CHAPTER

SOLVING LINEAR PROGRAMMING PROBLEMS: THE SIMPLEX METHOD

4

We now are ready to begin studying the *simplex method,* a general procedure for solving linear programming problems. Developed by the brilliant George Dantzig[1] in 1947, it has proved to be a remarkably efficient method that is used routinely to solve huge problems on today's computers. Except for its use on tiny problems, this method is always executed on a computer, and sophisticated software packages are widely available. Extensions and variations of the simplex method also are used to perform *postoptimality analysis* (including sensitivity analysis) on the model.

Because linear programming problems arise so frequently for a wide variety of applications, the simplex method receives a tremendous amount of usage. During the early years after its development in 1947, computers were still relatively primitive, so only relatively small problems were being solved by this new algorithm. This changed rapidly as computers became much more powerful. Toward the end of the 20th century, problems with several thousand functional constraints and variables were being solved routinely. The progress since then has been remarkable. Both because of further explosions of computer power and great improvements in the implementation of the simplex method and its variants (such as the dual simplex method described in Sec. 8.1), this remarkable algorithm now can sometimes solve *huge* problems with millions (or even tens of millions) of functional constraints and variables. We will not attempt to delve into advanced topics that further enable its exceptional efficiency.

This chapter describes and illustrates the main features of the simplex method. The first section introduces its general nature, including its geometric interpretation. The following three sections then develop the procedure for solving any linear programming model that is in our standard form (maximization, all functional constraints in $\leq$ form, and nonnegativity constraints on all variables) and has only *nonnegative* right-hand sides b_i in the functional constraints. Certain details on resolving ties are deferred to Sec. 4.5. Section 4.6 describes how to reformulate nonstandard forms of linear programming models to prepare for applying the simplex method. The subsequent two sections then present alternative methods for helping to solve these reformulated models. Next, we discuss postoptimality analysis (Sec. 4.9), and describe the computer implementation of the simplex method (Sec. 4.10). Section 4.11 then introduces an alternative to the simplex method (the interior-point approach) for solving huge linear programming problems.

---

[1]Widely revered as perhaps the most important pioneer of operations research, George Dantzig is commonly referred to as the *father of linear programming* because of the development of the simplex method and many key subsequent contributions. The authors had the privilege of being his faculty colleagues in the Department of Operations Research at Stanford University for over 30 years. Dr. Dantzig remained professionally active right up until he passed away in 2005 at the age of 90.

4.1 THE ESSENCE OF THE SIMPLEX METHOD

The simplex method is an *algebraic procedure*. However, its underlying concepts are *geometric*. Understanding these geometric concepts provide a strong intuitive feeling for how the simplex method operates and what makes it so efficient. Therefore, before delving into algebraic details, we focus in this section on the big picture from a geometric viewpoint.

To illustrate the general geometric concepts, we shall use the Wyndor Glass Co. example presented in Sec. 3.1. (Sections 4.2 and 4.3 use the *algebra* of the simplex method to solve this same example.) Section 5.1 will elaborate further on these geometric concepts for larger problems.

To refresh your memory, the model and graph for this example are repeated in Fig. 4.1. The five constraint boundaries and their points of intersection are highlighted in this figure because they are the keys to the analysis. Here, each **constraint boundary** is a line that forms the boundary of what is permitted by the corresponding constraint. The points of intersection are the **corner-point solutions** of the problem. The five that lie on the corners of the *feasible region*—(0, 0), (0, 6), (2, 6), (4, 3), and (4, 0)—are the *corner-point feasible solutions* (**CPF solutions**). [The other three—(0, 9), (4, 6), and (6, 0)—are called *corner-point infeasible solutions*.]

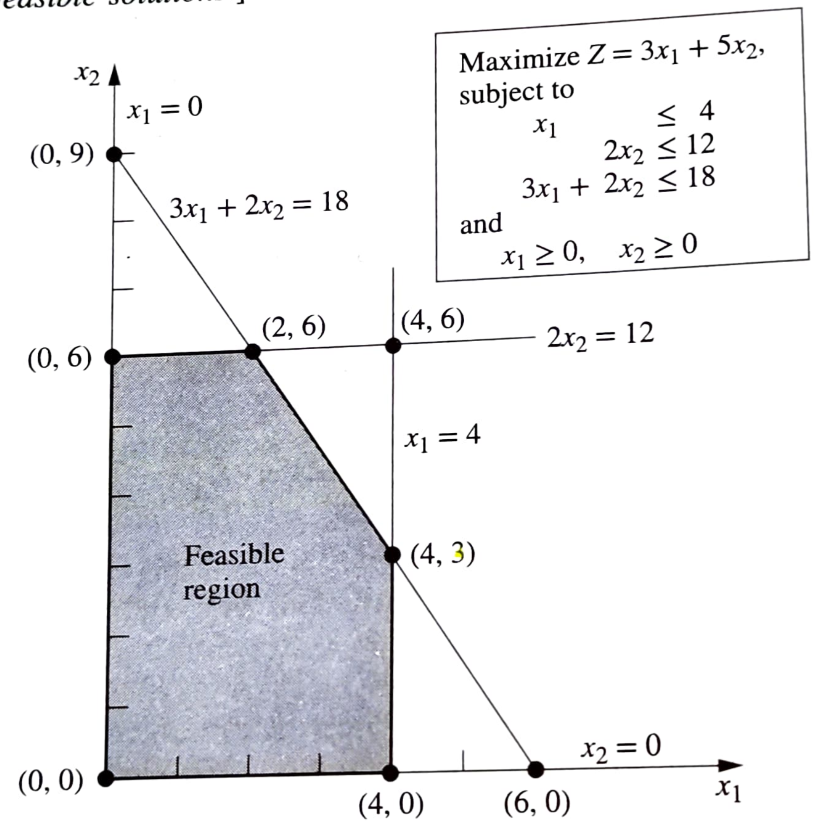

FIGURE 4.1
Constraint boundaries and corner-point solutions for the Wyndor Glass Co. problem.

In this example, each corner-point solution lies at the intersection of *two* constraint boundaries. (For a linear programming problem with *n* decision variables, each of its corner-point solutions lies at the intersection of *n* constraint boundaries.[2]) Certain pairs of the CPF solutions in Fig. 4.1 share a constraint boundary, and other pairs do not. It will be important to distinguish between these cases by using the following general definitions.

