

The dual simplex method can be thought of as the *mirror image* of the simplex method. The simplex method deals directly with basic solutions in the primal problem that are *primal feasible* but not dual feasible. It then moves toward an optimal solution by striving to achieve dual feasibility as well (the optimality test for the simplex method). By contrast, the dual simplex method deals with basic solutions in the primal problem that are *dual feasible* but not primal feasible. It then moves toward an optimal solution by striving to achieve primal feasibility as well.

Furthermore, the dual simplex method deals with a problem as if the simplex method were being applied simultaneously to its dual problem. If we make their *initial* basic solutions *complementary,* the two methods move in complete sequence, obtaining *complementary* basic solutions with each iteration.

The dual simplex method is very useful in certain special types of situations. Ordinarily it is easier to find an initial basic solution that is feasible than one that is dual feasible. However, it is occasionally necessary to introduce many *artificial* variables to construct an initial BF solution artificially. In such cases it may be easier to begin with a dual feasible basic solution and use the dual simplex method. Furthermore, fewer iterations may be required when it is not necessary to drive many artificial variables to zero.

As we mentioned several times in Chap. 6 as well as in Sec. 4.7, another important primary application of the dual simplex method is its use in conjunction with sensitivity analysis. Suppose that an optimal solution has been obtained by the simplex method but that it becomes necessary (or of interest for sensitivity analysis) to make minor changes in the model. If the formerly optimal basic solution is *no longer primal feasible* (but still satisfies the optimality test), you can immediately apply the dual simplex method by starting with this *dual feasible* basic solution. Applying the dual simplex method in this way usually leads to the new optimal solution much more quickly than would solving the new problem from the beginning with the simplex method.

The dual simplex method also can be useful in solving huge linear programming problems from scratch because it is such an efficient algorithm. Recent computational experience with the latest versions of CPLEX indicates that the dual simplex method often is more efficient than the simplex method for solving particularly massive problems encountered in practice.

The rules for the dual simplex method are very similar to those for the simplex method. In fact, once the methods are started, the only difference between them is in the criteria used for selecting the entering and leaving basic variables and for stopping the algorithm.

To start the dual simplex method (for a maximization problem), we must have all the coefficients in Eq. (0) *nonnegative* (so that the basic solution is dual feasible). The basic solutions will be infeasible (except for the last one) only because some of the variables are negative. The method continues to decrease the value of the objective function, always retaining *nonnegative coefficients* in Eq. (0), until all the *variables* are nonnegative. Such a basic solution is feasible (it satisfies all the equations) and is, therefore, optimal by the simplex method criterion of nonnegative coefficients in Eq. (0).

The details of the dual simplex method are summarized next.

Summary of the Dual Simplex Method.

1. *Initialization:* After converting any functional constraints in $\geq$ form to $\leq$ form (by multiplying through both sides by -1), introduce slack variables as needed to construct a set of equations describing the problem. Find a basic solution such that the coefficients in Eq. (0) are zero for basic variables and nonnegative for nonbasic variables (so the solution is optimal if it is feasible). Go to the feasibility test.

2. *Feasibility test:* Check to see whether all the basic variables are *nonnegative.* If they are, then this solution is feasible, and therefore optimal, so stop. Otherwise, go to an iteration.

3. *Iteration:*

Step 1 Determine the *leaving basic variable:* Select the *negative* basic variable that has the largest absolute value.

Step 2 Determine the *entering basic variable:* Select the nonbasic variable whose coefficient in Eq. (0) reaches zero first as an increasing multiple of the equation containing the leaving basic variable is added to Eq. (0). This selection is made by checking the nonbasic variables with *negative coefficients* in that equation (the one containing the leaving basic variable) and selecting the one with the smallest absolute value of the ratio of the Eq. (0) coefficient to the coefficient in that equation.

Step 3 Determine the *new basic solution:* Starting from the current set of equations, solve for the basic variables in terms of the nonbasic variables by Gaussian elimination. When we set the nonbasic variables equal to zero, each basic variable (and Z) equals the new right-hand side of the one equation in which it appears (with a coefficient of $+1$). Return to the feasibility test.

To fully understand the dual simplex method, you must realize that the method proceeds just as if the *simplex method* were being applied to the complementary basic solutions in the *dual problem.* (In fact, this interpretation was the motivation for constructing the method as it is.) Step 1 of an iteration, determining the leaving basic variable, is equivalent to determining the entering basic variable in the dual problem. The negative variable with the largest absolute value corresponds to the negative coefficient with the largest absolute value in Eq. (0) of the dual problem (see Table 6.3). Step 2, determining the entering basic variable, is equivalent to determining the leaving basic variable in the dual problem. The coefficient in Eq. (0) that reaches zero first corresponds to the variable in the dual problem that reaches zero first. The two criteria for stopping the algorithm are also complementary.

We shall now illustrate the dual simplex method by applying it to the *dual problem* for the Wyndor Glass Co. (see Table 6.1). Normally this method is applied directly to the problem of concern (a primal problem). However, we have chosen this problem because you have already seen the simplex method applied to *its* dual problem (namely, the primal problem[1]) in Table 4.8 so you can compare the two. To facilitate the comparison, we shall continue to denote the decision variables in the problem being solved by y_i rather than x_j.

In *maximization* form, the problem to be solved is

$$\text{Maximize} \quad Z = -4y_1 - 12y_2 - 18y_3,$$

subject to

$$\begin{aligned} y_1 \qquad + 3y_3 &\geq 3 \\ 2y_2 + 2y_3 &\geq 5 \end{aligned}$$

and

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

[1]Recall that the symmetry property in Sec. 6.1 points out that the dual of a dual problem is the original primal problem.

TABLE 7.1 **Dual simplex method applied to the Wyndor Glass Co. dual problem**

Iteration	Basic Variable	Eq.	Coefficient of: Z	y_1	y_2	y_3	y_4	y_5	Right Side
	Z	(0)	1	4	12	18	0	0	0
0	y_4	(1)	0	-1	0	-3	1	0	-3
	y_5	(2)	0	0	-2	-2	0	1	-5
	Z	(0)	1	4	0	6	0	6	-30
1	y_4	(1)	0	-1	0	-3	1	0	-3
	y_2	(2)	0	0	1	1	0	$-\frac{1}{2}$	$\frac{5}{2}$
	Z	(0)	1	2	0	0	2	6	-36
2	y_3	(1)	0	$\frac{1}{3}$	0	1	$-\frac{1}{3}$	0	1
	y_2	(2)	0	$-\frac{1}{3}$	1	0	$\frac{1}{3}$	$-\frac{1}{2}$	$\frac{3}{2}$

Since negative right-hand sides are now allowed, we do not need to introduce artificial variables to be the initial basic variables. Instead, we simply convert the functional constraints to $\leq$ form and introduce slack variables to play this role. The resulting initial set of equations is that shown for iteration 0 in Table 7.1. Notice that all the coefficients in Eq. (0) are nonnegative, so the solution is optimal if it is feasible.

The initial basic solution is $y_1 = 0$, $y_2 = 0$, $y_3 = 0$, $y_4 = -3$, $y_5 = -5$, with $Z = 0$, which is not feasible because of the negative values. The leaving basic variable is y_5 $(5 > 3)$, and the entering basic variable is y_2 $(12/2 < 18/2)$, which leads to the second set of equations, labeled as iteration 1 in Table 7.1. The corresponding basic solution is $y_1 = 0$, $y_2 = \frac{5}{2}$, $y_3 = 0$, $y_4 = -3$, $y_5 = 0$, with $Z = -30$, which is not feasible.

The next leaving basic variable is y_4, and the entering basic variable is y_3 $(6/3 < 4/1)$, which leads to the final set of equations in Table 7.1. The corresponding basic solution is $y_1 = 0$, $y_2 = \frac{3}{2}$, $y_3 = 1$, $y_4 = 0$, $y_5 = 0$, with $Z = -36$, which is feasible and therefore optimal.

Notice that the optimal solution for the dual of this problem[1] is $x_1^* = 2$, $x_2^* = 6$, $x_3^* = 2$, $x_4^* = 0$, $x_5^* = 0$, as was obtained in Table 4.8 by the simplex method. We suggest that you now trace through Tables 7.1 and 4.8 simultaneously and compare the complementary steps for the two mirror-image methods.

7.2 PARAMETRIC LINEAR PROGRAMMING

At the end of Sec. 6.7 we described *parametric linear programming* and its use for conducting sensitivity analysis systematically by gradually changing various model parameters simultaneously. We shall now present the algorithmic procedure, first for the case where the c_j parameters are being changed and then where the b_i parameters are varied.

[1]The *complementary optimal basic solutions property* presented in Sec. 6.3 indicates how to read the optimal solution for the dual problem from row 0 of the final simplex tableau for the primal problem. This same conclusion holds regardless of whether the simplex method or the dual simplex method is used to obtain the final tableau.

Systematic Changes in the c_j Parameters

For the case where the c_j parameters are being changed, the *objective function* of the ordinary linear programming model

$$Z = \sum_{j=1}^{n} c_j x_j$$

is replaced by

$$Z(\theta) = \sum_{j=1}^{n} (c_{j+} \alpha_j \theta) x_j,$$

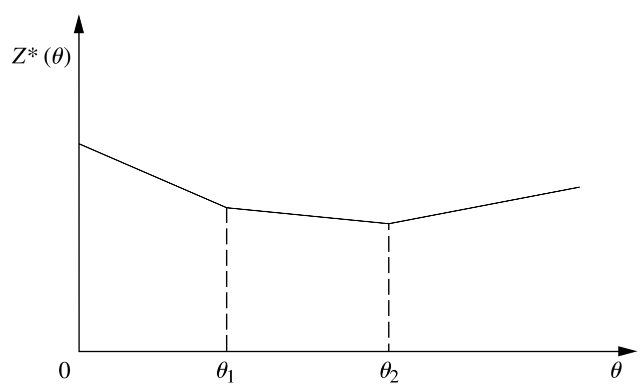
where the α_j are given input constants representing the *relative* rates at which the coefficients are to be changed. Therefore, gradually increasing θ from zero changes the coefficients at these relative rates.

The values assigned to the α_j may represent interesting simultaneous changes of the c_j for systematic sensitivity analysis of the effect of increasing the magnitude of these changes. They may also be based on how the coefficients (e.g., unit profits) would change together with respect to some factor measured by θ. This factor might be uncontrollable, e.g., the state of the economy. However, it may also be under the control of the decision maker, e.g., the amount of personnel and equipment to shift from some of the activities to others.

For any given value of θ, the optimal solution of the corresponding linear programming problem can be obtained by the simplex method. This solution may have been obtained already for the original problem where $\theta = 0$. However, the objective is to *find the optimal solution* of the modified linear programming problem [maximize $Z(\theta)$ subject to the original constraints] *as a function of* θ. Therefore, in the solution procedure you need to be able to determine when and how the optimal solution changes (if it does) as θ increases from zero to any specified positive number.

Figure 7.1 illustrates how $Z^*(\theta)$, the objective function value for the optimal solution (given θ), changes as θ increases. In fact, $Z^*(\theta)$ always has this *piecewise linear* and *con-*

FIGURE 7.1
The objective function value for an optimal solution as a function of θ for parametric linear programming with systematic changes in the c_j parameters.



vex[1] form (see Prob. 7.2-7). The corresponding optimal solution changes (as θ increases) *just* at the values of θ where the slope of the $Z^*(\theta)$ function changes. Thus, Fig. 7.1 depicts a problem where three different solutions are optimal for different values of θ, the first for $0 \le \theta \le \theta_1$, the second for $\theta_1 \le \theta \le \theta_2$, and the third for $\theta \ge \theta_2$. Because the value of each x_j remains the same within each of these intervals for θ, the value of $Z^*(\theta)$ varies with θ only because the *coefficients* of the x_j are changing as a linear function of θ. The solution procedure is based directly upon the sensitivity analysis procedure for investigating changes in the c_j parameters (Cases 2*a* and 3, Sec. 6.7). As described in the last subsection of Sec. 6.7, the only basic difference with parametric linear programming is that the changes now are expressed in terms of θ rather than as specific numbers.

To illustrate, suppose that $\alpha_1 = 2$ and $\alpha_2 = -1$ for the original Wyndor Glass Co. problem presented in Sec. 3.1, so that

$$Z(\theta) = (3 + 2\theta)x_1 + (5 - \theta)x_2.$$

Beginning with the final simplex tableau for $\theta = 0$ (Table 4.8), we see that its Eq. (0)

$$(0) \qquad Z + \frac{3}{2}x_4 + x_5 = 36$$

would first have these changes from the original ($\theta = 0$) coefficients added into it on the left-hand side:

$$(0) \qquad Z - 2\theta x_1 + \theta x_2 + \frac{3}{2}x_4 + x_5 = 36.$$

Because both x_1 and x_2 are basic variables [appearing in Eqs. (3) and (2), respectively], they both need to be eliminated algebraically from Eq. (0):

$$\begin{array}{ll} & Z - 2\theta x_1 + \theta x_2 + \frac{3}{2}x_4 + x_5 = 36 \\ & \qquad + 2\theta \text{ times Eq. (3)} \\ & \qquad - \ \ \theta \text{ times Eq. (2)} \\ \hline (0) & Z + \left(\frac{3}{2} - \frac{7}{6}\theta\right)x_4 + \left(1 + \frac{2}{3}\theta\right)x_5 = 36 - 2\theta. \end{array}$$

The optimality test says that the current BF solution will remain optimal as long as these coefficients of the nonbasic variables remain nonnegative:

$$\frac{3}{2} - \frac{7}{6}\theta \ge 0, \qquad \text{for } 0 \le \theta \le \frac{9}{7},$$

$$1 + \frac{2}{3}\theta \ge 0, \qquad \text{for all } \theta \ge 0.$$

Therefore, after θ is increased past $\theta = \frac{9}{7}$, x_4 would need to be the entering basic variable for another iteration of the simplex method to find the new optimal solution. Then θ would be increased further until another coefficient goes negative, and so on until θ has been increased as far as desired.

This entire procedure is now summarized, and the example is completed in Table 7.2.

[1]See Appendix 2 for a definition and discussion of convex functions.

