

Chapter 4: The Mechanics of the Simplex Method

The simplex method is a remarkably simple and elegant algorithmic engine for solving linear programs. In this chapter we will examine the internal mechanics of the simplex method as formalized in the *simplex tableau*, a table representation of the basis at any cornerpoint. The tableau contains all the information that is needed to decide on the exchange of variables that drives the movement between cornerpoints as the simplex method advances. Using the tableau, you can solve small LPs by hand, though this is tedious and error-prone. More importantly, the tableau calculations are similar to the calculations carried out by computer implementations of the simplex method. You need a good grasp of the tableau calculations to understand how the computer does its work on large LPs, how the computer calculations can go wrong, and why certain modifications to the basic method permit faster and larger solutions by computer.

Let's review what we know so far about the simplex method by presenting a high level view of the algorithm, as in Figure 4.1.

Phase 1: Find an initial cornerpoint feasible solution (basic feasible solution). If none is found, then the model is infeasible, so exit.

Note: If dealing with a standard form LP, the origin is always a basic feasible solution, so you can always start there. If it is not a standard form LP, then you must use a special phase 1 procedure that we will study later.

Phase 2: Iterate until the stopping conditions are met.

2.1 *Are we optimal yet?* Look at the current version of the objective function to see if an entering basic variable is available. If none is available, then exit with the current basic feasible solution as the optimum solution.

2.2 *Select entering basic variable:* choose the nonbasic variable that gives the fastest rate of increase in the objective function value.

2.3 *Select the leaving basic variable* by applying the Minimum Ratio Test.

2.4 *Update the equations* to reflect the new basic feasible solution.

2.5 Go to Step 2.1.

Figure 4.1: Overview of the simplex method.

The Simplex Tableau

The Acme Bicycle Company problem is a standard form LP, so we know that the origin is a basic feasible solution (feasible cornerpoint). Here is the simplex tableau for the basic feasible solution for ABC at the origin:

basic variable	eqn. no.	Z	x_1	x_2	s_1	s_2	s_3	RHS	MRT
Z	0	1	-15	-10	0	0	0	0	<i>never</i>
s_1	1	0	1	0	1	0	0	2	
s_2	2	0	0	1	0	1	0	3	
s_3	3	0	1	1	0	0	1	4	

Tableau 4.1: ABC at the origin.

The tableau is just a way of writing the equation versions of the objective function and the constraints in a nice aligned manner. The second column, titled *eqn. no.*, gives a label to each of the equations. Equation 0 is the objective function, and equations 1-3 are the three constraint equations. The *RHS* column is the right hand side of the equation, where the constant goes. For example, equation 3 could be written out in full as:

$$0Z + 1x_1 + 1x_2 + 0s_1 + 0s_2 + 1s_3 = 4$$

or more simply as:

$$x_1 + x_2 + s_3 = 4$$

which you will recognize as the equality version of the original constraint on the metal finishing machine production limit.

The objective function is also rewritten to align with the constraints. The original version was $Z = 15x_1 + 10x_2$, but we have moved all of the variables to the left hand side to create this equivalent version: $Z - 15x_1 - 10x_2 = 0$.

The *MRT* column is for the minimum ratio test. We will fill this column in as we begin the simplex calculations. The *basic variable* column shows the single basic variable that occurs in each equation.

The most important thing about Tableau 4.1 is that it is in *proper form* (remember, *standard form* is a type of LP, but *proper form* is the way that a tableau is written). A tableau in proper form has these characteristics:

- exactly 1 basic variable per equation.
- the coefficient of the basic variable is always exactly +1, and the coefficients above and below the basic variable in the same column are all 0.
- Z is treated as the basic variable for the objective function row (equation 0).

A big advantage of proper form is that you can always read the current solution directly from the tableau. This is because there is exactly one basic variable per row, and the coefficient of that variable is +1. All the other variables in the row are nonbasic (set to zero, remember?), so the value of any variable is just given by the value shown in the right hand side column. For example, in Tableau 4.1, the value of s_3 is 4. What's the value of x_1 ? Well, x_1 does not appear in the list of basic variables, which means it must be nonbasic, and nonbasic variables are always zero, hence x_1 is zero in this basic feasible solution.

What's the value of the objective function in this tableau? Because Z is the basic variable for equation 0, we can read its current value from the RHS column: its value is 0.

Now we can use the tableau to keep track of our calculations as we go through the steps of phase 2 of the simplex method as described in Figure 4.1.

Step 2.1: Are we optimal yet?