[2]Although a corner-point solution is defined in terms of *n* constraint boundaries whose intersection gives this solution, it also is possible that one or more *additional* constraint boundaries pass through this same point.

For any linear programming problem with n decision variables, two CPF solutions are **adjacent** to each other if they share $n - 1$ constraint boundaries. The two adjacent CPF solutions are connected by a line segment that lies on these same shared constraint boundaries. Such a line segment is referred to as an **edge** of the feasible region.

Since $n = 2$ in the example, two of its CPF solutions are adjacent if they share *one* constraint boundary; for example, (0, 0) and (0, 6) are adjacent because they share the $x_1 = 0$ constraint boundary. The feasible region in Fig. 4.1 has five edges, consisting of the five line segments forming the boundary of this region. Note that two edges emanate from each CPF solution. Thus, each CPF solution has two adjacent CPF solutions (each lying at the other end of one of the two edges), as enumerated in Table 4.1. (In each row of this table, the CPF solution in the first column is adjacent to each of the two CPF solutions in the second column, but the two CPF solutions in the second column are *not* adjacent to each other.)

TABLE 4.1 Adjacent CPF solutions for each CPF solution of the Wyndor Glass Co. problem

CPF Solution	Its Adjacent CPF Solutions
(0, 0)	(0, 6) and (4, 0)
(0, 6)	(2, 6) and (0, 0)
(2, 6)	(4, 3) and (0, 6)
(4, 3)	(4, 0) and (2, 6)
(4, 0)	(0, 0) and (4, 3)

One reason for our interest in adjacent CPF solutions is the following general property about such solutions, which provides a very useful way of checking whether a CPF solution is an optimal solution.

Optimality test: Consider any linear programming problem that possesses at least one optimal solution. If a CPF solution has no *adjacent* CPF solutions that are *better* (as measured by Z), then it *must* be an *optimal* solution.

Thus, for the example, (2, 6) must be optimal simply because its $Z = 36$ is larger than $Z = 30$ for (0, 6) and $Z = 27$ for (4, 3). (We will delve further into why this property holds in Sec. 5.1.) This optimality test is the one used by the simplex method for determining when an optimal solution has been reached.

Now we are ready to apply the simplex method to the example.

Solving the Example

Here is an outline of what the simplex method does (from a geometric viewpoint) to solve the Wyndor Glass Co. problem. At each step, first the conclusion is stated and then the reason is given in parentheses. (Refer to Fig. 4.1 for a visualization.)

Initialization: Choose (0, 0) as the *initial* CPF solution to examine. (This is a convenient choice because no calculations are required to identify this CPF solution.)

Optimality Test: Conclude that (0, 0) is *not* an optimal solution. (Adjacent CPF solutions are better.)

Iteration 1: Move to a better *adjacent* CPF solution, (0, 6), by performing the following three steps.

1. Considering the two edges of the feasible region that emanate from (0, 0), choose to move along the edge that leads up the x_2 axis. (With an objective function of $Z = 3x_1 + 5x_2$, moving up the x_2 axis increases Z at a faster rate than moving along the x_1 axis.)
2. Stop at the first new constraint boundary: $2x_2 = 12$. [Moving farther in the direction selected in step 1 leaves the feasible region; e.g., moving to the second new constraint boundary hit when moving in that direction gives (0, 9), which is a corner-point *infeasible* solution.]

3. Solve for the intersection of the new set of constraint boundaries: (0, 6). (The equations for these constraint boundaries, $x_1 = 0$ and $2x_2 = 12$, immediately yield this solution.)

Optimality Test: Conclude that (0, 6) is *not* an optimal solution. (An adjacent CPF solution is better.)

Iteration 2: Move to a better adjacent CPF solution, (2, 6), by performing the following three steps:

1. Considering the two edges of the feasible region that emanate from (0, 6), choose to move along the edge that leads to the right. (Moving along this edge increases Z, whereas backtracking to move back down the x_2 axis decreases Z.)
2. Stop at the first new constraint boundary encountered when moving in that direction: $3x_1 + 2x_2 = 18$. (Moving farther in the direction selected in step 1 leaves the feasible region.)
3. Solve for the intersection of the new set of constraint boundaries: (2, 6). (The equations for these constraint boundaries, $3x_1 + 2x_2 = 18$ and $2x_2 = 12$, immediately yield this solution.)

Optimality Test: Conclude that (2, 6) *is* an optimal solution, so stop. (None of the adjacent CPF solutions are better.)

This sequence of CPF solutions examined is shown in Fig. 4.2, where each circled number identifies which iteration obtained that solution. (See the Solved Examples section for this chapter on the book's website for **another example** of how the simplex method marches through a sequence of CPF solutions to reach the optimal solution.)

Now let us look at the six key solution concepts of the simplex method that provide the rationale behind the above steps. (Keep in mind that these concepts also apply for solving problems with more than two decision variables where a graph like Fig. 4.2 is not available to help quickly find an optimal solution.)

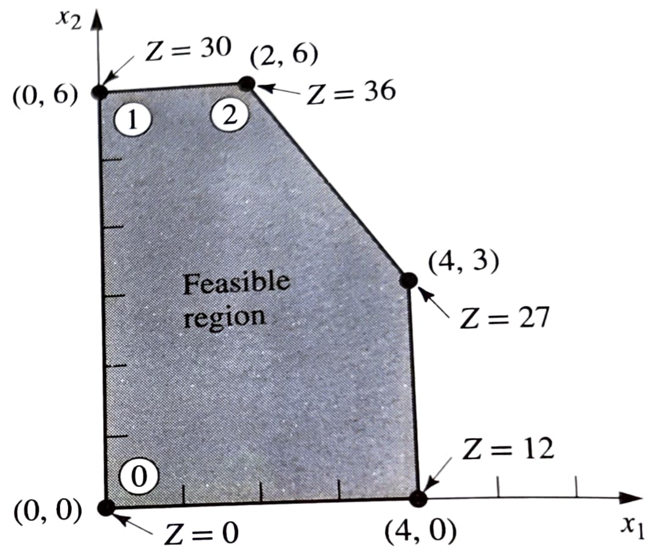

FIGURE 4.2
This graph shows the sequence of CPF solutions (⓪, ①, ②) examined by the simplex method for the Wyndor Glass Co. problem. The optimal solution (2, 6) is found after just three solutions are examined.