TABLE 7.2 The c_j parametric linear programming procedure applied to the Wyndor Glass Co. example

Range of θ	Basic Variable	Eq.	Z	x_1	x_2	x_3	x_4	x_5	Right Side	Optimal Solution
$0 \le \theta \le \frac{9}{7}$	$Z(\theta)$	(0)	1	0	0	0	$\frac{9-7\theta}{6}$	$\frac{3+2\theta}{3}$	$36-2\theta$	$x_4 = 0$, $x_5 = 0$
	x_3	(1)	0	0	0	1	$\frac{1}{3}$	$-\frac{1}{3}$	2	$x_3 = 2$
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	6	$x_2 = 6$
	x_1	(3)	0	1	0	0	$-\frac{1}{3}$	$\frac{1}{3}$	2	$x_1 = 2$
$\frac{9}{7} \le \theta \le 5$	$Z(\theta)$	(0)	1	0	0	$\frac{-9+7\theta}{2}$	0	$\frac{5-\theta}{2}$	$27+5\theta$	$x_3 = 0$, $x_5 = 0$
	x_4	(1)	0	0	0	3	1	-1	6	$x_4 = 6$
	x_2	(2)	0	0	1	$-\frac{3}{2}$	0	$\frac{1}{2}$	3	$x_2 = 3$
	x_1	(3)	0	1	0	1	0	0	4	$x_1 = 4$
$\theta \ge 5$	$Z(\theta)$	(0)	1	0	$-5+\theta$	$3+2\theta$	0	0	$12+8\theta$	$x_2 = 0$, $x_3 = 0$
	x_4	(1)	0	0	2	0	1	0	12	$x_4 = 12$
	x_5	(2)	0	0	2	-3	0	1	6	$x_5 = 6$
	x_1	(3)	0	1	0	1	0	0	4	$x_1 = 4$

Summary of the Parametric Linear Programming Procedure for Systematic Changes in the c_j Parameters.

1. Solve the problem with $\theta = 0$ by the simplex method.
2. Use the sensitivity analysis procedure (Cases 2*a* and 3, Sec. 6.7) to introduce the $\Delta c_j = \alpha_j\theta$ changes into Eq. (0).
3. Increase θ until one of the nonbasic variables has its coefficient in Eq. (0) go negative (or until θ has been increased as far as desired).
4. Use this variable as the entering basic variable for an iteration of the simplex method to find the new optimal solution. Return to step 3.

Systematic Changes in the b_i Parameters

For the case where the b_i parameters change systematically, the one modification made in the original linear programming model is that b_i is replaced by $b_i + \alpha_i\theta$, for $i = 1, 2, \ldots, m$, where the α_i are given input constants. Thus, the problem becomes

$$\text{Maximize} \qquad Z(\theta) = \sum_{j=1}^{n} c_j x_j,$$

subject to

$$\sum_{j=1}^{n} a_{ij}x_j \leq b_i + \alpha_i\theta \qquad \text{for } i = 1, 2, \ldots, m$$

and

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

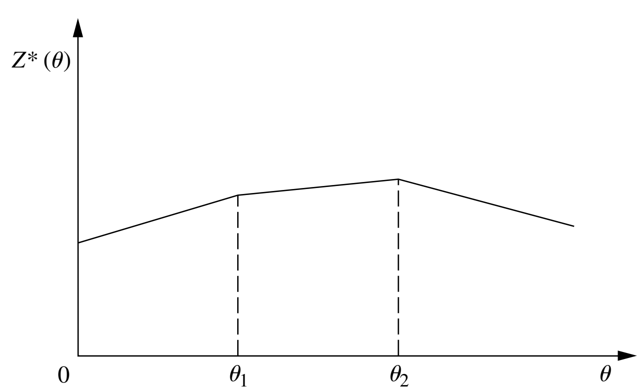
The goal is to identify the optimal solution as a function of θ.

With this formulation, the corresponding objective function value $Z^*(\theta)$ always has the *piecewise linear* and *concave*[1] form shown in Fig. 7.2. (See Prob. 7.2-8.) The set of basic variables in the optimal solution still changes (as θ increases) *only* where the slope of $Z^*(\theta)$ changes. However, in contrast to the preceding case, the values of these variables now change as a (linear) function of θ between the slope changes. The reason is that increasing θ changes the right-hand sides in the initial set of equations, which then causes changes in the right-hand sides in the final set of equations, i.e., in the values of the final set of basic variables. Figure 7.2 depicts a problem with three sets of basic variables that are optimal for different values of θ, the first for $0 \leq \theta \leq \theta_1$, the second for $\theta_1 \leq \theta \leq \theta_2$, and the third for $\theta \geq \theta_2$. Within each of these intervals of θ, the value of $Z^*(\theta)$ varies with θ despite the fixed coefficients c_j because the x_j values are changing.

The following solution procedure summary is very similar to that just presented for systematic changes in the c_j parameters. The reason is that changing the b_i values is equivalent to changing the coefficients in the objective function of the *dual* model. Therefore, the procedure for the primal problem is exactly *complementary* to applying simultaneously the procedure for systematic changes in the c_j parameters to the *dual* problem. Consequently, the *dual simplex method* (see Sec. 7.1) now would be used to obtain each new optimal solution, and the applicable sensitivity analysis case (see Sec. 6.7) now is Case 1, but these differences are the only major differences.

[1]See Appendix 2 for a definition and discussion of concave functions.

FIGURE 7.2
The objective function value for an optimal solution as a function of θ for parametric linear programming with systematic changes in the b_i parameters.



Summary of the Parametric Linear Programming Procedure for Systematic Changes in the b_i Parameters.

1. Solve the problem with $\theta = 0$ by the simplex method.
2. Use the sensitivity analysis procedure (Case 1, Sec. 6.7) to introduce the $\Delta b_i = \alpha_i\theta$ changes to the *right side* column.
3. Increase θ until one of the basic variables has its value in the *right side* column go negative (or until θ has been increased as far as desired).
4. Use this variable as the leaving basic variable for an iteration of the dual simplex method to find the new optimal solution. Return to step 3.

To illustrate this procedure in a way that demonstrates its *duality* relationship with the procedure for systematic changes in the c_j parameters, we now apply it to the dual problem for the Wyndor Glass Co. (see Table 6.1). In particular, suppose that $\alpha_1 = 2$ and $\alpha_2 = -1$ so that the functional constraints become

$$\begin{array}{llll} y_1 \quad\;\; + 3y_3 \geq 3 + 2\theta & \text{or} & -y_1 \quad\;\; - 3y_3 \leq -3 - 2\theta \\ \quad 2y_2 + 2y_3 \geq 5 - \theta & \text{or} & \quad -2y_2 - 2y_3 \leq -5 + \theta. \end{array}$$

Thus, the dual of *this* problem is just the example considered in Table 7.2.

This problem with $\theta = 0$ has already been solved in Table 7.1, so we begin with the final simplex tableau given there. Using the sensitivity analysis procedure for Case 1, Sec. 6.7, we find that the entries in the *right side* column of the tableau change to the values given below.

$$Z^* = \mathbf{y}^*\bar{\mathbf{b}} = [2, 6]\begin{bmatrix} -3 - 2\theta \\ -5 + \theta \end{bmatrix} = -36 + 2\theta,$$

$$\mathbf{b}^* = \mathbf{S}^*\bar{\mathbf{b}} = \begin{bmatrix} -\frac{1}{3} & 0 \\ \frac{1}{3} & -\frac{1}{2} \end{bmatrix}\begin{bmatrix} -3 - 2\theta \\ -5 + \theta \end{bmatrix} = \begin{bmatrix} 1 + \frac{2\theta}{3} \\ \frac{3}{2} - \frac{7\theta}{6} \end{bmatrix}.$$

Therefore, the two basic variables in this tableau

$$y_3 = \frac{3 + 2\theta}{3} \qquad \text{and} \qquad y_2 = \frac{9 - 7\theta}{6}$$

remain nonnegative for $0 \leq \theta \leq \frac{9}{7}$. Increasing θ past $\theta = \frac{9}{7}$ requires making y_2 a leaving basic variable for another iteration of the dual simplex method, and so on, as summarized in Table 7.3.

We suggest that you now trace through Tables 7.2 and 7.3 simultaneously to note the duality relationship between the two procedures.

7.3 THE UPPER BOUND TECHNIQUE

It is fairly common in linear programming problems for some of or all the *individual* x_j variables to have *upper bound constraints*

$$x_j \leq u_j,$$

where u_j is a positive constant representing the maximum *feasible* value of x_j. We pointed out in Sec. 4.8 that the most important determinant of computation time for the simplex

TABLE 7.3 The b_i parametric linear programming procedure applied to the dual of the Wyndor Glass Co. example

Range of θ	Basic Variable	Eq.	Z	y_1	y_2	y_3	y_4	y_5	Right Side	Optimal Solution
$0 \le \theta \le \frac{9}{7}$	$Z(\theta)$	(0)	1	2	0	0	2	6	$-36 + 2\theta$	$y_1 = y_4 = y_5 = 0$
	y_3	(1)	0	$\frac{1}{3}$	0	1	$-\frac{1}{3}$	0	$\frac{3 + 2\theta}{3}$	$y_3 = \frac{3 + 2\theta}{3}$
	y_2	(2)	0	$-\frac{1}{3}$	1	0	$\frac{1}{3}$	$-\frac{1}{2}$	$\frac{9 - 7\theta}{6}$	$y_2 = \frac{9 - 7\theta}{6}$
$\frac{9}{7} \le \theta \le 5$	$Z(\theta)$	(0)	1	0	6	0	4	3	$-27 - 5\theta$	$y_2 = y_4 = y_5 = 0$
	y_3	(1)	0	0	1	1	0	$-\frac{1}{2}$	$\frac{5 - \theta}{2}$	$y_3 = \frac{5 - \theta}{2}$
	y_1	(2)	0	1	-3	0	-1	$\frac{3}{2}$	$\frac{-9 + 7\theta}{2}$	$y_1 = \frac{-9 + 7\theta}{2}$
$\theta \ge 5$	$Z(\theta)$	(0)	1	0	12	6	4	0	$-12 - 8\theta$	$y_2 = y_3 = y_4 = 0$
	y_5	(1)	0	0	-2	-2	0	1	$-5 + \theta$	$y_5 = -5 + \theta$
	y_1	(2)	0	1	0	3	-1	0	$3 + 2\theta$	$y_1 = 3 + 2\theta$

method is the *number of functional constraints,* whereas the number of *nonnegativity* constraints is relatively unimportant. Therefore, having a large number of upper bound constraints among the functional constraints greatly increases the computational effort required.

The *upper bound technique* avoids this increased effort by removing the upper bound constraints from the functional constraints and treating them separately, essentially like nonnegativity constraints. Removing the upper bound constraints in this way causes no problems as long as none of the variables gets increased over its upper bound. The only time the simplex method increases some of the variables is when the entering basic variable is increased to obtain a new BF solution. Therefore, the upper bound technique simply applies the simplex method in the usual way to the *remainder* of the problem (i.e., without the upper bound constraints) but with the one additional restriction that each new BF solution must satisfy the upper bound constraints in addition to the usual lower bound (nonnegativity) constraints.

To implement this idea, note that a decision variable x_j with an upper bound constraint $x_j \le u_j$ can always be replaced by

$$x_j = u_j - y_j,$$

where y_j would then be the decision variable. In other words, you have a choice between letting the decision variable be the *amount above zero* (x_j) or the *amount below* u_j ($y_j = u_j - x_j$). (We shall refer to x_j and y_j as *complementary* decision variables.) Because

$$0 \le x_j \le u_j$$

it also follows that

$$0 \le y_j \le u_j.$$

Thus, at any point during the simplex method, you can either

1. Use x_j, where $0 \le x_j \le u_j$,
or **2.** Replace x_j by $u_j - y_j$, where $0 \le y_j \le u_j$.

The upper bound technique uses the following rule to make this choice:

Rule: Begin with choice 1.
Whenever $x_j = 0$, use choice 1, so x_j is *nonbasic*.
Whenever $x_j = u_j$, use choice 2, so $y_j = 0$ is *nonbasic*.
Switch choices only when the other extreme value of x_j is reached.

Therefore, whenever a basic variable reaches its upper bound, you should switch choices and use its complementary decision variable as the new nonbasic variable (the leaving basic variable) for identifying the new BF solution. Thus, the one substantive modification being made in the simplex method is in the rule for selecting the leaving basic variable.

Recall that the simplex method selects as the leaving basic variable the one that would be the first to become infeasible by going negative as the entering basic variable is increased. The modification now made is to select instead the variable that would be the first to become infeasible *in any way,* either by going negative or by going over the upper bound, as the entering basic variable is increased. (Notice that one possibility is that the entering basic variable may become infeasible first by going over its upper bound, so that its complementary decision variable becomes the leaving basic variable.) If the leaving basic variable reaches zero, then proceed as usual with the simplex method. However, if it reaches its upper bound instead, then switch choices and make its complementary decision variable the leaving basic variable.

To illustrate, consider this problem:

$$\text{Maximize} \qquad Z = 2x_1 + x_2 + 2x_3,$$

subject to

$$\begin{aligned} 4x_1 + x_2 &= 12 \\ -2x_1 + x_3 &= 4 \end{aligned}$$

and

$$0 \le x_1 \le 4, \qquad 0 \le x_2 \le 15, \qquad 0 \le x_3 \le 6.$$

Thus, all three variables have upper bound constraints ($u_1 = 4$, $u_2 = 15$, $u_3 = 6$).