We are optimal if no entering basic variable is available. Recall that the entering basic variable chosen is the one that gives the fastest rate of increase in the objective function value. In the original version of the objective function, $Z = 15x_1 + 10x_2$, the obvious entering basic variable is x_1 . In the tableau, however, we have moved all of the variables to the left hand side of the equation, to obtain the form $Z - 15x_1 - 10x_2 = 0$. Because of this the rule is a little bit different: *choose the variable in the objective function row that has the most negative value as the entering basic variable*.

In this step, then, just check to see if there are *any* negative coefficients in the objective function row. There are two potential entering basic variables in the tableau at this point, so we are *not* optimal yet: the simplex iterations must continue.

Of course, we are only selecting from among currently nonbasic variables at this point: these are the only ones that could enter the basis. A nice feature of proper form is that *only* nonbasic variables can have a nonzero value in the objective function row, so you never have to worry about accidentally selecting a basic variable (which is already in the basic set) as the entering basic variable. Proper form enforces this by requiring that each basic variable have all zeroes in its column, except for the row in which it is the basic variable.

Step 2.2: Select the entering basic variable

As described above, when using the simplex tableau, the entering basic variable is the nonbasic variable in the objective function row (equation 0) that has the most negative coefficient. In Tableau 4.1, this is x_1 whose objective function coefficient of -15 is more negative than the objective function coefficient of -10 for x_2 . The column for the entering basic variable is called the *pivot column*. The pivot column is highlighted for ease of reference in Tableau 4.2.

Recall that the entering basic variable, which is currently nonbasic and therefore set to zero, will now be allowed to take on a positive value.

Step 2.3: Select the leaving basic variable

The minimum ratio test is used to determine the leaving basic variable. Recall that this test determines which constraint most limits the increase in the value of the entering basic variable. The most limiting constraint is the one whose basic variable is driven to zero first as the entering basic variable increases in value.

For the minimum ratio test, look only at the entries in the pivot column (the column for the entering basic variable) for the constraint rows, and calculate:

$$(\text{RHS})/(\text{coefficient of entering basic variable})$$

There are two special cases:

- if the coefficient of the entering basic variable is zero: enter *no limit* in the minimum ratio test column
- if the coefficient of the entering basic variable is negative: enter *no limit* in the minimum ratio test column.

The minimum ratio test is *never* applied to the objective function row. Why? The objective function row is not a constraint, so it can never limit the increase in the value of the entering basic variable. The objective function always goes along for the ride, just measuring the value of the objective at any point that it is given.

As usual in the minimum ratio test, the leaving basic variable is associated with the row that has the minimum value of the ratio test. This row is called the *pivot row* and is highlighted for ease of reference in Tableau 4.2.

At this point, we have determined that we are not optimal, and have selected an entering basic variable and a leaving basic variable. These can now be swapped so that we can move to the next basic feasible solution (feasible cornerpoint). The calculations so far are summarized in Tableau 4.2.

basic variable	eqn. no.	Z	x_1	x_2	s_1	s_2	s_3	RHS	MRT
Z	0	1	-15	-10	0	0	0	0	<i>never</i>
s_1	1	0	1	0	1	0	0	2	$2/1 = 2$
s_2	2	0	0	1	0	1	0	3	<i>no limit</i>
s_3	3	0	1	1	0	0	1	4	$4/1 = 4$

Tableau 4.2: Updated ABC at the origin.

Step 2.4: Update the tableau

Now that the entering and leaving basic variables are known, we can move to the new and better basic feasible solution. This is easily done by putting the tableau back into proper form. It is not in proper form now because the exchange of the entering and leaving basic variables has not yet been done, so for example, s_1 still shows as a basic variable when it should now be nonbasic, and the column for x_1 , the new basic variable does not have zeros everywhere except in its own row.

Here are the steps for putting the tableau back into proper form:

Step 2.4.1: In the column entitled *basic variable*, replace the leaving basic variable listed for the pivot row by the entering basic variable. In Tableau 4.2 the basic variable listed for equation 1 is s_1 , but in Tableau 4.3 it is x_1 .

Step 2.4.2: The tableau element where the pivot row and the pivot column intersect is known as the *pivot element*. This will be the coefficient associated with the new basic

variable for this row, so if it is not already +1, then we must divide all of the elements in the pivot row by the pivot element to obtain a +1 in the pivot element position. Luckily, we already have a +1 in the pivot element position in Tableau 4.2.