The Key Solution Concepts

The first solution concept is based directly on the relationship between optimal solutions and CPF solutions given at the end of Sec. 3.2.

Solution concept 1: The simplex method focuses solely on CPF solutions. For any problem with at least one optimal solution, finding one requires only finding a best CPF solution.[3]

Since the number of feasible solutions generally is infinite, reducing the number of solutions that need to be examined to a small finite number (just three in Fig. 4.2) is a tremendous simplification.

The next solution concept defines the flow of the simplex method.

Solution concept 2: The simplex method is an *iterative algorithm* (a systematic solution procedure that keeps repeating a fixed series of steps, called an *iteration*, until a desired result has been obtained) with the following structure.

[3]The only restriction is that the problem must possess CPF solutions. This is ensured if the feasible region is bounded.

Initialization: Set up to start iterations, including finding an initial CPF solution.

$\downarrow$

Optimality test: Is the current CPF solution optimal?

If no $\downarrow$ If yes $\longrightarrow$ Stop.

Iteration: Perform an iteration to find a better CPF solution.

When the example was solved, note how this flow diagram was followed through two iterations until an optimal solution was found.

We next focus on how to get started.

> **Solution concept 3:** Whenever possible, the initialization of the simplex method chooses the *origin* (all decision variables equal to zero) to be the initial CPF solution. When there are too many decision variables to find an initial CPF solution graphically, this choice eliminates the need to use algebraic procedures to find and solve for an initial CPF solution.

Choosing the origin commonly is possible when all the decision variables have nonnegativity constraints, because the intersection of these constraint boundaries yields the origin as a corner-point solution. This solution then is a CPF solution *unless* it is *infeasible* because it violates one or more of the functional constraints. If it is infeasible, special procedures described in Secs. 4.6–4.8 are needed to find the initial CPF solution.

The next solution concept concerns the choice of a better CPF solution at each iteration.

> **Solution concept 4:** Given a CPF solution, it is much quicker computationally to gather information about its *adjacent* CPF solutions than about other CPF solutions. Therefore, each time the simplex method performs an iteration to move from the current CPF solution to a better one, it *always* chooses a CPF solution that is *adjacent* to the current one. No other CPF solutions are considered. Consequently, the entire path followed to eventually reach an optimal solution is along the *edges* of the feasible region.

The next focus is on which adjacent CPF solution to choose at each iteration.

> **Solution concept 5:** After the current CPF solution is identified, the simplex method examines each of the edges of the feasible region that emanate from this CPF solution. Each of these edges leads to an *adjacent* CPF solution at the other end, but the simplex method does not even take the time to solve for the adjacent CPF solution. Instead, it simply identifies the *rate of improvement in Z* that would be obtained by moving along the edge. Among the edges with a *positive* rate of improvement in Z, it then chooses to move along the one with the *largest* rate of improvement in Z. The iteration is completed by first solving for the adjacent CPF solution at the other end of this one edge and then relabeling this adjacent CPF solution as the *current* CPF solution for the optimality test and (if needed) the next iteration.

At the first iteration of the example, moving from (0, 0) along the edge on the x_1 axis would give a rate of improvement in Z of 3 (Z increases by 3 per unit increase in x_1), whereas moving along the edge on the x_2 axis would give a rate of improvement in Z of 5 (Z increases by 5 per unit increase in x_2), so the decision is made to move along the latter edge. At the second iteration, the only edge emanating from

(0, 6) that would yield a *positive* rate of improvement in Z is the edge leading to (2, 6), so the decision is made to move next along this edge.

The final solution concept clarifies how the optimality test is performed efficiently.

Solution concept 6: Solution concept 5 describes how the simplex method examines each of the edges of the feasible region that emanate from the current CPF solution. This examination of an edge leads to quickly identifying the rate of improvement in Z that would be obtained by moving along this edge toward the adjacent CPF solution at the other end. A *positive* rate of improvement in Z implies that the adjacent CPF solution is *better* than the current CPF solution, whereas a *negative* rate of improvement in Z implies that the adjacent CPF solution is *worse*. Therefore, the optimality test consists simply of checking whether *any* of the edges give a *positive* rate of improvement in Z. If *none* do, then the current CPF solution is optimal.

In the example, moving along *either* edge from (2, 6) decreases Z. Since we want to maximize Z, this fact immediately gives the conclusion that (2, 6) is optimal.

If you would like to see **another example** illustrating the geometric concepts underlying the simplex method, one is provided in the Solved Examples section for this chapter on the book's website.

4.2 SETTING UP THE SIMPLEX METHOD

Section 4.1 stressed the geometric concepts that underlie the simplex method. However, this algorithm normally is run on a computer, which can follow only algebraic instructions. Therefore, it is necessary to translate the conceptually geometric procedure just described into a usable algebraic procedure. In this section, we introduce the *algebraic language* of the simplex method and relate it to the concepts of the preceding section. We are assuming (prior to Sec. 4.6) that we are dealing with linear programming models that are in *our standard form* (as defined at the end of the introduction to this chapter).

The algebraic procedure is based on solving systems of equations. Therefore, the first step in setting up the simplex method is to convert the functional *inequality constraints* into equivalent *equality constraints*. (The nonnegativity constraints are left as inequalities because they are treated separately.) This conversion is accomplished by introducing **slack variables.** To illustrate, consider the first functional constraint in the Wyndor Glass Co. example of Sec. 3.1,

$$x_1 \leq 4.$$

The slack variable for this constraint is defined to be

$$x_3 = 4 - x_1,$$

which is the amount of slack in the left-hand side of the inequality. Thus,

$$x_1 + x_3 = 4.$$

Given this equation, $x_1 \leq 4$ if and only if $4 - x_1 = x_3 \geq 0$. Therefore, the original constraint $x_1 \leq 4$ is entirely *equivalent* to the pair of constraints

$$x_1 + x_3 = 4 \quad \text{and} \quad x_3 \geq 0.$$

$x_1 + s_1 = 4$

$s_1 \geq 0$

Upon the introduction of slack variables for the other functional constraints, the original linear programming model for the example (shown below on the left) can now be replaced by the equivalent model (called the *augmented form* of the model) shown below on the right:

Original Form of the Model

$$
\begin{array}{ll}
\text{Maximize} & Z = 3x_1 + 5x_2, \\
\text{subject to} & \\
& x_1 \le 4 \\
& 2x_2 \le 12 \\
& 3x_1 + 2x_2 \le 18 \\
\text{and} & \\
& x_1 \ge 0, \quad x_2 \ge 0.
\end{array}
$$

Augmented Form of the Model[4]

$$
\begin{array}{lll}
\text{Maximize} & Z = 3x_1 + 5x_2, & \\
\text{subject to} & & \\
(1) & x_1 + \overset{s_1}{x_3} & = 4 \\
(2) & 2x_2 + \overset{s_2}{x_4} & = 12 \\
(3) & 3x_1 + 2x_2 + \underset{s_3}{x_5} & = 18 \\
\text{and} & & \\
& x_j \ge 0, \quad \text{for } j = 1, 2, 3, 4, 5. &
\end{array}
$$

Although both forms of the model represent exactly the same problem, the new form is much more convenient for algebraic manipulation and for identification of CPF solutions. We call this the **augmented form** of the problem because the original form has been *augmented* by some supplementary variables needed to apply the simplex method.