The two equality constraints are already in proper form from Gaussian elimination for identifying the initial BF solution ($x_1 = 0$, $x_2 = 12$, $x_3 = 4$), and none of the variables in this solution exceeds its upper bound, so x_2 and x_3 can be used as the initial basic variables without artificial variables being introduced. However, these variables then need to be eliminated algebraically from the objective function to obtain the initial Eq. (0), as follows:

$$\begin{array}{llrrrl} & Z & - \;2x_1 & - \;x_2 & - \;2x_3 & = \;\;0 \\ & & + (4x_1 & + \;x_2 & & = 12) \\ & & + 2(-\;2x_1 & & + \;\;x_3 & = \;\;4) \\ \hline (0) & Z & - \;2x_1 & & & = 20. \end{array}$$

TABLE 7.4 **Equations and calculations for the initial leaving basic variable in the example for the upper bound technique**

Initial Set of Equations	Maximum Feasible Value of x_1
(0) $Z - 2x_1 \qquad = 20$	$x_1 \le 4$ (since $u_1 = 4$)
(1) $4x_1 + x_2 \qquad = 12$	$x_1 \le \frac{12}{4} = 3$
(2) $-2x_1 \qquad + x_3 = 4$	$x_1 \le \frac{6-4}{2} = 1 \leftarrow$ minimum (because $u_3 = 6$)

To start the first iteration, this initial Eq. (0) indicates that the initial *entering* basic variable is x_1. Since the upper bound constraints are not to be included, the entire initial set of equations and the corresponding calculations for selecting the leaving basic variables are those shown in Table 7.4. The second column shows how much the entering basic variable x_1 can be *increased* from zero before some basic variable (including x_1) becomes infeasible. The maximum value given next to Eq. (0) is just the upper bound constraint for x_1. For Eq. (1), since the coefficient of x_1 is *positive, increasing* x_1 to 3 decreases the basic variable in this equation (x_2) from 12 to its *lower* bound of *zero*. For Eq. (2), since the coefficient of x_1 is *negative, increasing* x_1 to 1 *increases* the basic variable in this equation (x_3) from 4 to its *upper bound* of 6.

Because Eq. (2) has the *smallest* maximum feasible value of x_1 in Table 7.4, the basic variable in this equation (x_3) provides the *leaving* basic variable. However, because x_3 reached its *upper* bound, replace x_3 by $6 - y_3$, so that $y_3 = 0$ becomes the new nonbasic variable for the next BF solution and x_1 becomes the new basic variable in Eq. (2). This replacement leads to the following changes in this equation:

$$
\begin{aligned}
(2) \qquad & -2x_1 + x_3 = 4 \\
\rightarrow \; & -2x_1 + 6 - y_3 = 4 \\
\rightarrow \; & -2x_1 - y_3 = -2 \\
\rightarrow \; & x_1 + \frac{1}{2}y_3 = 1
\end{aligned}
$$

Therefore, after we eliminate x_1 algebraically from the other equations, the *second* complete set of equations becomes

$$
\begin{aligned}
(0) \qquad & Z \qquad + y_3 = 22 \\
(1) \qquad & x_2 - 2y_3 = 8 \\
(2) \qquad & x_1 \qquad + \frac{1}{2}y_3 = 1.
\end{aligned}
$$

The resulting BF solution is $x_1 = 1$, $x_2 = 8$, $y_3 = 0$. By the optimality test, it also is an optimal solution, so $x_1 = 1$, $x_2 = 8$, $x_3 = 6 - y_3 = 6$ is the desired solution for the original problem.

7.4 AN INTERIOR-POINT ALGORITHM

In Sec. 4.9 we discussed a dramatic development in linear programming that occurred in 1984, the invention by Narendra Karmarkar of AT&T Bell Laboratories of a powerful algorithm for solving huge linear programming problems with an approach very different from the simplex method. We now introduce the nature of Karmarkar's approach by describing a relatively elementary variant (the "affine" or "affine-scaling" variant) of his algorithm.[1] (Your OR Courseware also includes this variant under the title, *Solve Automatically by the Interior-Point Algorithm.*)

Throughout this section we shall focus on Karmarkar's main ideas on an intuitive level while avoiding mathematical details. In particular, we shall bypass certain details that are needed for the full implementation of the algorithm (e.g., how to find an initial feasible trial solution) but are not central to a basic conceptual understanding. The ideas to be described can be summarized as follows:

Concept 1: Shoot through the *interior* of the feasible region toward an optimal solution.
Concept 2: Move in a direction that improves the objective function value at the fastest possible rate.
Concept 3: Transform the feasible region to place the current trial solution near its center, thereby enabling a large improvement when concept 2 is implemented.

To illustrate these ideas throughout the section, we shall use the following example:

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

subject to

$$x_1 + x_2 \leq 8$$

and

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

This problem is depicted graphically in Fig. 7.3, where the optimal solution is seen to be $(x_1, x_2) = (0, 8)$ with $Z = 16$.

The Relevance of the Gradient for Concepts 1 and 2

The algorithm begins with an initial trial solution that (like all subsequent trial solutions) lies in the *interior* of the feasible region, i.e., *inside the boundary* of the feasible region. Thus, for the example, the solution must not lie on any of the three lines ($x_1 = 0$, $x_2 = 0$, $x_1 + x_2 = 8$) that form the boundary of this region in Fig. 7.3. (A trial solution that lies on the boundary cannot be used because this would lead to the undefined mathematical operation of division by zero at one point in the algorithm.) We have arbitrarily chosen $(x_1, x_2) = (2, 2)$ to be the initial trial solution.

To begin implementing concepts 1 and 2, note in Fig. 7.3 that the direction of movement from (2, 2) that increases Z at the fastest possible rate is *perpendicular* to (and toward) the objective function line $Z = 16 = x_1 + 2x_2$. We have shown this direction by the arrow from (2, 2) to (3, 4). Using vector addition, we have

$$(3, 4) = (2, 2) + (1, 2),$$

[1]The basic approach for this variant actually was proposed in 1967 by a Russian mathematician I. I. Dikin and then rediscovered soon after the appearance of Karmarkar's work by a number of researchers, including E. R. Barnes, T. M. Cavalier, and A. L. Soyster. Also see R. J. Vanderbei, M. S. Meketon, and B. A. Freedman, "A Modification of Karmarkar's Linear Programming Algorithm," *Algorithmica,* **1**(4) (Special Issue on New Approaches to Linear Programming): 395–407, 1986.

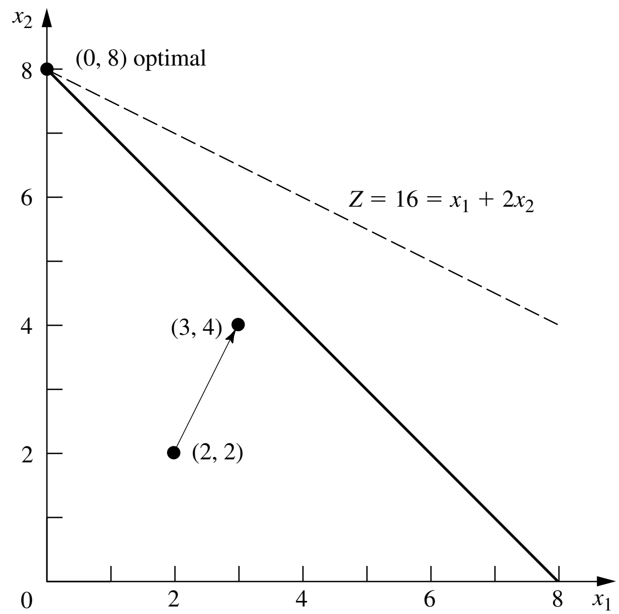


FIGURE 7.3
Example for the interior-point algorithm.

where the vector (1, 2) is the **gradient** of the objective function. (We will discuss gradients further in Sec. 13.5 in the broader context of *nonlinear programming,* where algorithms similar to Karmarkar's have long been used.) The components of (1, 2) are just the coefficients in the objective function. Thus, with one subsequent modification, the gradient (1, 2) defines the ideal direction to which to move, where the question of the *distance to move* will be considered later.

The algorithm actually operates on linear programming problems after they have been rewritten in augmented form. Letting x_3 be the slack variable for the functional constraint of the example, we see that this form is

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

subject to

$$x_1 + x_2 + x_3 = 8$$

and

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

In matrix notation (slightly different from Chap. 5 because the slack variable now is incorporated into the notation), the augmented form can be written in general as

$$\text{Maximize} \quad Z = \mathbf{c}^T\mathbf{x},$$

subject to

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

and

$$\mathbf{x} \geq \mathbf{0},$$

where

$$\mathbf{c} = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}, \qquad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \qquad \mathbf{A} = [1, \quad 1, \quad 1], \qquad \mathbf{b} = [8], \qquad \mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

for the example. Note that $\mathbf{c}^T = [1, 2, 0]$ now is the gradient of the objective function.

The augmented form of the example is depicted graphically in Fig. 7.4. The feasible region now consists of the triangle with vertices (8, 0, 0), (0, 8, 0), and (0, 0, 8). Points in the interior of this feasible region are those where $x_1 > 0$, $x_2 > 0$, and $x_3 > 0$. Each of these three $x_j > 0$ conditions has the effect of forcing (x_1, x_2) away from one of the three lines forming the boundary of the feasible region in Fig. 7.3.

Using the Projected Gradient to Implement Concepts 1 and 2

In augmented form, the initial trial solution for the example is $(x_1, x_2, x_3) = (2, 2, 4)$. Adding the gradient (1, 2, 0) leads to

$$(3, 4, 4) = (2, 2, 4) + (1, 2, 0).$$

However, now there is a complication. The algorithm cannot move from (2, 2, 4) toward (3, 4, 4), because (3, 4, 4) is infeasible! When $x_1 = 3$ and $x_2 = 4$, then $x_3 = 8 - x_1 - x_2 = 1$ instead of 4. The point (3, 4, 4) lies on the near side as you look down on the feasible triangle in Fig. 7.4. Therefore, to remain feasible, the algorithm (indirectly) *projects* the point (3, 4, 4) down onto the feasible triangle by dropping a line that is *perpendicular* to this triangle. A vector from (0, 0, 0) to (1, 1, 1) is perpendicular to this triangle, so the perpendicular line through (3, 4, 4) is given by the equation

$$(x_1, x_2, x_3) = (3, 4, 4) - \theta(1, 1, 1),$$

where θ is a scalar. Since the triangle satisfies the equation $x_1 + x_2 + x_3 = 8$, this perpendicular line intersects the triangle at (2, 3, 3). Because

$$(2, 3, 3) = (2, 2, 4) + (0, 1, -1),$$

the **projected gradient** of the objective function (the gradient projected onto the feasible region) is $(0, 1, -1)$. It is this projected gradient that defines the direction of movement for the algorithm, as shown by the arrow in Fig. 7.4.

A formula is available for computing the projected gradient directly. By defining the *projection matrix* $\mathbf{P}$ as

$$\mathbf{P} = \mathbf{I} - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A},$$

the *projected gradient* (in column form) is

$$\mathbf{c}_p = \mathbf{P}\mathbf{c}.$$

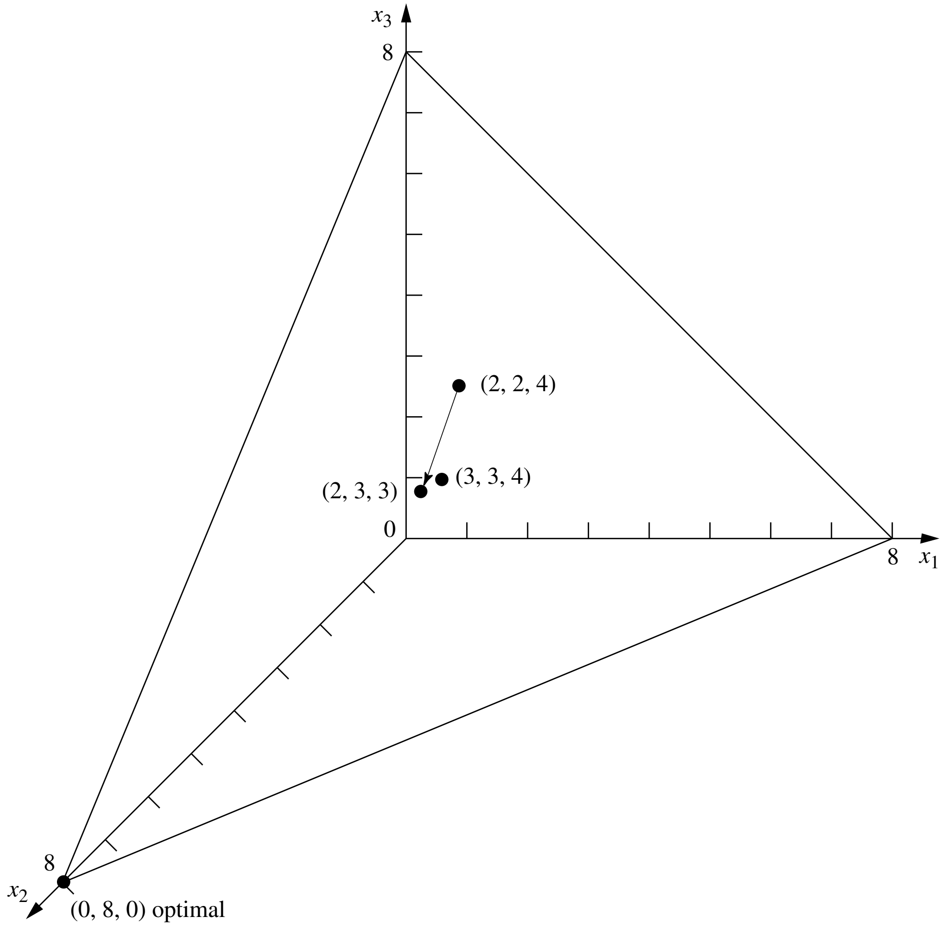


FIGURE 7.4
Example in augmented form for the interior-point algorithm.

Thus, for the example,

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \left([1 \quad 1 \quad 1] \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right)^{-1} [1 \quad 1 \quad 1]$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{1}{3} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} [1 \quad 1 \quad 1]$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3} \end{bmatrix},$$

so

$$\mathbf{c}_p = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}.$$

Moving from (2, 2, 4) in the direction of the projected gradient (0, 1, -1) involves increasing α from zero in the formula

$$\mathbf{x} = \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix} + 4\alpha\mathbf{c}_p = \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix} + 4\alpha \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix},$$

where the coefficient 4 is used simply to give an upper bound of 1 for α to maintain feasibility (all $x_j \geq 0$). Note that increasing α to $\alpha = 1$ would cause x_3 to decrease to $x_3 = 4 + 4(1)(-1) = 0$, where $\alpha > 1$ yields $x_3 < 0$. Thus, α measures the fraction used of the distance that could be moved before the feasible region is left.