Step 2.4.3: Now we need to eliminate all of the coefficients in the pivot column except the pivot element. This is done by simple Gaussian operations. To remove the pivot column element in some row k for example, proceed as follows:

$$(\text{new row } k) = (\text{row } k) - (\text{pivot column coefficient in row } k) \times (\text{pivot row})$$

Remember that the pivot row has been updated by now so that the pivot element is +1.

For example, to clean out the -15 that appears in the objective function row,

- new Z coefficient = $1 - (-15) \times 0 = 1$
- new x_1 coefficient = $-15 - (-15) \times 1 = 0$
- new x_2 coefficient = $-10 - (-15) \times 0 = -10$
- new s_1 coefficient = $0 - (-15) \times 1 = 15$
- new s_2 coefficient = $0 - (-15) \times 0 = 0$
- new s_3 coefficient = $0 - (-15) \times 0 = 0$
- new RHS = $0 - (-15) \times 2 = 30$

Always follow the procedure given above to make sure that there is never any change in the Z column, because Z will always be the basic variable for the objective function, so you must always maintain a +1 in the equation 0 row under the Z column.

Tableau 4.3 shows the tableau after it has been put into proper form by eliminating all of the elements in the pivot column except the +1 for the pivot element. This now summarizes the second basic feasible solution (or cornerpoint, if thinking graphically).

basic variable	eqn. no.	Z	x_1	x_2	s_1	s_2	s_3	RHS	MRT
Z	0	1	0	-10	15	0	0	30	<i>never</i>
x_1	1	0	1	0	1	0	0	2	
s_2	2	0	0	1	0	1	0	3	
s_3	3	0	0	1	-1	0	1	2	

Tableau 4.3: The second basic feasible solution.

As before, the values of the basic variables at the second feasible solution can be read directly from Tableau 4.3. For example, the value of x_1 is now 2. Recalling that nonbasic variables are always zero, the new basic feasible solution is $(x_1, x_2, s_1, s_2, s_3) = (2, 0, 0, 3, 2)$. The value of Z can also be read directly from the tableau because it is the basic variable for equation 0: its value now is 30, which is an improvement over the value at the initial basic feasible solution.

In putting the tableau into proper form, we have reached the second basic feasible solution. This process of moving from one basic feasible solution to the next is called *pivoting*, and so we have just completed one pivot.

Completing the ABC solution

We can see from Tableau 4.3 that have not yet reached the optimal solution because there are still negative coefficients available in the objective function row, equation 0. So the solution process continues for another iteration, as follows:

- **Step 2.2:** x_2 is the entering basic variable (see Tableau 4.4).
- **Step 2.3:** the minimum ratio test shows that s_3 is the leaving basic variable (see Tableau 4.4).
- **Step 2.4:** Tableau 4.5 shows the revised tableau in proper form.

basic variable	eqn. no.	Z	x_1	x_2	s_1	s_2	s_3	RHS	MRT
Z	0	1	0	-10	15	0	0	30	<i>never</i>
x_1	1	0	1	0	1	0	0	2	no limit
s_2	2	0	0	1	0	1	0	3	$3/1 = 3$
s_3	3	0	0	1	-1	0	1	2	$2/1 = 2$

Tableau 4.4: The second iteration.

basic variable	eqn. no.	Z	x_1	x_2	s_1	s_2	s_3	RHS	MRT
Z	0	1	0	0	5	0	10	50	<i>never</i>
x_1	1	0	1	0	1	0	0	2	
s_2	2	0	0	0	1	1	-1	1	
x_2	3	0	0	1	-1	0	1	2	

Tableau 4.5: The optimal tableau.

Applying Step 2.1 shows that we have reached the optimum solution in Tableau 4.5: there are no negative variable coefficients in the objective function. This means that there is no improving direction in which to move. We can now read the optimum solution from the tableau (remembering that any variables not shown in the *basic variables* column must be nonbasic and hence zero): $(x_1, x_2, s_1, s_2, s_3) = (2, 2, 0, 1, 0)$. The optimum value of the objective function is the value of Z shown in the tableau: 50.

The optimum solution shows that the Acme Bicycle Company should produce 2 mountain bikes per day and 2 racing bicycles per day for a total daily profit of \$50 per day.

Special Cases in Tableau Manipulation

There are several unusual cases that arise as you progress from tableau to tableau while carrying out the simplex method. Each of these special cases has an interesting interpretation.

Tie for the Entering Basic Variable

Suppose that there are two or more nonbasic variables that have the same most negative value in the objective function row. For example, suppose that the objective function for the Acme Bicycle Company problem was actually $Z = 15x_1 + 15x_2$ (i.e. $Z - 15x_1 - 15x_2 = 0$). What is the interpretation? This means that both x_1 and x_2 , if allowed to become basic and increase in value, would both increase Z at the same rate. See Figure 4.2 for an example of a cornerpoint at which the rate of increase would be the same for two entering basic variables.

So how to handle this situation? Simple: choose the entering basic variable arbitrarily from among those tied at the most negative value, because there is no good way to determine in advance which selection will reach the optimum solution in the smallest number of pivots. Figure 4.3 shows an example in which two variables are tied, but for which many more pivots are required to reach the optimum via one choice (clockwise) than via the other choice (counterclockwise).

In fact, the choice of the “most negative objective function coefficient” is only a *heuristic* method (rule of thumb) for choosing the entering basic variable. It usually provides a good entering basic variable, but not always. On the other hand it is very quick. In commercial solvers for very large LPs, the solver may not even calculate the objective function coefficients for all of the variables. Instead it might only calculate them for say 1000 of the variables in a 500,000 variable problem, and take the most negative from among those 1000. Next time around it might calculate the objective function coefficients for the next 1000 variables and choose the entering basic variable only from among that 1000. The only time it might calculate the objective function coefficients for all 500,000 variables is at the optimum, when it must check that there are no negative coefficients left at all.

Tie for the Leaving Basic Variable

Recall the mantra: “nonbasic, variable set to zero, corresponding constraint is active”. This should remind you that the choosing the leaving basic variable via the minimum ratio test is the same as finding out which constraint you first bump into as the entering basic variable increases in value. So what does it mean when there is a tie during the minimum ratio test?

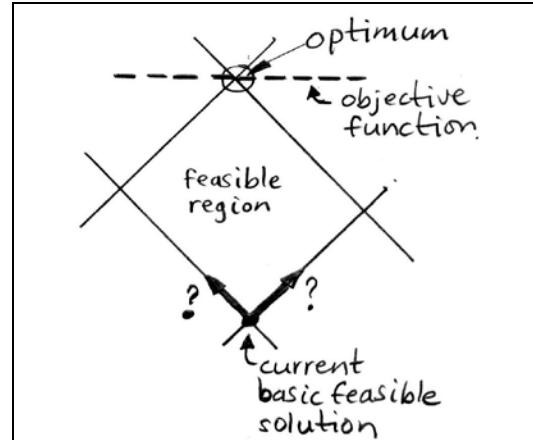


Figure 4.2: A tie for the entering basic variable.

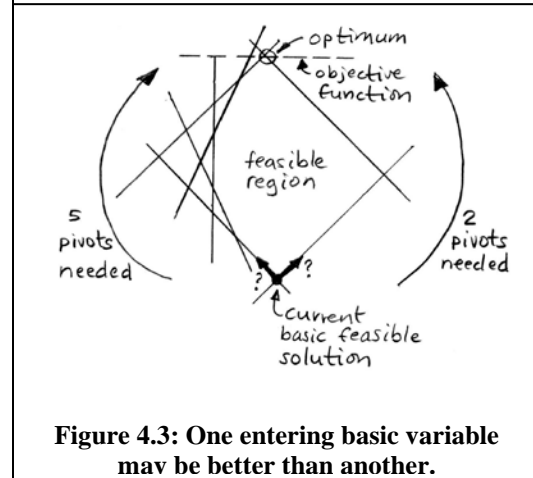
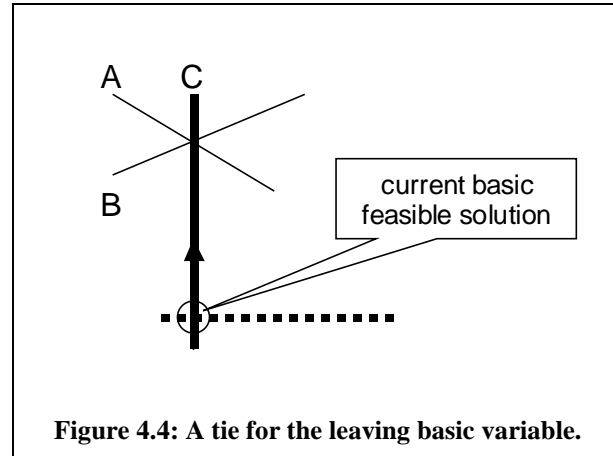


Figure 4.3: One entering basic variable may be better than another.

As Figure 4.4 shows, it must mean that there are two constraints (with their corresponding basic variables) that are tied as the first constraints you bump into. Choosing the basic variable associated with constraint A *or* the basic variable associated with constraint B as the leaving basic variable will define the same new cornerpoint. Technically speaking, the basic feasible solutions will be different, because each will define a different partitioning of the variables into basic and nonbasic sets.



So how do we choose between the two possibilities for the leaving basic variable? Again, simply choose arbitrarily. The variable that is *not* chosen as the leaving basic variable will remain basic, but will have a calculated value of zero. In contrast, the variable chosen as the leaving basic variable will of course be forced to zero by simplex. Both variables must be zero simultaneously because both constraints are active at that point; it's just that simplex only needs one of them to define the basic feasible solution.

When there is a tie for the leaving basic variable as we have described, the basic feasible solution defines what is known as a *degenerate solution*. Degenerate solutions can lead to an infinite loop of solutions that traps the simplex solution method; this is known as *cycling*. This does not happen in 2-dimensional problems, so it is difficult to show a simple diagram. But here is the general idea of how cycling happens, if it could happen in two dimensions (refer to Figure 4.4): first define the basic feasible solution using constraints A and C, then pivot to a new basic feasible solution defined by constraints B and C, then pivot back to the basic solution defined by constraints A and C, then continue around this loop infinitely.

Cycling is actually fairly rare in real problems, and most good-quality commercial linear programming solvers have special code to detect and exit from cycling behavior. One of the most often used anti-cycling routines for network LPs was in fact invented by former Carleton University professor William Cunningham.

Now suppose that all of the minimum ratio tests are tied at “no limit”: what does this mean? The interpretation is straightforward: no constraint puts a limit on the increase in the value of the entering basic variable! Because of this, there is then no limit on the increase in the value of the objective function. Your result might then show that you can make an infinite amount of profit for the Acme Bicycle Company, for example. Sounds great, but it usually means that you forgot a constraint. You may have assumed, for example, that you can sell whatever quantity of widgets you manufacture, or that a key ingredient is available in infinite quantities. Problems of this type are called *unbounded*, and have an *unbounded solution*. Figure 4.5 gives an example of an unbounded problem. An unbounded problem is recognized while pivoting when all of the entries in the pivot column (the entering basic

variable column) are zero or negative, and hence place no limit on the increase in the value of the entering basic variable.

At the Optimum, the Coefficients of Some Nonbasic Variables are Zero in the Objective Function Row

You expect that the objective function coefficients of the *basic* variables will be zero because putting the tableau into proper form requires that those coefficients be zeroed. But the coefficients of the *nonbasic* variables are normally nonzero, and in fact are normally positive at the optimum (if any are negative, you are not yet at the optimum). So what does it mean if a nonbasic variable has a coefficient of zero in the objective function at the optimum? It means that choosing this variable as the entering basic variable increases Z at the rate of zero, i.e. that choosing this variable as the entering basic variable has *no effect* on Z !

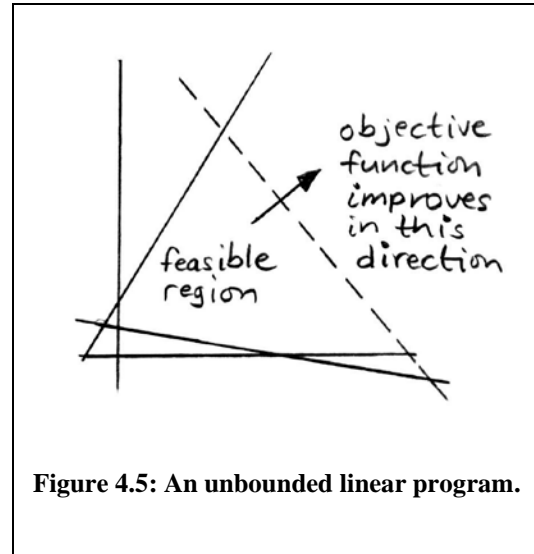


Figure 4.5: An unbounded linear program.

But choosing such a variable as the entering basic variable means that you will pivot to a *different* basic feasible solution. And the resulting new basic feasible solution will have the *same* value of Z . Two different basic feasible solutions, same value of Z ? That means that there must be *multiple optimum solutions*. You will have the same value of the objective function at both of the two basic feasible solutions *and* at any point between those two solutions. Remember that there may in fact be more than two basic feasible solutions with the same value of Z .

You may wish to find the other basic feasible solutions. Sometimes one solution is preferred over another for non-quantifiable reasons, so you would like to see a selection of solutions that give the optimum value. To see the other optima, choose one of the nonbasic variables whose objective function coefficient is zero as the entering basic variable, and pivot to another basic feasible solution as you normally would.