If a slack variable equals 0 in the current solution, then this solution lies on the constraint boundary for the corresponding functional constraint. A value greater than 0 means that the solution lies on the *feasible* side of this constraint boundary, whereas a value less than 0 means that the solution lies on the *infeasible* side of this constraint boundary. A demonstration of these properties is provided by the **demonstration example** in your OR Tutor entitled *Interpretation of the Slack Variables.*

The terminology used in Sec. 4.1 (corner-point solutions, etc.) applies to the original form of the problem. We now introduce the corresponding terminology for the augmented form.

> An **augmented solution** is a solution for the original variables (the *decision variables*) that has been augmented by the corresponding values of the *slack variables.*

For example, augmenting the solution (3, 2) in the example yields the augmented solution (3, 2, 1, 8, 5) because the corresponding values of the slack variables are $x_3 = 1$, $x_4 = 8$, and $x_5 = 5$.

> A **basic solution** is an *augmented* corner-point solution.

To illustrate, consider the corner-point infeasible solution (4, 6) in Fig. 4.1. Augmenting it with the resulting values of the slack variables $x_3 = 0$, $x_4 = 0$, and $x_5 = -6$ yields the corresponding basic solution (4, 6, 0, 0, −6).

The fact that corner-point solutions (and so basic solutions) can be either feasible or infeasible implies the following definition:

> A **basic feasible (BF) solution** is an *augmented* CPF solution.

Thus, the CPF solution (0, 6) in the example is equivalent to the BF solution (0, 6, 4, 0, 6) for the problem in augmented form.

The only difference between basic solutions and corner-point solutions (or between BF solutions and CPF solutions) is whether the values of the slack variables are included. For any basic solution, the corresponding corner-point solution is obtained simply by deleting the slack variables. Therefore, the geometric and algebraic relationships between these two solutions are very close, as we will describe further in Sec. 5.1.

---

[4]The slack variables are not shown in the objective function because the coefficients there are 0.

Because the terms *basic solution* and *basic feasible solution* are very important parts of the standard vocabulary of linear programming, we now need to clarify their algebraic properties. For the augmented form of the example, notice that the system of functional constraints has 5 variables and 3 equations, so

Number of variables − number of equations = 5 − 3 = 2.

This fact gives us 2 *degrees of freedom* in solving the system, since any two variables can be chosen to be set equal to any arbitrary value in order to solve the three equations in terms of the remaining three variables.[5] The simplex method uses zero for this arbitrary value. Thus, two of the variables (called the *nonbasic variables*) are set equal to zero, and then the simultaneous solution of the three equations for the other three variables (called the *basic variables*) is a *basic solution*. These properties are described in the following general definitions.

A **basic solution** has the following properties:

1. Each variable is designated as either a nonbasic variable or a basic variable.
2. The *number of basic variables* equals the number of functional constraints (now equations). Therefore, the *number of nonbasic variables* equals the total number of variables *minus* the number of functional constraints.
3. The **nonbasic variables** are set equal to zero.
4. The values of the **basic variables** are obtained as the simultaneous solution of the system of equations (functional constraints in augmented form). (The set of basic variables is often referred to as **the basis**.)
5. If the basic variables satisfy the *nonnegativity constraints*, the basic solution is a **BF solution**. (Remember that **BF** is an abbreviation for *basic feasible*.)

To illustrate these definitions, consider again the BF solution (0, 6, 4, 0, 6). This solution was obtained before by augmenting the CPF solution (0, 6). However, another way to obtain this same solution is to choose x_1 and x_4 to be the two nonbasic variables, and so the two variables are set equal to zero. The three equations then yield, respectively, $x_3 = 4$, $x_2 = 6$, and $x_5 = 6$ as the solution for the three basic variables, as shown below (with the basic variables in bold type):

$$
\begin{array}{llllll}
 & & & & & x_1 = 0 \text{ and } x_4 = 0 \text{ so} \\
(1) & x_1 & +\boldsymbol{x_3} & & = 4 & \boldsymbol{x_3} = 4 \\
(2) & 2\boldsymbol{x_2} & & +x_4 & = 12 & \boldsymbol{x_2} = 6 \\
(3) & 3x_1 + 2\boldsymbol{x_2} & & & +\boldsymbol{x_5} = 18 & \boldsymbol{x_5} = 6
\end{array}
$$

Because all three of these basic variables are nonnegative, this *basic solution* (0, 6, 4, 0, 6) is indeed a *BF solution*. The Solved Examples section for this chapter on the book's website includes **another example** of the relationship between CPF solutions and BF solutions.

Just as certain pairs of CPF solutions are *adjacent*, the corresponding pairs of BF solutions also are said to be adjacent. Here is an easy way to tell when two BF solutions are adjacent.

Two BF solutions are **adjacent** if *all but one* of their *nonbasic variables* are the same. This implies that *all but one* of their *basic variables* also are the same, although perhaps with different numerical values.

Consequently, moving from the current BF solution to an adjacent one involves switching one variable from nonbasic to basic and vice versa for one other variable (and then adjusting the values of the basic variables to continue satisfying the system of equations).