How large should α be made for moving to the next trial solution? Because the increase in Z is proportional to α, a value close to the upper bound of 1 is good for giving a relatively large step toward optimality on the current iteration. However, the problem with a value too close to 1 is that the next trial solution then is jammed against a constraint boundary, thereby making it difficult to take large improving steps during subsequent iterations. Therefore, it is very helpful for trial solutions to be near the center of the feasible region (or at least near the center of the portion of the feasible region in the vicinity of an optimal solution), and not too close to any constraint boundary. With this in mind, Karmarkar has stated for his algorithm that a value as large as $\alpha = 0.25$ should be "safe." In practice, much larger values (for example, $\alpha = 0.9$) sometimes are used. For the purposes of this example (and the problems at the end of the chapter), we have chosen $\alpha = 0.5$. (Your OR Courseware uses $\alpha = 0.5$ as the default value, but also has $\alpha = 0.9$ available.)

A Centering Scheme for Implementing Concept 3

We now have just one more step to complete the description of the algorithm, namely, a special scheme for transforming the feasible region to place the current trial solution near its center. We have just described the benefit of having the trial solution near the center, but another important benefit of this centering scheme is that it keeps turning the direction of the projected gradient to point more nearly toward an optimal solution as the algorithm converges toward this solution.

The basic idea of the centering scheme is straightforward—simply change the scale (units) for each of the variables so that the trial solution becomes equidistant from the constraint boundaries in the new coordinate system. (Karmarkar's original algorithm uses a more sophisticated centering scheme.)

For the example, there are three constraint boundaries in Fig. 7.3, each one corresponding to a zero value for one of the three variables of the problem in augmented form, namely, $x_1 = 0$, $x_2 = 0$, and $x_3 = 0$. In Fig. 7.4, see how these three constraint boundaries intersect the $\mathbf{Ax} = \mathbf{b}$ ($x_1 + x_2 + x_3 = 8$) plane to form the boundary of the feasible region. The initial trial solution is $(x_1, x_2, x_3) = (2, 2, 4)$, so this solution is 2 units away from the $x_1 = 0$ and $x_2 = 0$ constraint boundaries and 4 units away from the $x_3 = 0$ constraint boundary, when the units of the respective variables are used. However, whatever these units are in each case, they are quite arbitrary and can be changed as desired without changing the problem. Therefore, let us rescale the variables as follows:

$$\tilde{x}_1 = \frac{x_1}{2}, \qquad \tilde{x}_2 = \frac{x_2}{2}, \qquad \tilde{x}_3 = \frac{x_3}{4}$$

in order to make the current trial solution of $(x_1, x_2, x_3) = (2, 2, 4)$ become

$$(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) = (1, 1, 1).$$

In these new coordinates (substituting $2\tilde{x}_1$ for x_1, $2\tilde{x}_2$ for x_2, and $4\tilde{x}_3$ for x_3), the problem becomes

$$\text{Maximize} \qquad Z = 2\tilde{x}_1 + 4\tilde{x}_2,$$

subject to

$$2\tilde{x}_1 + 2\tilde{x}_2 + 4\tilde{x}_3 = 8$$

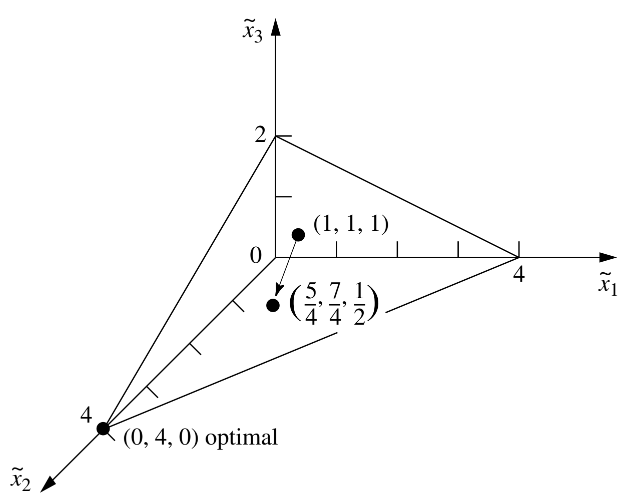
and

$$\tilde{x}_1 \geq 0, \qquad \tilde{x}_2 \geq 0, \qquad \tilde{x}_3 \geq 0,$$

as depicted graphically in Fig. 7.5.

Note that the trial solution (1, 1, 1) in Fig. 7.5 is equidistant from the three constraint boundaries $\tilde{x}_1 = 0$, $\tilde{x}_2 = 0$, $\tilde{x}_3 = 0$. For each subsequent iteration as well, the problem is rescaled again to achieve this same property, so that the current trial solution always is (1, 1, 1) in the current coordinates.

FIGURE 7.5
Example after rescaling for iteration 1.



Summary and Illustration of the Algorithm

Now let us summarize and illustrate the algorithm by going through the first iteration for the example, then giving a summary of the general procedure, and finally applying this summary to a second iteration.

Iteration 1. Given the initial trial solution $(x_1, x_2, x_3) = (2, 2, 4)$, let $\mathbf{D}$ be the corresponding *diagonal matrix* such that $\mathbf{x} = \mathbf{D}\tilde{\mathbf{x}}$, so that

$$\mathbf{D} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix}.$$

The rescaled variables then are the components of

$$\tilde{\mathbf{x}} = \mathbf{D}^{-1}\mathbf{x} = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{4} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \frac{x_1}{2} \\ \frac{x_2}{2} \\ \frac{x_3}{4} \end{bmatrix}.$$

In these new coordinates, $\mathbf{A}$ and $\mathbf{c}$ have become

$$\tilde{\mathbf{A}} = \mathbf{AD} = [1 \quad 1 \quad 1] \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} = [2 \quad 2 \quad 4],$$

$$\tilde{\mathbf{c}} = \mathbf{Dc} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 0 \end{bmatrix}.$$

Therefore, the projection matrix is

$$\begin{aligned} \mathbf{P} &= \mathbf{I} - \tilde{\mathbf{A}}^T(\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^{-1}\tilde{\mathbf{A}} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix} \left([2 \quad 2 \quad 4] \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix} \right)^{-1} [2 \quad 2 \quad 4] \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{1}{24} \begin{bmatrix} 4 & 4 & 8 \\ 4 & 4 & 8 \\ 8 & 8 & 16 \end{bmatrix} = \begin{bmatrix} \frac{5}{6} & -\frac{1}{6} & -\frac{1}{3} \\ -\frac{1}{6} & \frac{5}{6} & -\frac{1}{3} \\ -\frac{1}{3} & -\frac{1}{3} & \frac{1}{3} \end{bmatrix}, \end{aligned}$$

so that the projected gradient is

$$\mathbf{c}_p = \mathbf{P}\tilde{\mathbf{c}} = \begin{bmatrix} \frac{5}{6} & -\frac{1}{6} & -\frac{1}{3} \\ -\frac{1}{6} & \frac{5}{6} & -\frac{1}{3} \\ -\frac{1}{3} & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 2 \\ 4 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ -2 \end{bmatrix}.$$

Define v as the *absolute value* of the *negative* component of $\mathbf{c}_p$ having the *largest* absolute value, so that $v = |-2| = 2$ in this case. Consequently, in the current coordinates, the algorithm now moves from the current trial solution $(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) = (1, 1, 1)$ to the next trial solution

$$\tilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \frac{\alpha}{v}\mathbf{c}_p = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \frac{0.5}{2}\begin{bmatrix} 1 \\ 3 \\ -2 \end{bmatrix} = \begin{bmatrix} \frac{5}{4} \\ \frac{7}{4} \\ \frac{1}{2} \end{bmatrix},$$

as shown in Fig. 7.5. (The definition of v has been chosen to make the smallest component of $\tilde{\mathbf{x}}$ equal to zero when $\alpha = 1$ in this equation for the next trial solution.) In the original coordinates, this solution is

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \mathbf{D}\tilde{\mathbf{x}} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix}\begin{bmatrix} \frac{5}{4} \\ \frac{7}{4} \\ \frac{1}{2} \end{bmatrix} = \begin{bmatrix} \frac{5}{2} \\ \frac{7}{2} \\ 2 \end{bmatrix}.$$

This completes the iteration, and this new solution will be used to start the next iteration.

These steps can be summarized as follows for any iteration.

Summary of the Interior-Point Algorithm.

1. Given the current trial solution $(x_1, x_2, \ldots, x_n)$, set

$$\mathbf{D} = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ 0 & x_2 & 0 & \cdots & 0 \\ 0 & 0 & x_3 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & x_n \end{bmatrix}$$

2. Calculate $\tilde{\mathbf{A}} = \mathbf{AD}$ and $\tilde{\mathbf{c}} = \mathbf{Dc}$.

3. Calculate $\mathbf{P} = \mathbf{I} - \tilde{\mathbf{A}}^T(\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^{-1}\tilde{\mathbf{A}}$ and $\mathbf{c}_p = \mathbf{P}\tilde{\mathbf{c}}$.

4. Identify the negative component of $\mathbf{c}_p$ having the largest absolute value, and set v equal to this absolute value. Then calculate

$$\tilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + \frac{\alpha}{v}\mathbf{c}_p,$$

where α is a selected constant between 0 and 1 (for example, $\alpha = 0.5$).

5. Calculate $\mathbf{x} = \mathbf{D}\tilde{\mathbf{x}}$ as the trial solution for the next iteration (step 1). (If this trial solution is virtually unchanged from the preceding one, then the algorithm has virtually converged to an optimal solution, so stop.)

Now let us apply this summary to iteration 2 for the example.

Iteration 2.

Step 1:
Given the current trial solution $(x_1, x_2, x_3) = (\frac{5}{2}, \frac{7}{2}, 2)$, set

$$\mathbf{D} = \begin{bmatrix} \frac{5}{2} & 0 & 0 \\ 0 & \frac{7}{2} & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

(Note that the rescaled variables are

$$\begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \tilde{x}_3 \end{bmatrix} = \mathbf{D}^{-1}\mathbf{x} = \begin{bmatrix} \frac{2}{5} & 0 & 0 \\ 0 & \frac{2}{7} & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \frac{2}{5}x_1 \\ \frac{2}{7}x_2 \\ \frac{1}{2}x_3 \end{bmatrix},$$

so that the BF solutions in these new coordinates are

$$\tilde{\mathbf{x}} = \mathbf{D}^{-1} \begin{bmatrix} 8 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{16}{5} \\ 0 \\ 0 \end{bmatrix}, \qquad \tilde{\mathbf{x}} = \mathbf{D}^{-1} \begin{bmatrix} 0 \\ 8 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{16}{7} \\ 0 \end{bmatrix},$$

and

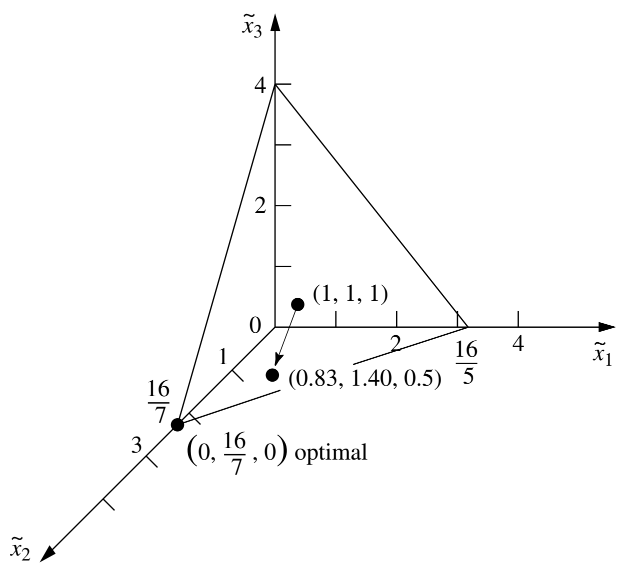
$$\tilde{\mathbf{x}} = \mathbf{D}^{-1} \begin{bmatrix} 0 \\ 0 \\ 8 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 4 \end{bmatrix},$$

as depicted in Fig. 7.6.)

Step 2:

$$\tilde{\mathbf{A}} = \mathbf{AD} = [\tfrac{5}{2}, \tfrac{7}{2}, 2] \qquad \text{and} \qquad \tilde{\mathbf{c}} = \mathbf{Dc} = \begin{bmatrix} \frac{5}{2} \\ 7 \\ 0 \end{bmatrix}.$$

FIGURE 7.6
Example after rescaling for iteration 2.



Step 3:

$$\mathbf{P} = \begin{bmatrix} \frac{13}{18} & -\frac{7}{18} & -\frac{2}{9} \\ -\frac{7}{18} & \frac{41}{90} & -\frac{14}{45} \\ -\frac{2}{9} & -\frac{14}{45} & \frac{37}{45} \end{bmatrix} \quad \text{and} \quad \mathbf{c}_p = \begin{bmatrix} -\frac{11}{12} \\ \frac{133}{60} \\ -\frac{41}{15} \end{bmatrix}.$$

Step 4:

$\left|-\frac{41}{15}\right| > \left|-\frac{11}{12}\right|$, so $v = \frac{41}{15}$ and

$$\tilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \frac{0.5}{\frac{41}{15}} \begin{bmatrix} -\frac{11}{12} \\ \frac{133}{60} \\ -\frac{41}{15} \end{bmatrix} = \begin{bmatrix} \frac{273}{328} \\ \frac{461}{328} \\ \frac{1}{2} \end{bmatrix} \approx \begin{bmatrix} 0.83 \\ 1.40 \\ 0.50 \end{bmatrix}.$$

Step 5:

$$\mathbf{x} = \mathbf{D}\tilde{\mathbf{x}} = \begin{bmatrix} \frac{1365}{656} \\ \frac{3227}{656} \\ 1 \end{bmatrix} \approx \begin{bmatrix} 2.08 \\ 4.92 \\ 1.00 \end{bmatrix}$$

is the trial solution for iteration 3.