[5]This method of determining the number of degrees of freedom for a system of equations is valid as long as the system does not include any redundant equations. This condition always holds for the system of equations formed from the functional constraints in the augmented form of a linear programming model.

non-basic variables

To illustrate *adjacent BF solutions,* consider one pair of adjacent CPF solutions in Fig. 4.1: (0, 0) and (0, 6). Their augmented solutions, (0, 0, 4, 12, 18) and (0, 6, 4, 0, 6), automatically are adjacent BF solutions. However, you do not need to look at Fig. 4.1 to draw this conclusion. Another signpost is that their nonbasic variables, (x_1, x_2) and (x_1, x_4), are the same with just the one exception—x_2 has been replaced by x_4. Consequently, moving from (0, 0, 4, 12, 18) to (0, 6, 4, 0, 6) involves switching x_2 from nonbasic to basic and vice versa for x_4.

When we deal with the problem in augmented form, it is convenient to consider and manipulate the objective function equation at the same time as the new constraint equations. Therefore, before we start the simplex method, the problem needs to be rewritten once again in an equivalent way:

Maximize $\quad Z,$

subject to

$$
\begin{array}{llcl}
(0) & Z - 3x_1 - 5x_2 & & = 0 \\
(1) & \quad\;\; x_1 \quad\quad + x_3 & & = 4 \\
(2) & \quad\quad\quad 2x_2 \quad\; + x_4 & & = 12 \\
(3) & \quad\;\; 3x_1 + 2x_2 & + x_5 & = 18
\end{array}
$$

and

$$x_j \geq 0, \qquad \text{for } j = 1, 2, \ldots, 5.$$

LPP the entire model has been transformed into solving a system of linear eqn.

It is just as if Eq. (0) actually were one of the original constraints; but because it already is in equality form, no slack variable is needed. While adding one more equation, we also have added one more unknown (Z) to the system of equations. Therefore, when using Eqs. (1) to (3) to obtain a basic solution as described above, we use Eq. (0) to solve for Z at the same time.

Somewhat fortuitously, the model for the Wyndor Glass Co. problem fits *our standard form,* and all its functional constraints have nonnegative right-hand sides b_i. If this had not been the case, then additional adjustments would have been needed at this point before the simplex method was applied. These details are deferred to Sec. 4.6, and we now focus on the simplex method itself.

4.3 THE ALGEBRA OF THE SIMPLEX METHOD

Building on the descriptions in the two preceding sections, we now can sketch a conceptual outline of the simplex method from either a geometric or algebraic viewpoint.

Conceptual Outline of the Simplex Method

1. **Perform initialization** to identify the initial solution for starting the simplex method.
2. **Apply the optimality test** to determine if the current solution is optimal.
   - **a.** If so, stop.
   - **b.** If not, perform an iteration.
3. **Step 1 of an iteration:** Determine which direction in which to move to get to the next solution.
4. **Step 2 of an iteration:** Determine where to stop to reach this next solution.
5. **Step 3 of an iteration:** Solve for this new solution.
6. **Return to the optimality test.**

Initialization

The choice of x_1 and x_2 to be the *nonbasic* variables (the variables set equal to zero) for the initial BF solution is based on solution concept 3 in Sec. 4.1. This choice eliminates the work required to solve for the *basic variables* (x_3, x_4, x_5) from the following system of equations (where the basic variables are shown in bold type):

$$
\begin{array}{llllll}
 & & & & & x_1 = 0 \text{ and } x_2 = 0 \text{ so} \\
(1) & x_1 & & + \mathbf{x}_3 & = 4 & \mathbf{x}_3 = 4 \\
(2) & & 2x_2 & + \mathbf{x}_4 & = 12 & \mathbf{x}_4 = 12 \\
(3) & 3x_1 + 2x_2 & & + \mathbf{x}_5 & = 18 & \mathbf{x}_5 = 18
\end{array}
$$

Thus, the **initial BF solution** is (0, 0, 4, 12, 18).

Notice that this solution can be read immediately because each equation has just one basic variable, which has a coefficient of 1, and this basic variable does not appear in any other equation. You will soon see that when the set of basic variables changes, the simplex method uses an algebraic procedure (Gaussian elimination) to convert the equations to this same convenient form for reading every subsequent BF solution as well. This form is called **proper form from Gaussian elimination**.

The objective fn eqn is always written in terms of non-basic (0) variables; the coeff. of "these" non-basic variables determine the dir. of movement in the algebraic framew'k

Optimality Test

The objective function is

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

so $Z = 0$ for the initial BF solution. Because none of the basic variables (x_3, x_4, x_5) have a *nonzero* coefficient in this objective function, the coefficient of each nonbasic variable (x_1, x_2) gives the rate of improvement in Z if that variable were to be increased from zero (while the values of the basic variables are adjusted to continue satisfying the system of equations).[6] These rates of improvement (3 and 5) are *positive*. Therefore, based on solution concept 6 in Sec. 4.1, we conclude that (0, 0, 4, 12, 18) is not optimal.

For each BF solution examined after subsequent iterations, at least one basic variable has a nonzero coefficient in the objective function. Therefore, the optimality test then will use the new Eq. (0) to rewrite the objective function in terms of just the nonbasic variables, as you will see later.

Determining the Direction of Movement (Step 1 of an Iteration)

Increasing one nonbasic variable from zero (while adjusting the values of the basic variables to continue satisfying the system of equations) corresponds to moving along one edge emanating from the current CPF solution. Based on solution concepts 4 and 5 in Sec. 4.1, the choice of which nonbasic variable to increase is made as follows:

$$Z = 3x_1 + 5x_2$$

Increase x_1? Rate of improvement in $Z = 3$.
Increase x_2? Rate of improvement in $Z = 5$.
$5 > 3$, so choose x_2 to increase.

w/ (0, 0) as initial CPF soln & (0, 0, 4, 12, 18)

As indicated next, we call x_2 the *entering basic variable* for iteration 1.

x_2 goes from non-basic variable to BF soln.

At any iteration of the simplex method, the purpose of step 1 is to choose one *nonbasic variable* to increase from zero (while the values of the basic variables are adjusted to continue satisfying the system of equations). Increasing this nonbasic variable from zero will convert it to a *basic variable* for the next BF solution. Therefore, this variable is called the **entering basic variable** for the current iteration (because it is entering the basis).

Determining Where to Stop (Step 2 of an Iteration)

Step 2 addresses the question of how far to increase the entering basic variable x_2 before stopping. Increasing x_2 increases Z, so we want to go as far as possible without leaving the feasible region. The requirement to satisfy the functional constraints in augmented form (shown below) means that increasing x_2 (while keeping the nonbasic variable $x_1 = 0$) changes the values of some of the basic variables as shown on the right.

			$x_1 = 0$,	so
(1)	$x_1 + x_3$	$= 4$	$x_3 = 4$	
(2)	$2x_2 + x_4$	$= 12$	$x_4 = 12 - 2x_2$	
(3)	$3x_1 + 2x_2 + x_5$	$= 18$	$x_5 = 18 - 2x_2$.	

Write the basic variables in terms of the "leaving non-basic" variable.

[6]Note that this interpretation of the coefficients of the x_j variables is based on these variables being on the right-hand side, $Z = 3x_1 + 5x_2$. When these variables are brought to the left-hand side for Eq. (0), $Z - 3x_1 - 5x_2 = 0$, the nonzero coefficients change their signs.

The other requirement for feasibility is that all the variables be *nonnegative*. The non-basic variables (including the entering basic variable) are nonnegative, but we need to check how far x_2 can be increased without violating the nonnegativity constraints for the basic variables.

$$x_3 = 4 \geq 0 \qquad \Rightarrow \text{no upper bound on } x_2.$$

$$x_4 = 12 - 2x_2 \geq 0 \quad \Rightarrow x_2 \leq \frac{12}{2} = 6 \quad \leftarrow \text{minimum.}$$

$$x_5 = 18 - 2x_2 \geq 0 \quad \Rightarrow x_2 \leq \frac{18}{2} = 9.$$

Min^m Ratio Test:
Min^m ratio determines which var is the "leaving basic var" of "entering non-basic var."

Thus, x_2 can be increased just to 6, at which point x_4 has dropped to 0. Increasing x_2 beyond 6 would cause x_4 to become negative, which would violate feasibility.

These calculations are referred to as the **minimum ratio test**. The objective of this test is to determine which basic variable drops to zero first as the entering basic variable is increased. We can immediately rule out the basic variable in any equation where the coefficient of the entering basic variable is zero or negative, since such a basic variable would not decrease as the entering basic variable is increased. [This is what happened with x_3 in Eq. (1) of the example.] However, for each equation where the coefficient of the entering basic variable is *strictly positive* (> 0), this test calculates the *ratio* of the right-hand side to the coefficient of the entering basic variable. The basic variable in the equation with the *minimum ratio* is the one that drops to zero first as the entering basic variable is increased.

> At any iteration of the simplex method, step 2 uses the *minimum ratio test* to determine which basic variable drops to zero first as the entering basic variable is increased. Decreasing this basic variable to zero will convert it to a *nonbasic variable* for the next BF solution. Therefore, this variable is called the **leaving basic variable** for the current iteration (because it is leaving the basis).

ıus, x_4 is the leaving basic variable for iteration 1 of the example.

›olving for the New BF Solution (Step 3 of an Iteration)

ncreasing $x_2 = 0$ to $x_2 = 6$ moves us from the *initial* BF solution on the left to the *new* BF solution on the right.

	Initial BF solution	New BF solution
Nonbasic variables:	$x_1 = 0, \quad x_2 = 0$	$x_1 = 0, \quad x_4 = 0$
Basic variables:	$x_3 = 4, \quad x_4 = 12, \quad x_5 = 18$	$x_3 = ?, \quad x_2 = 6, \quad x_5 = ?$

the Gauss "Elimin" step.

The purpose of step 3 is to convert the system of equations to a more convenient form (proper form from Gaussian elimination) for conducting the optimality test and (if needed) the next iteration with this new BF solution. In the process, this form also will identify the values of x_3 and x_5 for the new solution.

Here again is the complete original system of equations, where the new basic variables are shown in old type (with Z playing the role of the basic variable in the objective function equation):

$$\begin{array}{lrcrcrcrcrcl}
(0) & Z & - & 3x_1 & - & 5\mathbf{x_2} & & & & & = & 0 \\
(1) & & & x_1 & & & + & \mathbf{x_3} & & & = & 4 \\
(2) & & & & & 2\mathbf{x_2} & & & + & x_4 & = & 12 \\
(3) & & & 3x_1 & + & 2\mathbf{x_2} & & & & + \mathbf{x_5} & = & 18.
\end{array}$$

non-basic Variables

$x_1 = 0, x_2 = 6, x_4 = 0$

unknowns: Z, x_3, x_5

(Calc. Z from eq (0)

$x_3 = 4$

eq (2) is

$x_5 = ($

$x_4 \leftarrow x_2$

(0,0,1,0) → for x_2 in the sys.

Thus, x_2 has replaced x_4 as the basic variable in Eq. (2). To solve this system of equations for Z, x_2, and x_3, we need to perform some **elementary algebraic operations** to reproduce the current pattern of coefficients of x_4 (0, 0, 1, 0) as the new coefficients of x_2. We can use either of two types of elementary algebraic operations:

1. Multiply (or divide) an equation by a nonzero constant.
2. Add (or subtract) a multiple of one equation to (or from) another equation.

To prepare for performing these operations, note that the coefficients of x_2 in the above system of equations are -5, 0, 2, and 2, respectively, whereas we want these coefficients to become 0, 0, 1, and 0, respectively. To turn the coefficient of 2 in Eq. (2) into 1, we use the first type of elementary algebraic operation by dividing Eq. (2) by 2 to obtain

* Also confer pg. 126

$$(2) \quad x_2 + \frac{1}{2}x_4 = 6.$$

To turn the coefficients of -5 and 2 into zeros, we need to use the second type of elementary algebraic operation. In particular, we add 5 times this new Eq. (2) to Eq. (0), and subtract 2 times this new Eq. (2) from Eq. (3). The resulting complete new system of equations is

$$
\begin{array}{llrl}
(0) & Z - 3x_1 & + \frac{5}{2}x_4 & = 30 \\
(1) & x_1 + x_3 & & = 4 \\
(2) & x_2 & + \frac{1}{2}x_4 & = 6 \\
(3) & 3x_1 & - x_4 + x_5 & = 6.
\end{array}
$$

basic variable

5 + 2 unknown

$x_1, x_4 = 0$ known

6 − 2 = 4 unknowns

four eqs!

Since $x_1 = 0$ and $x_4 = 0$, the equations in this form immediately yield the new BF solution, (x_1, x_2, x_3, x_4, x_5) = (0, 6, 4, 0, 6), which yields $Z = 30$.