Since there is little to be learned by repeating these calculations for additional iterations, we shall stop here. However, we do show in Fig. 7.7 the reconfigured feasible region after rescaling based on the trial solution just obtained for iteration 3. As always, the rescaling has placed the trial solution at $(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) = (1, 1, 1)$, equidistant from the $\tilde{x}_1 = 0$, $\tilde{x}_2 = 0$, and $\tilde{x}_3 = 0$ constraint boundaries. Note in Figs. 7.5, 7.6, and 7.7 how the sequence of iterations and rescaling have the effect of "sliding" the optimal solution toward (1, 1, 1) while the other BF solutions tend to slide away. Eventually, after enough iterations, the optimal solution will lie very near $(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) = (0, 1, 0)$ after rescaling, while the other two BF solutions will be *very* far from the origin on the $\tilde{x}_1$ and $\tilde{x}_3$ axes. Step 5 of that iteration then will yield a solution in the original coordinates very near the optimal solution of $(x_1, x_2, x_3) = (0, 8, 0)$.

Figure 7.8 shows the progress of the algorithm in the original $x_1 = x_2$ coordinate system before the problem is augmented. The three points—$(x_1, x_2) = (2, 2)$, (2.5, 3.5), and (2.08, 4.92)—are the trial solutions for initiating iterations 1, 2, and 3, respectively. We then have drawn a smooth curve through and beyond these points to show the trajectory of the algorithm in subsequent iterations as it approaches $(x_1, x_2) = (0, 8)$.

The functional constraint for this particular example happened to be an inequality constraint. However, equality constraints cause no difficulty for the algorithm, since it deals with the constraints only after any necessary augmenting has been done to convert them to equality form ($\mathbf{Ax} = \mathbf{b}$) anyway. To illustrate, suppose that the only change in the example is that the constraint $x_1 + x_2 \leq 8$ is changed to $x_1 + x_2 = 8$. Thus, the feasible region in Fig. 7.3 changes to just the line segment between (8, 0) and (0, 8). Given an initial feasible trial solution in the interior ($x_1 > 0$ and $x_2 > 0$) of this line segment—say, $(x_1, x_2) = (4, 4)$—the algorithm can proceed just as presented in the five-step summary with just the two variables and $\mathbf{A} = [1, 1]$. For each iteration, the projected gradient points along this line segment in the direction of (0, 8). With $\alpha = \frac{1}{2}$, iteration 1 leads from (4, 4)

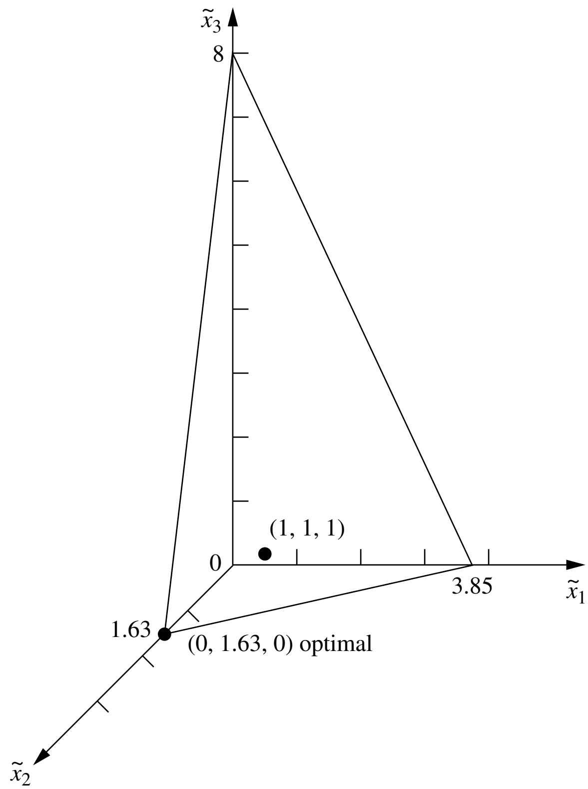


FIGURE 7.7
Example after rescaling for iteration 3.

to (2, 6), iteration 2 leads from (2, 6) to (1, 7), etc. (Problem 7.4-3 asks you to verify these results.)

Although either version of the example has only one functional constraint, having more than one leads to just one change in the procedure as already illustrated (other than more extensive calculations). Having a single functional constraint in the example meant that $\mathbf{A}$ had only a single row, so the $(\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^{-1}$ term in step 3 only involved taking the reciprocal of the number obtained from the vector product $\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T$. Multiple functional constraints mean that $\mathbf{A}$ has multiple rows, so then the $(\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T)^{-1}$ term involves finding the *inverse* of the matrix obtained from the matrix product $\tilde{\mathbf{A}}\tilde{\mathbf{A}}^T$.

To conclude, we need to add a comment to place the algorithm into better perspective. For our extremely small example, the algorithm requires relatively extensive calculations and then, after many iterations, obtains only an approximation of the optimal solution. By contrast, the graphical procedure of Sec. 3.1 finds the optimal solution in Fig. 7.3 immediately, and the simplex method requires only one quick iteration. However, do not let this contrast fool you into downgrading the efficiency of the interior-point algorithm. This algorithm is designed for dealing with *big* problems having many hundreds or thousands of functional constraints. The simplex method typically requires thousands of iterations on such problems. By "shooting" through the interior of the feasible region, the interior-point algorithm tends to require a substantially smaller number of iterations

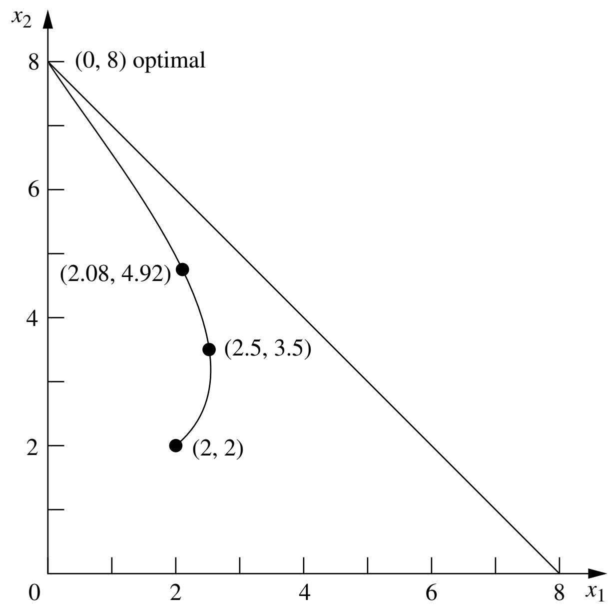


FIGURE 7.8
Trajectory of the interior-point algorithm for the example in the original x_1-x_2 coordinate system.

(although with considerably more work per iteration). Therefore, interior-point algorithms similar to the one presented here should play an important role in the future of linear programming.

See Sec. 4.9 for a further discussion of this role and a comparison of the interior-point approach with the simplex method.

7.5 LINEAR GOAL PROGRAMMING AND ITS SOLUTION PROCEDURES

We have assumed throughout the preceding chapters that the objectives of the organization conducting the linear programming study can be encompassed within a single overriding objective, such as maximizing total profit or minimizing total cost. However, this assumption is not always realistic. In fact, as we discussed in Sec. 2.1, studies have found that the management of U.S. corporations frequently focuses on a variety of other objectives, e.g., to maintain stable profits, increase (or maintain) market share, diversify products, maintain stable prices, improve worker morale, maintain family control of the business, and increase company prestige. *Goal programming* provides a way of striving toward several such objectives *simultaneously*.

The basic approach of **goal programming** is to establish a specific numeric goal for each of the objectives, formulate an objective function for each objective, and then seek a solution that minimizes the (weighted) sum of deviations of these objective functions from their respective goals. There are three possible types of goals:

1. A **lower, one-sided goal** sets a *lower limit* that we do not want to fall under (but exceeding the limit is fine).

2. An **upper, one-sided goal** sets an *upper limit* that we do not want to exceed (but falling under the limit is fine).
3. A **two-sided goal** sets a *specific target* that we do not want to miss on either side.

Goal programming problems can be categorized according to the type of mathematical programming model (linear programming, integer programming, nonlinear programming, etc.) that it fits except for having multiple goals instead of a single objective. In this book, we only consider *linear* goal programming—those goal programming problems that fit linear programming otherwise (each objective function is linear, etc.) and so we will drop the adjective *linear* from now on.

Another categorization is according to how the goals compare in importance. In one case, called **nonpreemptive goal programming,** all the goals are of *roughly comparable importance*. In another case, called **preemptive goal programming,** there is a *hierarchy of priority levels* for the goals, so that the goals of primary importance receive first-priority attention, those of secondary importance receive second-priority attention, and so forth (if there are more than two priority levels).

We begin with an example that illustrates the basic features of nonpreemptive goal programming and then discuss the preemptive case.

Prototype Example for Nonpreemptive Goal Programming

The DEWRIGHT COMPANY is considering three new products to replace current models that are being discontinued, so their OR department has been assigned the task of determining which mix of these products should be produced. Management wants primary consideration given to three factors: long-run profit, stability in the workforce, and the level of capital investment that would be required now for new equipment. In particular, management has established the goals of (1) achieving a long-run profit (net present value) of at least \$125 million from these products, (2) maintaining the current employment level of 4,000 employees, and (3) holding the capital investment to less than \$55 million. However, management realizes that it probably will not be possible to attain all these goals simultaneously, so it has discussed priorities with the OR department. This discussion has led to setting *penalty weights* of 5 for missing the profit goal (per \$1 million under), 2 for going over the employment goal (per 100 employees), 4 for going under this same goal, and 3 for exceeding the capital investment goal (per \$1 million over). Each new product's contribution to profit, employment level, and capital investment level is *proportional* to the rate of production. These contributions per unit rate of production are shown in Table 7.5, along with the goals and penalty weights.

Formulation. The Dewright Company problem includes all three possible types of goals: a lower, one-sided goal (long-run profit); a two-sided goal (employment level); and an upper, one-sided goal (capital investment). Letting the decision variables x_1, x_2, x_3 be the production rates of products 1, 2, and 3, respectively, we see that these goals can be stated as

$$\begin{aligned}
12x_1 + 9x_2 + 15x_3 &\geq 125 &\quad& \text{profit goal}\\
5x_1 + 3x_2 + 4x_3 &= 40 && \text{employment goal}\\
5x_1 + 7x_2 + 8x_3 &\leq 55 && \text{investment goal.}
\end{aligned}$$

TABLE 7.5 Data for the Dewright Co. nonpreemptive goal programming problem

	Unit Contribution				
	Product:				
Factor	**1**	**2**	**3**	**Goal (Units)**	**Penalty Weight**
Long-run profit	12	9	15	$\geq$ 125 (millions of dollars)	5
Employment level	5	3	4	= 40 (hundreds of employees)	2(+), 4(−)
Capital investment	5	7	8	$\leq$ 55 (millions of dollars)	3

More precisely, given the penalty weights in the rightmost column of Table 7.5, let Z be the *number of penalty points* incurred by missing these goals. The overall objective then is to choose the values of x_1, x_2, and x_3 so as to

$$\begin{aligned}\text{Minimize} \quad Z = {} & 5(\text{amount under the long-run profit goal}) \\ & + 2(\text{amount over the employment level goal}) \\ & + 4(\text{amount under the employment level goal}) \\ & + 3(\text{amount over the capital investment goal}),\end{aligned}$$

where no penalty points are incurred for being over the long-run profit goal or for being under the capital investment goal. To express this overall objective mathematically, we introduce some *auxiliary variables* (extra variables that are helpful for formulating the model) y_1, y_2, and y_3, defined as follows:

$$\begin{aligned}y_1 &= 12x_1 + 9x_2 + 15x_3 - 125 && \text{(long-run profit minus the target).} \\ y_2 &= 5x_1 + 3x_2 + 4x_3 - 40 && \text{(employment level minus the target).} \\ y_3 &= 5x_1 + 7x_2 + 8x_3 - 55 && \text{(capital investment minus the target).}\end{aligned}$$

Since each y_i can be either positive or negative, we next use the technique described at the end of Sec. 4.6 for dealing with such variables; namely, we replace each one by the difference of two nonnegative variables:

$$\begin{aligned}y_1 &= y_1^+ - y_1^-, && \text{where } y_1^+ \geq 0,\ y_1^- \geq 0, \\ y_2 &= y_2^+ - y_2^-, && \text{where } y_2^+ \geq 0,\ y_2^- \geq 0, \\ y_3 &= y_3^+ - y_3^-, && \text{where } y_3^+ \geq 0,\ y_3^- \geq 0.\end{aligned}$$

As discussed in Sec. 4.6, for any BF solution, these new auxiliary variables have the interpretation

$$y_j^+ = \begin{cases} y_j & \text{if } y_j \geq 0, \\ 0 & \text{otherwise;} \end{cases}$$

$$y_j^- = \begin{cases} |y_j| & \text{if } y_j \leq 0, \\ 0 & \text{otherwise;} \end{cases}$$

so that y_j^+ represents the positive part of the variable y_j and y_j^- its negative part (as suggested by the superscripts).

Given these new auxiliary variables, the overall objective can be expressed mathematically as

$$\text{Minimize} \quad Z = 5y_1^- + 2y_2^+ + 4y_2^- + 3y_3^+,$$

which now is a legitimate objective function for a linear programming model. (Because there is no penalty for exceeding the profit goal of 125 or being under the investment goal of 55, neither y_1^+ nor y_3^- should appear in this objective function representing the total penalty for deviations from the goals.)

To complete the conversion of this goal programming problem to a linear programming model, we must incorporate the above definitions of the y_j^+ and y_j^- directly into the model. (It is not enough to simply record the definitions, as we just did, because the simplex method considers only the objective function and constraints that constitute the model.) For example, since $y_1^+ - y_1^- = y_1$, the above expression for y_1 gives

$$12x_1 + 9x_2 + 15x_3 - 125 = y_1^+ - y_1^-.$$

After we move the variables $(y_1^+ - y_1^-)$ to the left-hand side and the constant (125) to the right-hand side,

$$12x_1 + 9x_2 + 15x_3 - (y_1^+ - y_1^-) = 125$$

becomes a legitimate equality constraint for a linear programming model. Furthermore, this constraint forces the auxiliary variables $(y_1^+ - y_1^-)$ to satisfy their definition in terms of the decision variables (x_1, x_2, x_3).

Proceeding in the same way for $y_2^+ - y_2^-$ and $y_3^+ - y_3^-$, we obtain the following linear programming formulation of this goal programming problem:

$$\text{Minimize} \qquad Z = 5y_1^- + 2y_2^+ + 4y_2^- + 3y_3^+,$$

subject to

$$\begin{aligned} 12x_1 + 9x_2 + 15x_3 - (y_1^+ - y_1^-) &= 125 \\ 5x_1 + 3x_2 + 4x_3 - (y_2^+ - y_2^-) &= 40 \\ 5x_1 + 7x_2 + 8x_3 - (y_3^+ - y_3^-) &= 55 \end{aligned}$$

and

$$x_j \ge 0, \qquad y_k^+ \ge 0, \qquad y_k^- \ge 0 \qquad (j = 1, 2, 3;\ k = 1, 2, 3).$$

(If the original problem had any actual linear programming constraints, such as constraints on fixed amounts of certain resources being available, these would be included in the model.)

Applying the simplex method to this formulation yields an optimal solution $x_1 = \frac{25}{3}$, $x_2 = 0$, $x_3 = \frac{5}{3}$, with $y_1^+ = 0$, $y_1^- = 0$, $y_2^+ = \frac{25}{3}$, $y_2^- = 0$, $y_3^+ = 0$, and $y_3^- = 0$. Therefore, $y_1 = 0$, $y_2 = \frac{25}{3}$, and $y_3 = 0$, so the first and third goals are fully satisfied, but the employment level goal of 40 is exceeded by $8\frac{1}{3}$ (833 employees). The resulting penalty for deviating from the goals is $Z = 16\frac{2}{3}$.

As usual, you can see how Excel, LINGO/LINDO, and MPL/CPLEX are used to set up and solve this example by referring to their files for this chapter in your OR Courseware.

Preemptive Goal Programming

In the preceding example we assume that all the goals are of roughly comparable importance. Now consider the case of *preemptive* goal programming, where there is a hierarchy of priority levels for the goals. Such a case arises when one or more of the goals clearly are far more important than the others. Thus, the initial focus should be on achieving as closely as possible these *first-priority* goals. The other goals also might naturally divide further into second-priority goals, third-priority goals, and so on. After we find an optimal solution with respect to the first-priority goals, we can break any ties for the optimal solution by considering the second-priority goals. Any ties that remain after this reoptimization can be broken by considering the third-priority goals, and so on.

When we deal with goals on the *same* priority level, our approach is just like the one described for nonpreemptive goal programming. Any of the same three types of goals (lower one-sided, two-sided, upper one-sided) can arise. Different penalty weights for deviations from different goals still can be included, if desired. The same formulation technique of introducing auxiliary variables again is used to reformulate this portion of the problem to fit the linear programming format.

There are two basic methods based on linear programming for solving preemptive goal programming problems. One is called the *sequential procedure,* and the other is the *streamlined procedure.* We shall illustrate these procedures in turn by solving the following example.

Example. Faced with the unpleasant recommendation to increase the company's workforce by more than 20 percent, the management of the Dewright Company has reconsidered the original formulation of the problem that was summarized in Table 7.5. This increase in workforce probably would be a rather temporary one, so the very high cost of training 833 new employees would be largely wasted, and the large (undoubtedly well-publicized) layoffs would make it more difficult for the company to attract high-quality employees in the future. Consequently, management has concluded that a very high priority should be placed on avoiding an increase in the workforce. Furthermore, management has learned that raising *more than* \$55 million for capital investment for the new products would be extremely difficult, so a very high priority also should be placed on avoiding capital investment above this level.

Based on these considerations, management has concluded that a *preemptive goal programming* approach now should be used, where the two goals just discussed should be the first-priority goals, and the other two original goals (exceeding \$125 million in long-run profit and avoiding a decrease in the employment level) should be the second-priority goals. Within the two priority levels, management feels that the relative penalty weights still should be the same as those given in the rightmost column of Table 7.5. This reformulation is summarized in Table 7.6, where a factor of M (representing a huge positive number) has been included in the penalty weights for the first-priority goals to emphasize that these goals preempt the second-priority goals. (The portions of Table 7.5 that are not included in Table 7.6 are *unchanged.*)

The Sequential Procedure for Preemptive Goal Programming

The *sequential procedure* solves a preemptive goal programming problem by solving a *sequence* of linear programming models.

At the first stage of the sequential procedure, the only goals included in the linear programming model are the first-priority goals, and the simplex method is applied in the

TABLE 7.6 Revised formulation for the Dewright Co. preemptive goal programming problem

Priority Level	Factor	Goal	Penalty Weight
First priority	Employment level Capital investment	≤40 ≤55	2*M* 3*M*
Second priority	Long-run profit Employment level	≥125 ≥40	5 4

usual way. If the resulting optimal solution is *unique,* we adopt it immediately without considering any additional goals.

However, if there are *multiple* optimal solutions with the same optimal value of Z (call it Z^*), we prepare to break the tie among these solutions by moving to the second stage and adding the second-priority goals to the model. If $Z^* = 0$, all the auxiliary variables representing the *deviations from first-priority goals* must equal zero (full achievement of these goals) for the solutions remaining under consideration. Thus, in this case, all these auxiliary variables now can be completely deleted from the model, where the equality constraints that contain these variables are replaced by the mathematical expressions (inequalities or equations) for these first-priority goals, to ensure that they continue to be fully achieved. On the other hand, if $Z^* > 0$, the second-stage model simply adds the second-priority goals to the first-stage model (as if these additional goals actually were first-priority goals), but then it also adds the constraint that the *first-stage objective function* equals Z^* (which enables us again to delete the terms involving first-priority goals from the second-stage objective function). After we apply the simplex method again, if there still are multiple optimal solutions, we repeat the same process for any lower-priority goals.

Example. We now illustrate this procedure by applying it to the example summarized in Table 7.6.

At the first stage, only the two *first-priority* goals are included in the linear programming model. Therefore, we can drop the common factor M for their penalty weights, shown in Table 7.6. By proceeding just as for the nonpreemptive model if these were the only goals, the resulting linear programming model is

$$\text{Minimize} \qquad Z = 2y_2^+ + 3y_3^+,$$

subject to

$$5x_1 + 3x_2 + 4x_3 - (y_2^+ - y_2^-) = 40$$
$$5x_1 + 7x_2 + 8x_3 - (y_3^+ - y_3^-) = 55$$

and

$$x_j \geq 0, \qquad y_k^+ \geq 0, \qquad y_k^- \geq 0 \qquad (j = 1, 2, 3;\ k = 2, 3).$$

(For ease of comparison with the nonpreemptive model with all four goals, we have kept the same subscripts on the auxiliary variables.)

By using the simplex method (or inspection), an optimal solution for this linear programming model has $y_2^+ = 0$ and $y_3^+ = 0$, with $Z = 0$ (so $Z^* = 0$), because there are innumerable solutions for (x_1, x_2, x_3) that satisfy the relationships

$$\begin{aligned} 5x_1 + 3x_2 + 4x_3 &\le 40 \\ 5x_1 + 7x_2 + 8x_3 &\le 55 \end{aligned}$$

as well as the nonnegativity constraints. Therefore, these two first-priority goals should be used as *constraints* hereafter. Using them as constraints will force y_2^+ and y_3^+ to remain zero and thereby disappear from the model automatically.

If we drop y_2^+ and y_3^+ but add the second-priority goals, the second-stage linear programming model becomes

$$\text{Minimize} \quad Z = 5y_1^- + 4y_2^-,$$

subject to

$$\begin{aligned} 12x_1 + 9x_2 + 15x_3 - (y_1^+ - y_1^-) & & &= 125 \\ 5x_1 + 3x_2 + 4x_3 & + y_2^- & &= 40 \\ 5x_1 + 7x_2 + 8x_3 & & + y_3^- &= 55 \end{aligned}$$

and

$$x_j \ge 0, \qquad y_1^+ \ge 0, \qquad y_k^- \ge 0 \qquad (j = 1, 2, 3;\ k = 1, 2, 3).$$

Applying the simplex method to this model yields the unique optimal solution $x_1 = 5$, $x_2 = 0$, $x_3 = 3\frac{3}{4}$, $y_1^+ = 0$, $y_1^- = 8\frac{3}{4}$, $y_2^- = 0$, and $y_3^- = 0$, with $Z = 43\frac{3}{4}$.

Because this solution is unique (*or* because there are no more priority levels), the procedure can now stop, with $(x_1, x_2, x_3) = (5, 0, 3\frac{3}{4})$ as the optimal solution for the *overall* problem. This solution fully achieves both first-priority goals as well as one of the second-priority goals (no decrease in employment level), and it falls short of the other second-priority goal (long-run profit ≥ 125) by just $8\frac{3}{4}$.

The Streamlined Procedure for Preemptive Goal Programming

Instead of solving a sequence of linear programming models, like the sequential procedure, the *streamlined procedure* finds an optimal solution for a preemptive goal programming problem by solving just *one* linear programming model. Thus, the streamlined procedure is able to duplicate the work of the sequential procedure with just *one run* of the simplex method. This one run *simultaneously* finds optimal solutions based just on first-priority goals and breaks ties among these solutions by considering lower-priority goals. However, this does require a slight modification of the simplex method.

If there are just *two* priority levels, the modification of the simplex method is one you already have seen, namely, the form of the *Big M method* illustrated throughout Sec. 4.6. In this form, instead of replacing M throughout the model by some huge positive number before running the simplex method, we retain the *symbolic* quantity M in the sequence of simplex tableaux. Each coefficient in row 0 (for each iteration) is some linear function $aM + b$, where a is the current *multiplicative factor* and b is the current *additive term.* The usual decisions based on these coefficients (entering basic variable and optimality test) now are based solely on the *multiplicative* factors, except that any ties would be broken by using the *additive* terms. This is how the OR Courseware operates when solving interactively by the simplex method (and choosing the Big M method).

The linear programming formulation for the streamlined procedure with two priority levels would include *all* the goals in the model in the usual manner, but with basic penalty weights of M and 1 assigned to deviations from first-priority and second-priority goals, respectively. If different penalty weights are desired within the same priority level, these basic penalty weights then are multiplied by the individual penalty weights assigned within the level. This approach is illustrated by the following example.

Example. For the Dewright Co. preemptive goal programming problem summarized in Table 7.6, note that (1) different penalty weights are assigned within each of the two priority levels and (2) the individual penalty weights (2 and 3) for the first-priority goals have been multiplied by M. These penalty weights yield the following single linear programming model that incorporates all the goals.

$$\text{Minimize} \qquad Z = 5y_1^- + 2My_2^+ + 4y_2^- + 3My_3^+,$$

subject to

$$\begin{aligned} 12x_1 + 9x_2 + 15x_3 - (y_1^+ - y_1^-) &= 125 \\ 5x_1 + 3x_2 + 4x_3 - (y_2^+ - y_2^-) &= 40 \\ 5x_1 + 7x_2 + 8x_3 - (y_3^+ - y_3^-) &= 55 \end{aligned}$$

and

$$x_j \geq 0, \qquad y_k^+ \geq 0, \qquad y_k^- \geq 0 \qquad (j = 1, 2, 3;\ k = 1, 2, 3).$$

Because this model uses M to symbolize a huge positive number, the simplex method can be applied as described and illustrated throughout Sec. 4.6. Alternatively, a very large positive number can be substituted for M in the model and then any software package based on the simplex method can be applied. Doing either naturally yields the same unique optimal solution obtained by the sequential procedure.

More than Two Priority Levels. When there are more than two priority levels (say, p of them), the streamlined procedure generalizes in a straightforward way. The basic penalty weights for the respective levels now are $M_1, M_2, \ldots, M_{p-1}, 1$, where M_1 represents a number that is vastly larger than M_2, M_2 is vastly larger than $M_3, \ldots$, and M_{p-1} is vastly larger than 1. Each coefficient in row 0 of each simplex tableau is now a linear function of all of these quantities, where the multiplicative factor of M_1 is used to make the necessary decisions, with tie breakers beginning with the multiplicative factor of M_2 and ending with the additive term.

7.6 CONCLUSIONS

The *dual simplex method* and *parametric linear programming* are especially valuable for postoptimality analysis, although they also can be very useful in other contexts.

The *upper bound technique* provides a way of streamlining the simplex method for the common situation in which many or all of the variables have explicit upper bounds. It can greatly reduce the computational effort for large problems.

Mathematical-programming computer packages usually include all three of these procedures, and they are widely used. Because their basic structure is based largely upon the simplex method as presented in Chap. 4, they retain the exceptional computational efficiency to handle very large problems of the sizes described in Sec. 4.8.

Various other special-purpose algorithms also have been developed to exploit the special structure of particular types of linear programming problems (such as those to be discussed in Chaps. 8 and 9). Much research is currently being done in this area.

Karmarkar's interior-point algorithm has been an exciting development in linear programming. Variants of this algorithm now provide a powerful approach for efficiently solving some very large problems.

Linear goal programming and its solution procedures provide an effective way of dealing with problems where management wishes to strive toward several goals simultaneously. The key is a formulation technique of introducing auxiliary variables that enable converting the problem into a linear programming format.

SELECTED REFERENCES

1. Hooker, J. N.: "Karmarkar's Linear Programming Algorithm," *Interfaces,* **16:** 75–90, July–August 1986.
2. Lustig, I. J., R. E. Marsten, and D. F. Shanno: "Interior-Point Methods for Linear Programming: Computational State of the Art," *ORSA Journal on Computing,* **6:** 1–14, 1994. (Also see pp. 15–86 of this issue for commentaries on this article.)
3. Marsten, R., R. Subramanian, M. Saltzman, I. Lustig, and D. Shanno: "Interior-Point Methods for Linear Programming: Just Call Newton, Lagrange, and Fiacco and McCormick!," *Interfaces,* **20:** 105–116, July–August 1990.
4. Saigal, R.: *Linear Programming: A Modern Integrated Analysis,* Kluwer Academic Publishers, Boston, 1995.
5. Schneiderjans, M.: *Goal Programming: Methodology and Applications,* Kluwer Academic Publishers, Boston, 1995.
6. Terlaky, T. (ed.): *Interior Point Methods in Mathematical Programming,* Kluwer Academic Publishers, Boston, 1996.
7. Vanderbei, R. J.: "Affine-Scaling for Linear Programs with Free Variables," *Mathematical Programming,* **43:** 31–44, 1989.
8. Vanderbei, R. J.: *Linear Programming: Foundations and Extensions,* Kluwer Academic Publishers, Boston, 1996.

LEARNING AIDS FOR THIS CHAPTER IN YOUR OR COURSEWARE

Interactive Routines:

Enter or Revise a General Linear Programming Model
Set Up for the Simplex Method—Interactive Only
Solve Interactively by the Simplex Method

An Automatic Routine:

Solve Automatically by the Interior-Point Algorithm

An Excel Add-In:

Premium Solver

"Ch. 7—Other Algorithms for LP" Files for Solving the Examples:

Excel File
LINGO/LINDO File
MPL/CPLEX File

See Appendix 1 for documentation of the software.

PROBLEMS

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

I: We suggest that you use the above interactive routines (the printout records your work). For parametric linear programming, this only applies to $\theta = 0$, after which you should proceed manually.
C: Use the computer to solve the problem.

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

7.1-1. Consider the following problem.

Maximize $\quad Z = -x_1 - x_2,$

subject to

$$\begin{aligned} x_1 + x_2 &\le 8 \\ x_2 &\ge 3 \\ -x_1 + x_2 &\le 2 \end{aligned}$$

and

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

(a) Solve this problem graphically.
(b) Use the *dual simplex method* manually to solve this problem.
(c) Trace graphically the path taken by the dual simplex method.

7.1-2.* Use the *dual simplex method* manually to solve the following problem.

Minimize $\quad Z = 5x_1 + 2x_2 + 4x_3,$

subject to

$$\begin{aligned} 3x_1 + x_2 + 2x_3 &\ge 4 \\ 6x_1 + 3x_2 + 5x_3 &\ge 10 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

7.1-3. Use the *dual simplex method* manually to solve the following problem.

Minimize $\quad Z = 7x_1 + 2x_2 + 5x_3 + 4x_4,$

subject to

$$\begin{aligned} 2x_1 + 4x_2 + 7x_3 + x_4 &\ge 5 \\ 8x_1 + 4x_2 + 6x_3 + 4x_4 &\ge 8 \\ 3x_1 + 8x_2 + x_3 + 4x_4 &\ge 4 \end{aligned}$$

and

$$x_j \ge 0, \qquad \text{for } j = 1, 2, 3, 4.$$

7.1-4. Consider the following problem.

Maximize $\quad Z = 3x_1 + 2x_2,$

subject to

$$\begin{aligned} 3x_1 + x_2 &\le 12 \\ x_1 + x_2 &\le 6 \\ 5x_1 + 3x_2 &\le 27 \end{aligned}$$

and

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

I **(a)** Solve by the *original simplex method* (in tabular form). Identify the *complementary* basic solution for the dual problem obtained at each iteration.
(b) Solve the *dual* of this problem manually by the *dual simplex method.* Compare the resulting sequence of basic solutions with the complementary basic solutions obtained in part (*a*).

7.1-5. Consider the example for case 1 of sensitivity analysis given in Sec. 6.7, where the initial simplex tableau of Table 4.8 is modified by changing b_2 from 12 to 24, thereby changing the respective entries in the right-side column of the *final* simplex tableau to 54, 6, 12, and -2. Starting from this revised final simplex tableau, use the *dual simplex method* to obtain the new optimal solution shown in Table 6.21. Show your work.

7.1-6.* Consider parts (*a*) and (*b*) of Prob. 6.7-1. Use the *dual simplex method* manually to reoptimize for each of these two cases, starting from the revised final tableau.

7.2-1.* Consider the following problem.

Maximize $Z = 8x_1 + 24x_2$,

subject to

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

and

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

Suppose that Z represents profit and that it is possible to modify the objective function somewhat by an appropriate shifting of key personnel between the two activities. In particular, suppose that the unit profit of activity 1 can be increased above 8 (to a maximum of 18) at the expense of decreasing the unit profit of activity 2 below 24 by twice the amount. Thus, Z can actually be represented as

$$Z(\theta) = (8 + \theta)x_1 + (24 - 2\theta)x_2,$$

where θ is also a decision variable such that $0 \le \theta \le 10$.

(a) Solve the original form of this problem graphically. Then extend this graphical procedure to solve the parametric extension of the problem; i.e., find the optimal solution and the optimal value of $Z(\theta)$ as a function of θ, for $0 \le \theta \le 10$.

I **(b)** Find an optimal solution for the original form of the problem by the simplex method. Then use *parametric linear programming* to find an optimal solution and the optimal value of $Z(\theta)$ as a function of θ, for $0 \le \theta \le 10$. Plot $Z(\theta)$.

(c) Determine the optimal value of θ. Then indicate how this optimal value could have been identified directly by solving only two ordinary linear programming problems. (*Hint:* A convex function achieves its maximum at an endpoint.)

I **7.2-2.** Use *parametric linear programming* to find the optimal solution for the following problem as a function of θ, for $0 \le \theta \le 20$.

Maximize $Z(\theta) = (20 + 4\theta)x_1 + (30 - 3\theta)x_2 + 5x_3$,

subject to

$$3x_1 + 3x_2 + x_3 \le 30$$
$$8x_1 + 6x_2 + 4x_3 \le 75$$
$$6x_1 + x_2 + x_3 \le 45$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

7.2-3. Consider the following problem.

Maximize $Z(\theta) = (10 - \theta)x_1 + (12 + \theta)x_2 + (7 + 2\theta)x_3$,

subject to

$$x_1 + 2x_2 + 2x_3 \le 30$$
$$x_1 + x_2 + x_3 \le 20$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

I **(a)** Use *parametric linear programming* to find an optimal solution for this problem as a function of θ, for $\theta \ge 0$.

(b) Construct the dual model for this problem. Then find an optimal solution for this dual problem as a function of θ, for $\theta \ge 0$, by the method described in the latter part of Sec. 7.2. Indicate graphically what this algebraic procedure is doing. Compare the basic solutions obtained with the complementary basic solutions obtained in part (*a*).

I **7.2-4.*** Use the *parametric linear programming* procedure for making systematic changes in the b_i parameters to find an optimal solution for the following problem as a function of θ, for $0 \le \theta \le 25$.

Maximize $Z(\theta) = 2x_1 + x_2$,

subject to

$$x_1 \le 10 + 2\theta$$
$$x_1 + x_2 \le 25 - \theta$$
$$x_2 \le 10 + 2\theta$$

and

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

Indicate graphically what this algebraic procedure is doing.

I **7.2-5.** Use *parametric linear programming* to find an optimal solution for the following problem as a function of θ, for $0 \le \theta \le 30$.

Maximize $Z(\theta) = 5x_1 + 6x_2 + 4x_3 + 7x_4$,

subject to

$$3x_1 - 2x_2 + x_3 + 3x_4 \le 135 - 2\theta$$
$$2x_1 + 4x_2 - x_3 + 2x_4 \le 78 - \theta$$
$$x_1 + 2x_2 + x_3 + 2x_4 \le 30 + \theta$$

and

$$x_j \ge 0, \qquad \text{for } j = 1, 2, 3, 4.$$

Then identify the value of θ that gives the largest optimal value of $Z(\theta)$.

7.2-6. Consider Prob. 6.7-2. Use *parametric linear programming* to find an optimal solution as a function of θ over the following ranges of θ.
(a) $0 \le \theta \le 20$.
(b) $-20 \le \theta \le 0$. (*Hint:* Substitute $-\theta'$ for θ, and then increase θ' from zero.)

7.2-7. Consider the $Z^*(\theta)$ function shown in Fig. 7.1 for *parametric linear programming* with systematic changes in the c_j parameters.
(a) Explain why this function is piecewise linear.
(b) Show that this function must be convex.

7.2-8. Consider the $Z^*(\theta)$ function shown in Fig. 7.2 for *parametric linear programming* with systematic changes in the b_i parameters.
(a) Explain why this function is piecewise linear.
(b) Show that this function must be concave.

7.2-9. Let

$$Z^* = \max \left\{ \sum_{j=1}^{n} c_j x_j \right\},$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i, \qquad \text{for } i = 1, 2, \ldots, m,$$

and

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

(where the a_{ij}, b_i, and c_j are fixed constants), and let $(y_1^*, y_2^*, \ldots, y_m^*)$ be the corresponding optimal dual solution. Then let

$$Z^{**} = \max \left\{ \sum_{j=1}^{n} c_j x_j \right\},$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i + k_i, \qquad \text{for } i = 1, 2, \ldots, m,$$

and

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

where $k_1, k_2, \ldots, k_m$ are given constants. Show that

$$Z^{**} \le Z^* + \sum_{i=1}^{m} k_i y_i^*.$$

7.3-1. Use the *upper bound technique* manually to solve the Wyndor Glass Co. problem presented in Sec. 3.1.

7.3-2. Consider the following problem.

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

subject to

$$\begin{aligned} x_1 - x_2 &\le 5 \\ x_1 \qquad &\le 10 \\ x_2 &\le 10 \end{aligned}$$

and

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

(a) Solve this problem graphically.
(b) Use the *upper bound technique* manually to solve this problem.
(c) Trace graphically the path taken by the upper bound technique.

7.3-3.* Use the *upper bound technique* manually to solve the following problem.

$$\text{Maximize} \qquad Z = x_1 + 3x_2 - 2x_3,$$

subject to

$$\begin{aligned} x_2 - 2x_3 &\le 1 \\ 2x_1 + x_2 + 2x_3 &\le 8 \\ x_1 \qquad\qquad &\le 1 \\ x_2 \qquad &\le 3 \\ x_3 &\le 2 \end{aligned}$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

7.3-4. Use the *upper bound technique* manually to solve the following problem.

$$\text{Maximize} \qquad Z = 2x_1 + 3x_2 - 2x_3 + 5x_4,$$

subject to

$$\begin{aligned} 2x_1 + 2x_2 + x_3 + 2x_4 &\le 5 \\ x_1 + 2x_2 - 3x_3 + 4x_4 &\le 5 \end{aligned}$$

and

$$0 \le x_j \le 1, \qquad \text{for } j = 1, 2, 3, 4.$$

7.3-5. Use the *upper bound technique* manually to solve the following problem.

$$\text{Maximize} \qquad Z = 2x_1 + 5x_2 + 3x_3 + 4x_4 + x_5,$$

subject to

$$x_1 + 3x_2 + 2x_3 + 3x_4 + x_5 \le 6$$
$$4x_1 + 6x_2 + 5x_3 + 7x_4 + x_5 \le 15$$

and

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

7.3-6. Simultaneously use the *upper bound technique* and the *dual simplex method* manually to solve the following problem.

$$\text{Minimize} \qquad Z = 3x_1 + 4x_2 + 2x_3,$$

subject to

$$x_1 + x_2 + x_3 \ge 15$$
$$x_2 + x_3 \ge 10$$

and

$$0 \le x_1 \le 25, \qquad 0 \le x_2 \le 5, \qquad 0 \le x_3 \le 15.$$

C **7.4-1.** Reconsider the example used to illustrate the interior-point algorithm in Sec. 7.4. Suppose that $(x_1, x_2) = (1, 3)$ were used instead as the initial feasible trial solution. Perform two iterations manually, starting from this solution. Then use the automatic routine in your OR Courseware to check your work.

7.4-2. Consider the following problem.

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

subject to

$$x_1 + x_2 \le 4$$

and

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

(a) Solve this problem graphically. Also identify all CPF solutions.
C **(b)** Starting from the initial trial solution $(x_1, x_2) = (1, 1)$, perform four iterations of the interior-point algorithm presented in Sec. 7.4 manually. Then use the automatic routine in your OR Courseware to check your work.
(c) Draw figures corresponding to Figs. 7.4, 7.5, 7.6, 7.7, and 7.8 for this problem. In each case, identify the basic (or corner-point) feasible solutions in the current coordinate system. (Trial solutions can be used to determine projected gradients.)

7.4-3. Consider the following problem.

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

subject to

$$x_1 + x_2 = 8$$

and

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

C **(a)** Near the end of Sec. 7.4, there is a discussion of what the interior-point algorithm does on this problem when starting from the initial feasible trial solution $(x_1, x_2) = (4, 4)$. Verify the results presented there by performing two iterations manually. Then use the automatic routine in your OR Courseware to check your work.
(b) Use these results to predict what subsequent trial solutions would be if additional iterations were to be performed.
(c) Suppose that the stopping rule adopted for the algorithm in this application is that the algorithm stops when two successive trial solutions differ by no more than 0.01 in any component. Use your predictions from part (*b*) to predict the final trial solution and the total number of iterations required to get there. How close would this solution be to the optimal solution $(x_1, x_2) = (0, 8)$?

7.4-4. Consider the following problem.

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

subject to

$$x_1 + 2x_2 \le 9$$
$$2x_1 + x_2 \le 9$$

and

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

(a) Solve the problem graphically.
(b) Find the *gradient* of the objective function in the original x_1-x_2 coordinate system. If you move from the origin in the direction of the gradient until you reach the boundary of the feasible region, where does it lead relative to the optimal solution?
C **(c)** Starting from the initial trial solution $(x_1, x_2) = (1, 1)$, use your OR Courseware to perform 10 iterations of the interior-point algorithm presented in Sec. 7.4.
C **(d)** Repeat part (*c*) with $\alpha = 0.9$.

7.4-5. Consider the following problem.

$$\text{Maximize} \qquad Z = 2x_1 + 5x_2 + 7x_3,$$

subject to

$$x_1 + 2x_2 + 3x_3 = 6$$

and

$$x_1 \ge 0, \qquad x_2 \ge 0, \qquad x_3 \ge 0.$$

(a) Graph the feasible region.
(b) Find the *gradient* of the objective function, and then find the *projected gradient* onto the feasible region.
(c) Starting from the initial trial solution $(x_1, x_2, x_3) = (1, 1, 1)$, perform two iterations of the interior-point algorithm presented in Sec. 7.4 manually.

C **(d)** Starting from this same initial trial solution, use your OR Courseware to perform 10 iterations of this algorithm.

C **7.4-6.** Starting from the initial trial solution $(x_1, x_2) = (2, 2)$, use your OR Courseware to apply 15 iterations of the interior-point algorithm presented in Sec. 7.4 to the Wyndor Glass Co. problem presented in Sec. 3.1. Also draw a figure like Fig. 7.8 to show the trajectory of the algorithm in the original x_1-x_2 coordinate system.

7.5-1. One of management's goals in a goal programming problem is expressed algebraically as

$$3x_1 + 4x_2 + 2x_3 = 60,$$

where 60 is the specific numeric goal and the left-hand side gives the level achieved toward meeting this goal.

(a) Letting y^+ be the amount by which the level achieved exceeds this goal (if any) and y^- the amount under the goal (if any), show how this goal would be expressed as an equality constraint when reformulating the problem as a linear programming model.

(b) If each unit over the goal is considered twice as serious as each unit under the goal, what is the relationship between the coefficients of y^+ and y^- in the objective function being minimized in this linear programming model.

7.5-2. Management of the Albert Franko Co. has established goals for the market share it wants each of the company's two new products to capture in their respective markets. Specifically, management wants Product 1 to capture at least 15 percent of its market and Product 2 to capture at least 10 percent of its market. Three advertising campaigns are being planned to try to achieve these market shares. One is targeted directly on the first product. The second targets the second product. The third is intended to enhance the general reputation of the company and its products. Letting x_1, x_2, and x_3 be the amount of money allocated (in millions of dollars) to these respective campaigns, the resulting market share (expressed as a percentage) for the two products are estimated to be

$$\text{Market share for Product 1} = 0.5x_1 + 0.2x_3,$$
$$\text{Market share for Product 2} = 0.3x_2 + 0.2x_3.$$

A total of \$55 million is available for the three advertising campaigns, but management wants at least \$10 million devoted to the third campaign. If both market share goals cannot be achieved, management considers each 1 percent decrease in the market share from the goal to be equally serious for the two products. In this light, management wants to know how to most effectively allocate the available money to the three campaigns.

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

(b) Reformulate this model as a linear programming model.

C **(c)** Use the simplex method to solve this model.

7.5-3. The Research and Development Division of the Emax Corporation has developed three new products. A decision now needs to be made on which mix of these products should be produced. Management wants primary consideration given to three factors: total profit, stability in the workforce, and achieving an increase in the company's earnings next year from the \$75 million achieved this year. In particular, using the units given in the following table, they want to

$$\text{Maximize} \quad Z = P - 6C - 3D,$$

where P = total (discounted) profit over the life of the new products,

C = change (in either direction) in the current level of employment,

D = decrease (if any) in next year's earnings from the current year's level.

The amount of any increase in earnings does not enter into Z, because management is concerned primarily with just achieving some increase to keep the stockholders happy. (It has mixed feelings about a large increase that then would be difficult to surpass in subsequent years.)

The impact of each of the new products (per unit rate of production) on each of these factors is shown in the following table:

	Unit Contribution				
	Product:				
Factor	**1**	**2**	**3**	**Goal**	**Units**
Total profit	20	15	25	Maximize	Millions of dollars
Employment level	6	4	5	= 50	Hundreds of employees
Earnings next year	8	7	5	≥ 75	Millions of dollars

(a) Define y_1^+ and y_1^-, respectively, as the amount over (if any) and the amount under (if any) the employment level goal. Define y_2^+ and y_2^- in the same way for the goal regarding earnings next year. Define x_1, x_2, and x_3 as the production rates of Products 1, 2, and 3, respectively. With these definitions, use the goal programming technique to express y_1^+, y_1^-, y_2^+, and y_2^- algebraically in terms of x_1, x_2, and x_3. Also express P in terms of x_1, x_2, and x_3.

(b) Express management's objective function in terms of x_1, x_2, x_3, y_1^+, y_1^-, y_2^+, and y_2^-.

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

C **(d)** Use the simplex method to solve this model.

7.5-4. Reconsider the original version of the Dewright Co. problem presented in Sec. 7.5 and summarized in Table 7.5. After further reflection about the solution obtained by the simplex method, management now is asking some what-if questions.

(a) Management wonders what would happen if the penalty weights in the rightmost column of Table 7.5 were to be changed to 7, 4, 1, and 3, respectively. Would you expect the optimal solution to change? Why?

C **(b)** Management is wondering what would happen if the total profit goal were to be increased to wanting at least $140 million (without any change in the original penalty weights). Solve the revised model with this change.

C **(c)** Solve the revised model if both changes are made.

7.5-5. Montega is a developing country which has 15,000,000 acres of publicly controlled agricultural land in active use. Its government currently is planning a way to divide this land among three basic crops (labeled 1, 2, and 3) next year. A certain percentage of each of these crops is exported to obtain badly needed foreign capital (dollars), and the rest of each of these crops is used to feed the populace. Raising these crops also provides employment for a significant proportion of the population. Therefore, the main factors to be considered in allocating the land to these crops are (1) the amount of foreign capital generated, (2) the number of citizens fed, and (3) the number of citizens employed in raising these crops. The following table shows how much each 1,000 acres of each crop contributes toward these factors, and the last column gives the goal established by the government for each of these factors.

	Contribution per 1,000 Acres			
	Crop:			
Factor	**1**	**2**	**3**	**Goal**
Foreign capital	\$3,000	\$5,000	\$4,000	$\geq$ \$70,000,000
Citizens fed	150	75	100	$\geq$ 1,750,000
Citizens employed	10	15	12	= 200,000

In evaluating the relative seriousness of *not* achieving these goals, the government has concluded that the following deviations from the goals should be considered *equally undesirable:* (1) each $100 under the foreign-capital goal, (2) each person under the citizens-fed goal, and (3) each deviation of one (in either direction) from the citizens-employed goal.

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

(b) Reformulate this model as a linear programming model.

C **(c)** Use the simplex method to solve this model.

(d) Now suppose that the government concludes that the importance of the various goals differs greatly so that a preemptive goal programming approach should be used. In particular, the first-priority goal is citizens fed $\geq$ 1,750,000, the second-priority goal is foreign capital $\geq$ \$70,000,000, and the third-priority goal is citizens employed = 200,000. Use the goal programming technique to formulate one complete linear programming model for this problem.

(e) Use the streamlined procedure to solve the problem as formulated in part (*d*).

C **(f)** Use the sequential procedure to solve the problem as presented in part (*d*).

7.5-6.* Consider a *preemptive goal programming* problem with three priority levels, just one goal for each priority level, and just two activities to contribute toward these goals, as summarized in the following table:

	Unit Contribution		
	Activity:		
Priority Level	**1**	**2**	**Goal**
First priority	1	2	$\leq$ 20
Second priority	1	1	= 15
Third priority	2	1	$\geq$ 40

(a) Use the *goal programming technique* to formulate one complete linear programming model for this problem.

(b) Construct the initial simplex tableau for applying the *streamlined procedure.* Identify the *initial BF solution* and the *initial entering basic variable,* but do not proceed further.

(c) Starting from (*b*), use the *streamlined procedure* to solve the problem.

(d) Use the logic of preemptive goal programming to solve the problem graphically by focusing on just the two decision variables. Explain the logic used.

(e) Use the *sequential procedure* to solve this problem. After using the *goal programming technique* to formulate the linear programming model (including auxiliary variables) at each stage, solve the model *graphically* by focusing on just the two decision variables. Identify *all* optimal solutions obtained for each stage.

7.5-7. Redo Prob. 7.5-6 with the following revised table:

	Unit Contribution		
	Activity:		
Priority Level	**1**	**2**	**Goal**
First priority	1	1	$\leq$20
Second priority	1	1	$\geq$30
Third priority	1	2	$\geq$50

7.5-8. One of the most important problems in the field of *statistics* is the *linear regression problem*. Roughly speaking, this problem involves fitting a straight line to statistical data represented by points—$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$—on a graph. If we denote the line by $y = a + bx$, the objective is to choose the constants a and b to provide the "best" fit according to some criterion. The criterion usually used is the *method of least squares*, but there are other interesting criteria where linear programming can be used to solve for the optimal values of a and b.

Formulate a linear programming model for this problem under the following criterion:

Minimize the sum of the absolute deviations of the data from the line; that is,

$$\text{Minimize} \quad \sum_{i=1}^{n} \left| y_i - (a + bx_i) \right|.$$

(*Hint:* Note that this problem can be viewed as a nonpreemptive goal programming problem where each data point represents a "goal" for the regression line.)

CASE 7.1 A CURE FOR CUBA

Fulgencio Batista led Cuba with a cold heart and iron fist—greedily stealing from poor citizens, capriciously ruling the Cuban population that looked to him for guidance, and violently murdering the innocent critics of his politics. In 1958, tired of watching his fellow Cubans suffer from corruption and tyranny, Fidel Castro led a guerilla attack against the Batista regime and wrested power from Batista in January 1959. Cubans, along with members of the international community, believed that political and economic freedom had finally triumphed on the island. The next two years showed, however, that Castro was leading a Communist dictatorship—killing his political opponents and nationalizing all privately held assets. The United States responded to Castro's leadership in 1961 by invoking a trade embargo against Cuba. The embargo forbade any country from selling Cuban products in the United States and forbade businesses from selling American products to Cuba. Cubans did not feel the true impact of the embargo until 1989 when the Soviet economy collapsed. Prior to the disintegration of the Soviet Union, Cuba had received an average of $5 billion in annual economic assistance from the Soviet Union. With the disappearance of the economy that Cuba had almost exclusively depended upon for trade, Cubans had few avenues from which to purchase food, clothes, and medicine. The avenues narrowed even further when the United States passed the Torricelli Act in 1992 that forbade American subsidiaries in third countries from doing business with Cuba that had been worth a total of $700 million annually.

Since 1989, the Cuban economy has certainly felt the impact from decades of frozen trade. Today poverty ravages the island of Cuba. Families do not have money to purchase bare necessities, such as food, milk, and clothing. Children die from malnutrition or exposure. Disease infects the island because medicine is unavailable. Optical neuritis, tuberculosis, pneumonia, and influenza run rampant among the population.

Few Americans hold sympathy for Cuba, but Robert Baker, director of Helping Hand, leads a handful of tender souls on Capitol Hill who cannot bear to see politics destroy so many human lives. His organization distributes humanitarian aid annually to needy countries around the world. Mr. Baker recognizes the dire situation in Cuba, and he wants to allocate aid to Cuba for the coming year.

Mr. Baker wants to send numerous aid packages to Cuban citizens. Three different types of packages are available. The basic package contains only food, such as grain and powdered milk. Each basic package costs \$300, weighs 120 pounds, and aids 30 people. The advanced package contains food and clothing, such as blankets and fabrics. Each advanced package costs \$350, weighs 180 pounds, and aids 35 people. The supreme package contains food, clothing, and medicine. Each supreme package costs \$720, weighs 220 pounds, and aids 54 people.

Mr. Baker has several goals he wants to achieve when deciding upon the number and types of aid packages to allocate to Cuba. First, he wants to aid at least 20 percent of Cuba's 11 million citizens. Second, because disease runs rampant among the Cuban population, he wants at least 3,000 of the aid packages sent to Cuba to be the supreme packages. Third, because he knows many other nations also require humanitarian aid, he wants to keep the cost of aiding Cuba below \$20 million.

Mr. Baker places different levels of importance on his three goals. He believes the most important goal is keeping costs down since low costs mean that his organization is able to aid a larger number of needy nations. He decides to penalize his plan by 1 point for every \$1 million above his \$20 million goal. He believes the second most important goal is ensuring that at least 3,000 of the aid packages sent to Cuba are supreme packages, since he does not want to see an epidemic develop and completely destroy the Cuban population. He decides to penalize his plan by 1 point for every 1,000 packages below his goal of 3,000 packages. Finally, he believes the least important goal is reaching at least 20 percent of the population, since he would rather give a smaller number of individuals all they need to thrive instead of a larger number of individuals only some of what they need to thrive. He therefore decides to penalize his plan by 7 points for every 100,000 people below his 20 percent goal.

Mr. Baker realizes that he has certain limitations on the aid packages that he delivers to Cuba. Each type of package is approximately the same size, and because only a limited number of cargo flights from the United States are allowed into Cuba, he is only able to send a maximum of 40,000 packages. Along with a size limitation, he also encounters a weight restriction. He cannot ship more that 6 million pounds of cargo. Finally, he has a safety restriction. When sending medicine, he needs to ensure that the Cubans know how to use the medicine properly. Therefore, for every 100 supreme packages, Mr. Baker must send one doctor to Cuba at a cost of \$33,000 per doctor.

(a) Identify one of the techniques described in this chapter that is applicable to Mr. Baker's problem.
(b) How many basic, advanced, and supreme packages should Mr. Baker send to Cuba?
(c) Mr. Baker reevaluates the levels of importance he places on each of the three goals. To sell his efforts to potential donors, he must show that his program is effective. Donors generally judge the effectiveness of a program on the number of people reached by aid packages. Mr. Baker therefore decides that he must put more importance on the goal of reaching at least 20 percent of the population. He decides to penalize his plan by 10 points for every half a percentage point below his 20 percent goal. The penalties for his other two goals remain the same. Under this scenario, how many basic, advanced, and supreme packages should Mr. Baker send to Cuba? How sensitive is the plan to changes in the penalty weights?

(d) Mr. Baker realizes that sending more doctors along with the supreme packages will improve the proper use and distribution of the packages' contents, which in turn will increase the effectiveness of the program. He therefore decides to send one doctor with every 75 supreme packages. The penalties for the goals remain the same as in part (*c*). Under this scenario, how many basic, advanced, and supreme packages should Mr. Baker send to Cuba?

(e) The aid budget is cut, and Mr. Baker learns that he definitely cannot allocate more than $20 million in aid to Cuba. Due to the budget cut, Mr. Baker decides to stay with his original policy of sending one doctor with every 100 supreme packages. How many basic, advanced, and supreme packages should Mr. Baker send to Cuba assuming that the penalties for not meeting the other two goals remain the same as in part (*b*)?