This procedure for obtaining the simultaneous solution of a system of linear equations is called the *Gauss-Jordan method of elimination*, or **Gaussian elimination** for short.[7] The key concept for this method is the use of elementary algebraic operations to reduce the original system of equations to proper form from Gaussian elimination, where each basic variable has been eliminated from all but one equation (*its* equation) and has a coefficient of +1 in that equation.

* Always write the obj. fn. in terms of the (current) non-basic variables!

Optimality Test for the New BF Solution

The current Eq. (0) gives the value of the objective function in terms of just the current nonbasic variables:

$$Z = 30 + 3x_1 - \frac{5}{2}x_4.$$

Increasing either of these nonbasic variables from zero (while adjusting the values of the basic variables to continue satisfying the system of equations) would result in moving toward one of the two *adjacent* BF solutions. Because x_1 has a *positive* coefficient, increasing x_1 would lead to an adjacent BF solution that is better than the current BF solution, so the current solution is not optimal.

[7]Actually, there are some technical differences between the Gauss-Jordan method of elimination and Gaussian elimination, but we shall not make this distinction.

(Const

Iteration 2 and the Resulting Optimal Solution

Since $Z = 30 + 3x_1 - \frac{5}{2}x_4$, Z can be increased by increasing x_1, but not x_4. Therefore, step 1 chooses x_1 to be the entering basic variable.

For step 2, the current system of equations yields the following conclusions about how far x_1 can be increased (with $x_4 = 0$):

$$x_3 = 4 - x_1 \geq 0 \quad \Rightarrow x_1 \leq \frac{4}{1} = 4.$$

$$x_2 = 6 \geq 0 \quad \Rightarrow \text{no upper bound on } x_1.$$

$$x_5 = 6 - 3x_1 \geq 0 \quad \Rightarrow x_1 \leq \frac{6}{3} = 2 \quad \leftarrow \text{minimum.}$$

(nb) (nb)
$X_5 \leftarrow X_1$
$(0,0,0,1) \to$ for x_1

Therefore, the minimum ratio test indicates that x_5 is the leaving basic variable.

For step 3, with x_1 replacing x_5 as a basic variable, we perform elementary algebraic operations on the current system of equations to reproduce the current pattern of coefficients of x_5 (0, 0, 0, 1) as the new coefficients of x_1. This yields the following new system of equations:

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

$$(1) \qquad x_3 + \frac{1}{3}x_4 - \frac{1}{3}x_5 = 2$$

$$(2) \qquad x_2 + \frac{1}{2}x_4 \qquad = 6$$

$$(3) \quad x_1 \qquad - \frac{1}{3}x_4 + \frac{1}{3}x_5 = 2.$$

$3x_1 - x_4 + x_5 = 6$
$x_1 = \frac{6}{3} - \frac{x_5}{3}$
$= 2 - \frac{x_5}{3}$
$(-3x_1) = (x_5 - 6)$

Therefore, the next BF solution is $(x_1, x_2, x_3, x_4, x_5) = (2, 6, 2, 0, 0)$, yielding $Z = 36$. To apply the *optimality test* to this new BF solution, we use the current Eq. (0) to express Z in terms of just the current nonbasic variables:

$$Z = 36 - \frac{3}{2}x_4 - x_5.$$

26/11/

Increasing either x_4 or x_5 would *decrease* Z, so neither adjacent BF solution is as good as the current one. Therefore, based on solution concept 6 in Sec. 4.1, the current BF solution must be optimal.

In terms of the original form of the problem (no slack variables), the optimal solution is $x_1 = 2$, $x_2 = 6$, which yields $Z = 3x_1 + 5x_2 = 36$.

To see another example of applying the simplex method, we recommend that you now view the ... OR Tutor. This vivid demonstration simul-

(iii) $R_4 \leftarrow R_4 - 2R_3$

Now compare Table 4.8 with the work done in Sec. 4.3 to verify that these two forms of the simplex method really are *equivalent*. Then note how the algebraic form is superior for learning the logic behind the simplex method, but the tabular form organizes the work being done in a considerably more convenient and compact form. We generally use the tabular form from now on.

An **additional example** of applying the simplex method in tabular form is available to you in the OR Tutor. See the demonstration entitled *Simplex Method—Tabular Form*. **Another example** also is included in the Solved Examples section for this chapter on the book's website.

TABLE 4.8 Complete set of simplex tableaux for the Wyndor Glass Co. problem

Iteration	Basic Variable	Eq.	Z	Coefficient of: x_1	x_2	x_3	x_4	x_5	Right Side
0	Z	(0)	1	−3	−5	0	0	0	0
	x_3	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	2	0	1	0	12
	x_5	(3)	0	3	2	0	0	1	18
1	Z	(0)	1	−3	0	0	$\frac{5}{2}$	0	30
	x_3	(1)	0	1	0	1	0	0	4
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	6
	x_5	(3)	0	3	0	0	−1	1	6
2	Z	(0)	1	0	0	0	$\frac{3}{2}$	1	36
	x_3	(1)	0	0	0	1	$\frac{1}{3}$	$-\frac{1}{3}$	2
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	6
	x_1	(3)	0	1	0	0	$-\frac{1}{3}$	$\frac{1}{3}$	2

4.5 TIE BREAKING IN THE SIMPLEX METHOD

You may have noticed in the preceding two sections that we never said what to do if the various choice rules of the simplex method do not lead to a clear-cut decision, because of either ties or other similar ambiguities. We discuss these details now.

Tie for the Entering Basic Variable

Step 1 of each iteration chooses the nonbasic variable having the *negative* coefficient with the *largest absolute value* in the current Eq. (0) as the entering basic variable. Now suppose that two or more nonbasic variables are tied for having the largest negative coefficient (in absolute terms). For example, this would occur in the first iteration for the Wyndor Glass Co. problem if its objective function were changed to $Z = 3x_1 + 3x_2$, so that the initial Eq. (0) became $Z - 3x_1 - 3x_2 = 0$. How should this tie be broken?

The answer is that the selection between these contenders may be made *arbitrarily*. The optimal solution will be reached eventually, regardless of the tied variable chosen, and there is no convenient

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} Z \\ x_2 \\ x_3 \\ x_5 \end{pmatrix} = \begin{pmatrix} 30 \\ 4 \\ 6 \end{pmatrix}$$

Note: In this system of eqns. in matrix form, we do not include the non-basic variables b/c the[y] are each

method for predicting in advance which choice will lead there sooner. In this example, the simplex method happens to reach the optimal solution (2, 6) in three iterations with x_1 as the initial entering basic variable, versus two iterations if x_2 is chosen.

Tie for the Leaving Basic Variable—Degeneracy

Now suppose that two or more basic variables tie for being the leaving basic variable in step 2 of an iteration. Does it matter which one is chosen? Theoretically it does, and in a very critical way, because of the following sequence of events that could occur. First, all the tied basic variables reach zero simultaneously as the entering basic variable is increased. Therefore, the one or ones *not* chosen to be the leaving basic variable also will have a value of zero in the new BF solution. (Note that basic variables with a value of *zero* are called **degenerate**, and the same term is applied to the corresponding BF solution.) Second, if one of these degenerate basic variables retains its value of zero until it is chosen at a subsequent iteration to be a leaving basic variable, the corresponding entering basic variable also must remain zero (since it cannot be increased without making the leaving basic variable negative), so the value of Z must remain unchanged. Third, if Z may remain the same rather than increase at each iteration, the simplex method may then go around in a loop, repeating the same sequence of solutions periodically rather than eventually increasing Z toward an optimal solution. In fact, examples have been artificially constructed so that they do become entrapped in just such a perpetual loop.[10]

Fortunately, although a perpetual loop is theoretically possible, it has rarely been known to occur in practical problems. If a loop were to occur, one could always get out of it by changing the choice of the leaving basic variable. Furthermore, special rules[11] have been constructed for breaking ties so that such loops are always avoided. However, these rules frequently are ignored in actual application, and they will not be repeated here. For your purposes, just break this kind of tie arbitrarily and proceed without worrying about the degenerate basic variables that result.

TABLE 4.9 Initial simplex tableau for the Wyndor Glass Co. problem without the last two functional constraints

Basic Variable	Eq.	Coefficient of: Z	x_1	x_2	x_3	Right Side	Ratio
Z	(0)	1	−3	−5	0	0	
x_3	(1)	0	1	$\boxed{0}$	1	4	None

With $x_1 = 0$ and x_2 increasing, $x_3 = 4 - 1x_1 - 0x_2 = 4 > 0$.

No Leaving Basic Variable—Unbounded Z

In step 2 of an iteration, there is one other possible outcome that we have not yet discussed, namely, that *no* variable qualifies to be the leaving basic variable.[12] This outcome would occur if the entering basic variable could be increased *indefinitely* without giving negative values to *any* of the current basic variables. In tabular form, this means that *every* coefficient in the pivot column (excluding row 0) is either negative or zero.

[10] For further information about cycling around a perpetual loop, see J. A. J. Hall and K. I. M. McKinnon: "The Simplest Examples Where the Simplex Method Cycles and Conditions Where EXPAND Fails to Prevent Cycling," *Mathematical Programming*, Series B, **100**(1): 135–150, May 2004.

[11] See R. Bland: "New Finite Pivoting Rules for the Simplex Method," *Mathematics of Operations Research*, **2**: 103–107, 1977.

[12] Note that the analogous case (no *entering* basic variable) cannot occur in step 1 of an iteration, because the optimality test would stop the algorithm first by indicating that an optimal solution had been reached.

As illustrated in Table 4.9, this situation arises in the example displayed in Fig. 3.6. In this example, the last two functional constraints of the Wyndor Glass Co. problem have been overlooked and so are not included in the model. Note in Fig. 3.6 how x_2 can be increased indefinitely (thereby increasing Z indefinitely) without ever leaving the feasible region. Then note in Table 4.9 that x_2 is the entering basic variable but the only coefficient in the pivot column is zero. Because the minimum ratio test uses only coefficients that are greater than zero, there is no ratio to provide a leaving basic variable.

The interpretation of a tableau like the one shown in Table 4.9 is that the constraints do not prevent the value of the objective function Z from increasing indefinitely, so the simplex method would stop with the message that Z is *unbounded*. Because even linear programming has not discovered a way of making infinite profits, the real message for practical problems is that a mistake has been made! The model probably has been misformulated, either by omitting relevant constraints or by stating them incorrectly. Alternatively, a computational mistake may have occurred.

Multiple Optimal Solutions

We mentioned in Sec. 3.2 (under the definition of **optimal solution**) that a problem can have more than one optimal solution. This fact was illustrated in Fig. 3.5 by changing the objective function in the Wyndor Glass Co. problem to $Z = 3x_1 + 2x_2$, so that every point on the line segment between (2, 6) and (4, 3) is optimal. Thus, all optimal solutions are a *weighted average* of these two optimal CPF solutions

$$(x_1, x_2) = w_1(2, 6) + w_2(4, 3),$$

where the weights w_1 and w_2 are numbers that satisfy the relationships

$$w_1 + w_2 = 1 \quad \text{and} \quad w_1 \geq 0, \quad w_2 \geq 0.$$

For example, $w_1 = \frac{1}{3}$ and $w_2 = \frac{2}{3}$ give

$$(x_1, x_2) = \frac{1}{3}(2, 6) + \frac{2}{3}(4, 3) = \left(\frac{2}{3} + \frac{8}{3}, \ \frac{6}{3} + \frac{6}{3}\right) = \left(\frac{10}{3}, \ 4\right)$$

as one optimal solution.

In general, any weighted average of two or more solutions (vectors) where the weights are nonnegative and sum to 1 is called a **convex combination** of these solutions. Thus, every optimal solution in the example is a convex combination of (2, 6) and (4, 3).

This example is typical of problems with multiple optimal solutions.

As indicated at the end of Sec. 3.2, *any* linear programming problem with multiple optimal solutions (and a bounded feasible region) has at least two CPF solutions that are optimal. *Every* optimal solution is a convex combination of these optimal CPF solutions. Consequently, in augmented form, every optimal solution is a convex combination of the optimal BF solutions.

(Problems 4.5-5 and 4.5-6 guide you through the reasoning behind this conclusion.)

The simplex method automatically stops after *one* optimal BF solution is found. However, for many applications of linear programming, there are intangible factors not incorporated into the model that can be used to make meaningful choices between alternative optimal solutions. In such cases, these other optimal solutions should be identified as well. As indicated above, this requires finding all the other optimal BF solutions, and then every optimal solution is a convex combination of the optimal BF solutions.

After the simplex method finds one optimal BF solution, you can detect if there are any others and, if so, find them as follows: