CHAPTER 11: BASIC LINEAR PROGRAMMING CONCEPTS

Linear programming is a mathematical technique for finding optimal solutions to problems that can be expressed using linear equations and inequalities. If a real-world problem can be represented accurately by the mathematical equations of a linear program, the method will find the best solution to the problem. Of course, few complex real-world problems can be expressed perfectly in terms of a set of linear functions. Nevertheless, linear programs can provide reasonably realistic representations of many real-world problems — especially if a little creativity is applied in the mathematical formulation of the problem.

The subject of modeling was briefly discussed in the context of regulation. The regulation problems you learned to solve were very simple mathematical representations of reality. This chapter continues this trek down the modeling path. As we progress, the models will become more mathematical — and more complex. The real world is always more complex than a model. Thus, as we try to represent the real world more accurately, the models we build will inevitably become more complex. You should remember the maxim discussed earlier that a model should only be as complex as is necessary in order to represent the real world problem reasonably well. Therefore, the added complexity introduced by using linear programming should be accompanied by some significant gains in our ability to represent the problem and, hence, in the quality of the solutions that can be obtained. You should ask yourself, as you learn more about linear programming, what the benefits of the technique are and whether they outweigh the additional costs.

The jury is still out on the question of the usefulness of linear programming in forest planning. Nevertheless, linear programming has been widely applied in forest management planning. Initial applications of the technique to forest management planning problems started in the mid 1960s. The sophistication of these analyses grew until, by the mid-1970s, the technique was being applied in real-world forest planning and not just in academic exercises. The passage of the Forest and Rangeland Renewable Resource Planning Act in 1974 created a huge demand for analytical forest planning methods, and linear programming was subsequently applied on almost every national forest in the country. The forest products industry has also adopted linear programming in their planning. Today, most large forest landowners use linear programming, or more advanced techniques similar to linear programming, in their forest management planning.

Linear programming (LP) is a relatively complex technique. The objective in this class is only to provide you with an introduction to LP and it's application in forest management planning. You should not expect to finish the course a linear programming expert. This is unnecessary, since few of you will ever need to formulate a forest planning LP in your careers. However, since so much forest planning today is based on LP techniques — or, more generally, mathematical programming techniques — it is very likely that you will need to understand at an intuitive level how mathematical programming is used in forest

management planning. By the end of the course, you should have a basic understanding of how LP works; you should be able to formulate a small forest management planning problem as an LP; and you should be able to interpret the LP solution of a forest management planning problem. This background will help you understand modern forest planning better so you can be a better participant in forest planning processes and so you will feel more comfortable implementing forest plans that are based on mathematical programming techniques. Finally, you should understand the process of mathematical programming well enough to recognize some of the potential problems and pitfalls of applying these techniques.

1. A Brief Introduction to Linear Programming

Linear programming is <u>not</u> a programming language like C++, Java, or Visual Basic. Linear programming can be defined as:

> "A mathematical method to allocate scarce resources to competing activities in an optimal manner when the problem can be expressed using a linear objective function and linear inequality constraints."

A linear program consists of a set of variables, a linear objective function indicating the contribution of each variable to the desired outcome, and a set of linear constraints describing the limits on the values of the variables. The "answer" to a linear program is a set of values for the problem variables that results in the best — largest or smallest — value of the objective function and yet is consistent with all the constraints. *Formulation* is the process of translating a real-world problem into a linear program. Once a problem has been formulated as a linear program, a computer program can be used to solve the problem. In this regard, solving a linear program is relatively easy. The hardest part about applying linear programming is formulating the problem and interpreting the solution.

Linear Equations

All of the equations and inequalities in a linear program must, by definition, be linear. A linear function has the following form:

$$a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 + \ldots + a_n x_n = 0$$

In general, the a's are called the *coefficients* of the equation; they are also sometimes called *parameters*. The important thing to know about the coefficients is that they are fixed values, based on the underlying nature of the problem being solved. The x's are called the *variables* of the equation; they are allowed to take on a range of values within the limits defined by the constraints. Note that it is not necessary to always use x's to represent variables; any label could be used, and more descriptive labels are often more useful.

Linear equations and inequalities are often written using *summation notation*, which makes it possible to write an equation in a much more compact form. The linear equation above, for example, can be written as follows:

$$a_0 + \sum_{i=1}^{n} a_i x_i = 0$$

Note that the letter *i* is an *index*, or counter, that starts in this case at 1 and runs to *n*. There is a term in the sum for each value of the index. Just as a variable does not have to be specified with a letter *x*, the index does not have to be a letter *i*. Summation notation will be used a lot in the rest of this chapter and in all of the remaining chapters. You will need to become adept at interpreting it.

The Decision Variables

The *variables* in a linear program are a set of quantities that need to be determined in order to solve the problem; i.e., the problem is solved when the best values of the variables have been identified. The variables are sometimes called *decision variables* because the problem is to decide what value each variable should take. Typically, the variables represent the amount of a resource to use or the level of some activity. For example, a variable might represent the number of acres to cut from a particular part of the forest during a given period. Frequently, defining the variables of the problem is one of the hardest and/or most crucial steps in formulating a problem as a linear program. Sometimes creative variable definition can be used to dramatically reduce the size of the problem or make an otherwise non-linear problem linear.

As mentioned earlier, a variety of symbols, with subscripts and superscripts as needed, can be used to represent the variables of an LP. As a general rule, it is better to use variable names that help you remember what the variable represents in the real world. For this general introduction, the variables will be represented — very abstractly — as $X_1, X_2, \ldots, X_n$. (Note that there are *n* variables in this list.)

The Objective Function

The objective of a linear programming problem will be to maximize or to minimize some numerical value. This value may be the expected net present value of a project or a forest property; or it may be the cost of a project; it could also be the amount of wood produced, the expected number of visitor-days at a park, the number of endangered species that will be saved, or the amount of a particular type of habitat to be maintained. Linear programming is an extremely general technique, and its applications are limited mainly by our imaginations and our ingenuity.

The *objective function* indicates how each variable contributes to the value to be optimized in solving the problem. The objective function takes the following general form:

$$\text{maximize } or \text{ minimize } Z = \sum_{i=1}^{n} c_i X_i$$

where c_i = the objective function coefficient corresponding to the ith variable, and
X_i = the ith decision variable.[1]

The coefficients of the objective function indicate the contribution to the value of the objective function of one unit of the corresponding variable. For example, if the objective function is to maximize the present value of a project, and X_i is the ith possible activity in the project, then c_i (the objective function coefficient corresponding to X_i) gives the net present value generated by one unit of activity i. As another example, if the problem is to minimize the cost of achieving some goal, X_i might be the amount of resource i used in achieving the goal. In this case, c_i would be the cost of using one unit of resource i.

Note that the way the general objective function above has been written implies that there is a coefficient in the objective function corresponding to each variable. Of course, some variables may not contribute to the objective function. In this case, you can either think of the variable as having a coefficient of zero, or you can think of the variable as not being in the objective function at all.

The Constraints

Constraints define the possible values that the variables of a linear programming problem may take. They typically represent resource constraints, or the minimum or maximum level of some activity or condition. They take the following general form:

$$\text{subject to } \sum_{i=1}^{n} a_{j,i} X_i \leq b_j \quad j = 1, 2, \dots, m$$

where X_i = the ith decision variable,
$a_{j,i}$ = the coefficient on X_i in constraint j, and
b_j = the right-hand-side coefficient on constraint j.

Note that j is an *index* that runs from 1 to m, and each value of j corresponds to a constraint. Thus, the above expression represents m constraints (equations, or, more precisely,

---

[1] The summation notation used here was discussed in the section above on linear functions. The summation notation for the objective function can be expanded out as follows:

$$Z = \sum_{i=1}^{n} c_i X_i = c_1 X_1 + c_2 X_2 + c_3 X_3 + \dots + c_n X_n$$

inequalities) with this form. Resource constraints are a common type of constraint. In a resource constraint, the coefficient $a_{j,i}$ indicates the amount of resource j used for each unit of activity i, as represented by the value of the variable X_i. The right-hand side of the constraint (b_j) indicates the total amount of resource j available for the project.

Note also that while the constraint above is written as a less-than-or-equal constraint, greater-than-or-equal constraints can also be used. A greater-than-or-equal constraint can always be converted to a less-than-or-equal constraint by multiplying it by -1. Similarly, equality constraints can be written as two inequalities — a less-than-or-equal constraint and a greater-than-or-equal constraint.

The Non-negativity Constraints

For technical reasons beyond the scope of this book, the variables of linear programs must always take non-negative values (i.e., they must be greater than or equal to zero). In most cases, where, for example, the variables might represent the levels of a set of activities or the amounts of some resource used, this non-negativity requirement will be reasonable – even necessary. In the rare case where you actually want to allow a variable to take on a negative value there are certain formulation "tricks" that can be employed. These "tricks" also are beyond the scope of this class, however, and all of the variables we will use will only need to take on non-negative values. In any case, the non-negativity constraints are part of all LP formulations, and you should always include them in an LP formulation. They are written as follows:

$$X_i \$ \, 0 \quad i = 1, 2, \ldots, n$$

where X_i = the ith decision variable.

A General Linear Programming Problem

At this point, all of the pieces of a general linear programming problem have been discussed. It may be useful to put all of the pieces together. All LP problems have the following general form:

$$\text{maximize } or \text{ minimize } Z = \sum_{i=1}^{n} c_i X_i$$

$$\text{subject to } \sum_{i=1}^{n} a_{j,i} X_i \le b_j \quad j = 1, 2, \ldots, m$$

$$\text{and} \qquad X_i \$ \, 0 \quad i = 1, 2, \ldots, n$$

where X_i = the ith decision variable,

c_i = the objective function coefficient corresponding to the ith variable,
$a_{j,i}$ = the coefficient on X_i in constraint j, and
b_j = the right-hand-side coefficient on constraint j.

Now that you have a general idea – albeit, an abstract one – of the structure of a linear program, the next step is to consider the process of formulating a linear programming problem. The following section walks you through the process of formulating two example problems. This should help give you a more concrete idea of what a linear program is.

2. <u>Linear Programming Problem Formulation</u>

We are not going to be concerned in this class with the question of how LP problems are solved. Instead, we will focus on problem formulation — translating real-world problems into the mathematical equations of a linear program — and interpreting the solutions to linear programs. We will let the computer solve the problems for us.

This section introduces you to the process of formulating linear programs. The basic steps in formulation are:

 1. Identify the decision variables;
 2. Formulate the objective function; and
 3. Identify and formulate the constraints.
 4. A trivial step, but one you should not forget, is writing out the non-negativity
   constraints.

The only way to learn how to formulate linear programming problems is to do it. So, consider the following example problem.

 Example — A Lumber Mill Problem

  A lumber mill can produce pallets or high quality lumber. Its lumber capacity is limited by its kiln size. It can dry 200 mbf per day. Similarly, it can produce a maximum of 600 pallets per day. In addition, it can only process 400 logs per day through its main saw. Quality lumber sells for $490 per mbf, and pallets sell for $9 each. It takes 1.4 logs on average to make one mbf of lumber, and four pallets can be made from one log. Of course, different grades of logs are used in making each product. Grade 1 lumber logs cost $200 per log, and pallet-grade logs cost only $4 per log. Processing costs per mbf of quality lumber are $200 per mbf, and processing costs per pallet are only $5. How many pallets and how many mbf of lumber should the mill produce?

  Answer: The first step in formulating this problem is defining the decision variables. In this case, this step is fairly straightforward. Ask yourself, "what do I need to know in order to solve the problem?" Then read the last sentence (question) in the problem description. The problem here is to determine the

number of pallets and the number of mbf of lumber to produce each day. Thus, the decision variables will be:

P = the number of pallets to produce each day, and
L = the number of mbf of lumber to produce each day.

The next step is to formulate the objective function. First, consider what the manager of this mill would probably want to do. A common mistake here would be to assume that the objective is to maximize the daily revenue. The problem with this objective is that it ignores the cost of production. A more appropriate objective function would be to maximize the daily net revenue, taking into account both costs and revenues.

Now, we must translate this verbal objective into a mathematical function. Remember, the objective function must be a linear function of the variables. This means that the objective function must have the following general form:

$$\text{Max } Z = c_P \cdot P + c_L \cdot L$$

Now, consider the units of the variables and the parameters. The objective is to maximize daily net revenues, so the units of Z must be \$/day. The units of P are pallets/day, and the units of L are mbf/day. Therefore, the units of c_P must be \$/pallet, and the units of c_L must be \$/mbf. (Can you see why?) The coefficient c_P should give the net revenue per pallet, and the coefficient c_L should give the net revenue per mbf of lumber. A pallet sells for \$9. The production cost per pallet is \$5. To calculate the raw material cost — the log cost — per pallet, note that it takes 1/4 logs to produce one pallet and that each log costs \$4. The net revenue per pallet is therefore:

c_P (\$/pallet) = 9(\$/pallet) - 5(\$/pallet) - ¼(logs/pallet)·4(\$/log) = \$3/pallet

Similarly, the net revenue per mbf of lumber is:

c_L (\$/mbf) = 490(\$/mbf) - 200(\$/mbf) - 1.4(logs/mbf)·200(\$/log) = \$10/mbf

We have now completed the formulation of the objective function. It is:

Max Z (\$/day) = 3 (\$/pallet) · P (pallets/day) + 10 (\$/mbf) · L (mbf/day)

Or, without the units:

Max $Z = 3 \cdot P + 10 \cdot L$

The next step is to identify and formulate the constraints. When formulating class examples, read the problem carefully to identify any statements that imply some kind of limit on what you can do. Limits on what you can do will typically be represented by less-than-or-equal constraints. Greater-than-or-equal constraints will typically be needed when there is a minimum level of something that is required. Of course, in a real-world

situation, you will have to think harder because the constraints will be less obvious. Often an unrealistic solution to an initial formulation will be your best clue that some real-world constraint has been missed.

In the current example, the first constraint mentioned is the kiln size. Kiln capacity limits the total daily production of lumber to a level less than or equal to 200 mbf/day. This constraint can be written simply as:

> *L (mbf/day) # 200 (mbf/day)*   *(Kiln capacity constraint)*

Note that the coefficient on the variable L in this constraint is 1. The coefficient on P is 0.

The next constraint mentioned in the problem description is the maximum capacity for pallet production. Since a maximum of 600 pallets can be produced in a day, this constraint can be written as:

> *P (pallets/day) # 600 (pallets/day)*   *(Pallet capacity constraint)*

In this constraint, the coefficient on P is 1, and the coefficient on L is 0.

The third, and final, constraint mentioned in the problem description is the maximum number of logs that can be processed by the main saw. This constraint is more difficult to write than the previous two, where only one variable was involved and the coefficients were either 0 or 1. As with the objective function, formulating this constraint is much easier if you keep in mind the units of the right-hand-side of the constraint and the units of the variables. The units of the right-hand-side of this constraint are logs/day. Since the units of the variable P are pallets/day, the units of the coefficient on the pallets variable must be logs/pallet. Similarly, the units of the coefficient on the lumber variable must be logs/mbf. Knowing the units of the parameter can be a tremendous help when you are trying to figure out what a parameter value should be. In this case, we can determine from the problem description that it takes 1/4 of a log to make a pallet and that it takes 1.4 logs to make a thousand board feet of lumber. The *log capacity constraint* can be written:

> *¼(logs/pallet)·P(pallets/day) + 1.4(logs/mbf)·L(mbf/day) # 400 (logs/day)*

The only constraints remaining are the non-negativity constraints:

> *P(pallets/day) $ 0*   and   *L(mbf/day) $ 0*

Now, the entire linear programming problem can be written as follows:

> Max *Z ($/day) = 3 ($/pallet) · P (pallets/day) + 10 ($/mbf) · L (mbf/day)*
>   *(Objective function)*

Subject to:

L (mbf/day) # 200 mbf/day   *(Kiln capacity constraint)*

P (pallets/day) # 600 (pallets/day)   *(Pallet capacity constraint)*

¼(logs/pallet)·P(pallets/day) + 1.4(logs/mbf)·L(mbf/day) # 400 (logs/day)
                                 (Log capacity constraint)

P(pallets/day) $ 0 and   *L(mbf/day) $ 0(Non-negativity constraints)*

Or, without the units:

Max $Z = 3 \cdot P + 10 \cdot L$

Subject to:

L # 200
P # 600
¼·P + 1.4·L # 400
P $ 0 and   *L $ 0*

Example — A Logging Problem

A logging concern must allocate logging equipment between two sites in the manner that will maximize its daily net revenues. They have determined that the net revenue of a cord of wood is $1.90 from site 1 and $2.10 from site 2. At their disposal are two skidders, one forwarder, and one truck. Each kind of equipment can be used for 9 hours per day, and this time can be divided in any proportion between the two sites. The equipment needed to produce a cord of wood from each site varies as shown in the table below.

Table 11.1. Equipment hours needed to produce a cord of wood.

Site	Skidder	Forwarder	Truck
1	0.30	0.30	0.17
2	0.40	0.15	0.17

Answer: The first step is to define the variables. If you ask yourself, "what do I need to know to solve this problem?", you might answer the number of hours each piece of equipment must spend at each site per day. This would require 6 variables: 3 pieces of equipment times 2 sites. In this case, however, it would be much simpler to use two variables representing the production at each site.

If you know the daily production rate at each site, you can easily calculate the number of hours each piece of equipment will have to spend at each site. Thus, let:

X_1 = the number of cords per day to produce from site 1;
X_2 = the number of cords per day to produce from site 2.

The next step is to formulate the objective function. The problem statement makes it clear that the objective function in this example should be to maximize daily net revenues. Once again, to translate this conceptual objective into a mathematical function, start with the general form that the objective function must take:

Max $Z = c_1 \cdot X_1 + c_2 \cdot X_2$

Again, consider the units of the variables and the parameters. The objective is to maximize daily net revenues, so the units of Z must be \$/day. The units of both variables are cords/day. Therefore, the units of the coefficients must be \$/cord. The coefficient c_1 should give the net revenue per cord produced at site 1, and the coefficient c_2 should give the net revenue per cord produced at site 2. These values have already been given to us, so we can complete the formulation of the objective function. It is:

Max Z (\$/day) = 1.90 (\$/cd) · X_1 (cd/day) + 2.10 (\$/cd) · X_2 (cd/day)

The next step is to formulate the constraints. Each type of equipment is only available for a fixed number of hours per day. Consider the skidders first. There are two skidders that can work for 9 hours each. Thus a total of 18 skidder·hours are available each day. The right-hand-side of the skidder constraint will be 18 skidder·hours/day. Since the units of the variables are cords/day, the units of the coefficients must be skidder·hours/cord. These are the coefficients given in the table. Thus, the *skidder constraint* will be:

0.30 (skd-hrs/cd) ·X_1 (cd/day) + 0.40 (skd-hrs/cd) ·X_2 (cd/day) # 18 (skd-hrs/day)

The *forwarder constraint* can be obtained in an analogous way.

0.30 (fwd-hrs/cd) ·X_1 (cd/day) + 0.15 (fwd-hrs/cd) ·X_2 (cd/day) # 9 (fwd-hrs/day)

And the *truck constraint* is:

0.17 (trk-hrs/cd) ·X_1 (cd/day) + 0.17 (trk-hrs/cd) ·X_2 (cd/day) # 9 (trk-hrs/day)

Finally, we write the non-negativity constraints: X_1, X_2 $ 0.

The complete linear program for this problem is:

Max Z (\$/day) = 1.90 (\$/cd) · X_1 (cd/day) + 2.10 (\$/cd) · X_2 (cd/day)
(Objective function)

Subject to:

$0.30_{(skd\text{-}hrs/cd)} \cdot X_1$ _(cd/day)_ $+ 0.40$ _(skd-hrs/cd)_ $\cdot X_2$ _(cd/day)_ $\# 18$ _(skd-hrs/day)_

(Skidder constraint)

0.30 _(fwd-hrs/cd)_ $\cdot X_1$ _(cd/day)_ $+ 0.15$ _(fwd-hrs/cd)_ $\cdot X_2$ _(cd/day)_ $\# 9$ _(fwd-hrs/day)_

(Forwarder constraint)

0.17 _(trk-hrs/cd)_ $\cdot X_1$ _(cd/day)_ $+ 0.17$ _(trk-hrs/cd)_ $\cdot X_2$ _(cd/day)_ $\# 9$ _(trk-hrs/day)_

(Truck constraint)

$X_1,\ X_2 \$ 0$ (Non-negativity constraints)

Without the units, the linear programming formulation is:

Max $Z = 1.90 \cdot X_1 + 2.10 \cdot X_2$

Subject to:

$0.30 \cdot X_1 + 0.40 \cdot X_2 \# 18$
$0.30 \cdot X_1 + 0.15 \cdot X_2 \# 9$
$0.17 \cdot X_1 + 0.17 \cdot X_2 \# 9$
$X_1,\ X_2 \$ 0$

3. Graphical Solution of Two-Variable Linear Programming Problems

You have now seen how two word-problems can be translated into mathematical problems in the form of linear programs. Once a problem is formulated, it can be entered into a computer program to be solved. The *solution* is a set of values for each variable that:

     1. are consistent with the constraints (i.e., *feasible*), and
     2. result in the best possible value of the objective function (i.e., *optimal*).

Not all LP problems have a solution, however. There are two other possibilities:

     1. there may be no feasible solutions (i.e., there are no solutions that are consistent with all the constraints), or
     2. the problem may be *unbounded* (i.e., the optimal solution is infinitely large).

If the first of these problems occurs, one or more of the constraints will have to be relaxed. If the second problem occurs, then the problem probably has not been well formulated since few, if any, real world problems are truly unbounded.

As mentioned earlier, we will not be too concerned with how the computer solves linear programming problems. However, it is useful to solve a couple of simple problems graphically. This method only works with problems that have two variables, so obviously it has limited applicability. However, seeing the graphical representation and solution of a LP

problem will help you understand more intuitively what a LP is and how it is solved. Let's start by solving the Lumber Mill Problem.

Example — Graphical Solution of the Lumber Mill Problem.

Graphically solve the Lumber Mill Problem that was formulated earlier.

Answer: To graphically solve a two-variable linear program, we use a graph whose axes represent the values of the two variables. Figure 11.1 on the next page shows a graph where the y-axis represents the number of pallets produced per day (P) and the x-axis represents the number of thousand board feet (mbf) of lumber produced each day (L). Note that each point on this graph represents a combination of specific values of L and P. For example, the point **A** represents a combination of 150 mbf of lumber and 200 pallets. Each point can be identified by its coordinates as follows: (L, P). The point **A** has the coordinates (150, 200). Each point on the graph is a potential solution to the LP problem.

The first step is graphing the constraints. The first constraint says that lumber production must be less than 200 mbf/day. In Figure 11.1, this constraint is represented by the vertical line crossing the x-axis at 200 mbf. Since the constraint is a less-than-or-equal constraint, any point on the graph to the left side of this line satisfies this constraint.

The second constraint specifies that a maximum of 600 pallets can be produced per day. This constraint is represented in the graph by a horizontal line crossing the y-axis at 600. Again, since this is a less-than-or-equal constraint, any point on the graph that is below this line satisfies this constraint.

The third constraint, the log capacity constraint, is more difficult to graph because it involves both variables. This constraint is:

$$\tfrac{1}{4}{\cdot}P + 1.4{\cdot}L \ \# \ 400$$

The easiest way to graph this function is to set one variable equal to zero and solve for the value of other variable. This procedure gives you the points where the line representing the constraint crosses the x and y axes. If P is set equal to zero in the equation, then L must equal 285.7 (400 ÷ 1.4). This tells us that the constraint crosses the x-axis at the point (285.7, 0). Similarly, setting L equal to zero results in $P = 1,600$ (400 ÷ ¼). Thus, the constraint crosses the y-axis at the point (0, 1,600). With two points of the line plotted, we can draw the rest of the line by connecting the points with a ruler. Any point that is on this line or below and to the left of this line satisfies this constraint.
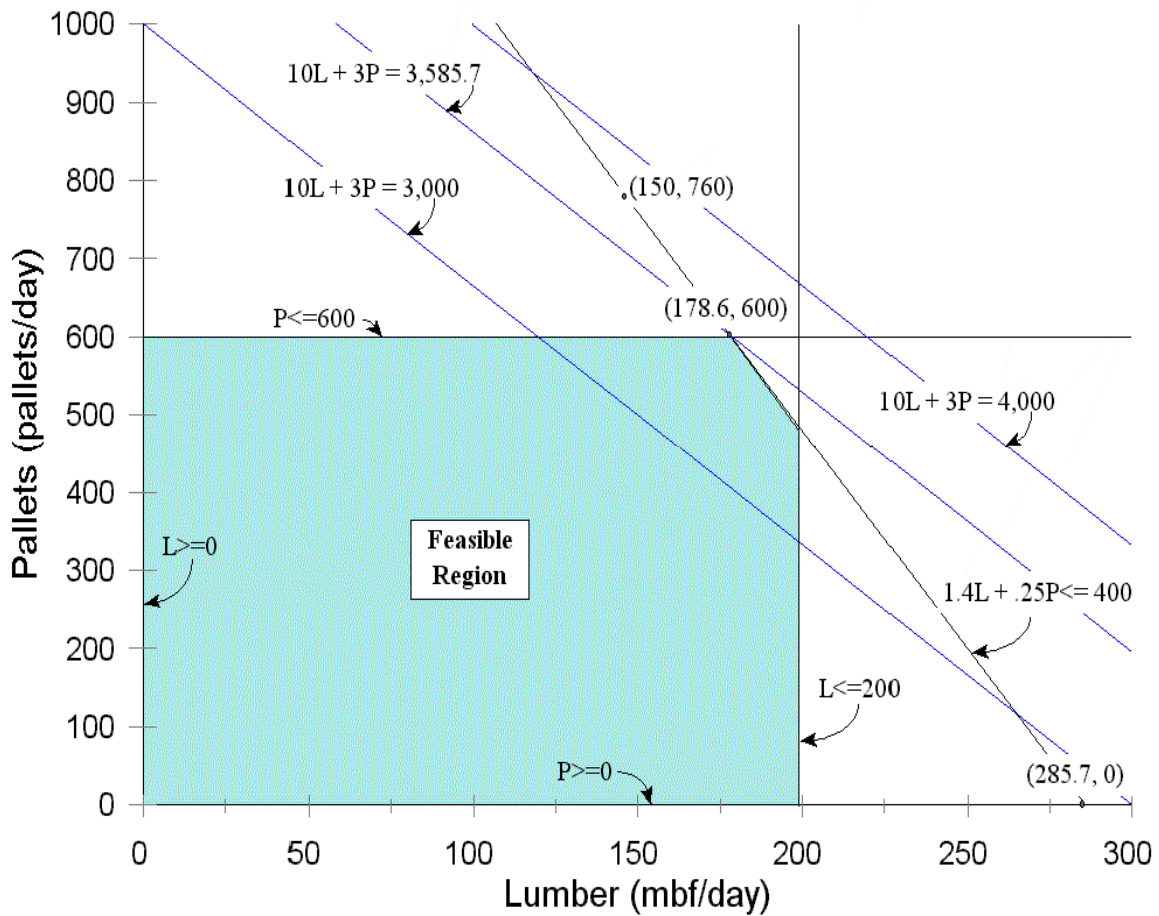
Figure 11.1.   Graphical Solution of the Lumber Mill Linear Programming Problem.

Sometimes one of these points is inconvenient to plot. The second point that was just identified is an example of this. The point (0, 1,600) is too far from the main part of the graph. In these cases, it may be better to plot a different point. Instead of fixing L at zero, L could also be fixed at 150. Plugging this value into the equation and solving for P gives $P = 760$ ([400 - 150·1.4]÷¼). In this case, the point (150, 760) could be used for plotting the line instead of (0, 1,600).

In addition to these three constraints, the non-negativity constraints should also be recognized on the graph. These constraints are already plotted, as they correspond to the axes themselves. Since the non-negativity constraints require the values of the variables to be positive, any values of the variables that are above the x-axis and to the right of the y-axis satisfy these constraints.

All of the constraints have now been identified on the graph. The constraints form a closed polygon containing all of the feasible solutions to the problem. Any point inside this polygon satisfies all of the constraints. This polygon is called the *feasible region*. Because the feasible region for this problem actually contains some points, we can conclude that there are feasible solutions to this problem – i.e., there are points, corresponding to different values of L and P, that satisfy all of the constraints. Also, the fact that the feasible region is closed indicates that the problem is not unbounded.

The problem now is to find the point (or set of points) within the feasible region that produces the highest value of the objective function. Unlike the constraints, the objective function does not correspond to a single line. Instead, it defines a series of parallel lines, each corresponding to a different value of the objective function. For example, the value of the objective function can be arbitrarily set at 3,000. That is, let

$$Z = 3{,}000 = 3 \cdot P + 10 \cdot L$$

The same techniques used earlier to graph the constraints can be used to graph this line. This line crosses the x-axis at the point (300,0) and the y-axis at the point (0, 1,000). Note where this line is in Figure 11.1. It crosses the feasible region between the points (120, 600) and (200, 333.3). (*Could you have identified these points?*) Any of the points on the line segment between these two points is feasible and will give an objective function value of 3,000.

Is 3,000 the highest objective function value that can be achieved? What happens if the objective function is set at a higher level, say 4,000. All of the points producing an objective function of 4,000 will fall on the following line:

$$Z = 4{,}000 = 3 \cdot P + 10 \cdot L$$

This line crosses the x-axis at the point (400, 0), and it crosses the y-axis at the point (0, 1,333.3). Locate the line in Figure 11.1. This line does not intersect the feasible region at any point. This means that there are no points that are feasible that give an objective function value of 4,000.

Note that the two lines identified by setting the objective function value to 3,000 and 4,000, respectively, are parallel. Each value that the objective function can take will correspond to a line on the graph. All of these lines defined by different values of the objective function will be parallel. Note also that as the objective function value increases, the line defined by the objective function moves farther out from the origin of the graph (the point (0, 0)). All of the space between the two objective function lines discussed so far can be filled with parallel lines corresponding to different values of the objective function between 3,000 and 4,000. The best possible value of the

objective function will correspond to the line that is as far from the origin as possible that still touches at least one point in the feasible region.

Imagine sliding the objective function line out from the origin by gradually increasing the objective function value. These lines will all be parallel to the lines that have already been drawn. Eventually, the line will move beyond the feasible region. The last point in the feasible region that is touched by the objective function as it is moved away from the origin is the optimal solution. The last feasible point that the line will touch will be one of the corners, or possibly the line segment (or *face*) between two corners. If the last point that is touched is a corner, then that corner is the optimal solution. If the line touches a face last, then every point on that face, including the two corners at the ends of the face, will be equally good. In either case, one or more of the corners will be in the optimal solution.

You can see by looking at the graph that the solution to the lumber mill problem will be at the corner where the pallet capacity constraint intersects the log capacity constraint. This corner corresponds to the optimal solution. Because this point is on the pallet capacity constraint, the value of P at this corner must be 600. The value of L at this point can be identified by setting P equal to 600 in the log capacity constraint. The values of the variables at this corner are $P = 600$ and $L = 178.6$. Thus, the solution to the lumber mill problem is:

$L = 178.6$ and $P = 600$.

In other words, the production strategy that will result in the highest daily net revenue is to produce 600 pallets and 178.6 mbf of lumber per day.

The best value of the objective function is obtained by plugging the values of the variables at this corner into the objective function.

$Z = 3 \cdot 600 + 10 \cdot 178.6 = 3{,}585.7$

The daily net revenue with this production strategy will therefore be $3,585.7. This value of the objective function — 3,585.7 — gives a line that just touches the feasible region at the point (178.6, 600). This line is shown in Figure 11.1.

Example — Graphical Solution of the Logging Problem

Use the graphical solution method to find the optimal solution to the Logging Problem.

Answer: Figure 11.2 shows the graphical solution of the logging problem. The axes of the logging problem correspond to the values of X_1 and X_2. As with the lumber mill problem, the constraints can be graphed by identifying two points on each line. The easiest points to identify are the points where the constraints cross the axes. The feasible region is defined by the area bounded by the constraints. Note that in this example, however, the truck constraint does not touch the feasible region. This is because the area defined by the skidder and forwarder constraints is fully within the area defined by the truck constraint. In other words, no matter how the skidders and the forwarder are utilized, it will not be possible to utilize the truck all of the time. In the terminology of LP, this means that the truck constraint will never be *binding*. (Note that in the lumber mill problem, the kiln capacity constraint was not binding at the optimal solution. In other words, not all of the available kiln capacity will be utilized in the optimal solution.)

Once the feasible region has been defined, the next step is to draw the objective function. Remember that the optimal solution will include at least one corner of the feasible region. The feasible region in this problem has four corners. We can easily rule out the origin (0, 0). The corners where the constraints cross the axes are easy to evaluate. When $X_1 = 0$ and $X_2 = 45$, the objective function value is 94.5 (45×2.1). When $X_2 = 0$ and $X_1 = 30$, the objective function value is 57 (30×1.9).

It is more difficult to identify the values of the variables at the corner where the skidder constraint intersects the forwarder constraint. To identify the values of the variables at this point, it is necessary to simultaneously solve the two constraints. First solve both constraints for X_2 :

$X_2 = 45 - 0.75 X_1$ *(Skidder constraint)*
$X_2 = 60 - 2 X_1$ *(Forwarder constraint)*

Now, since the left-hand side of both equations is equal to X_2, the right-hand sides must be equal. Thus,

$45 - 0.75 X_1 = 60 - 2 X_1$
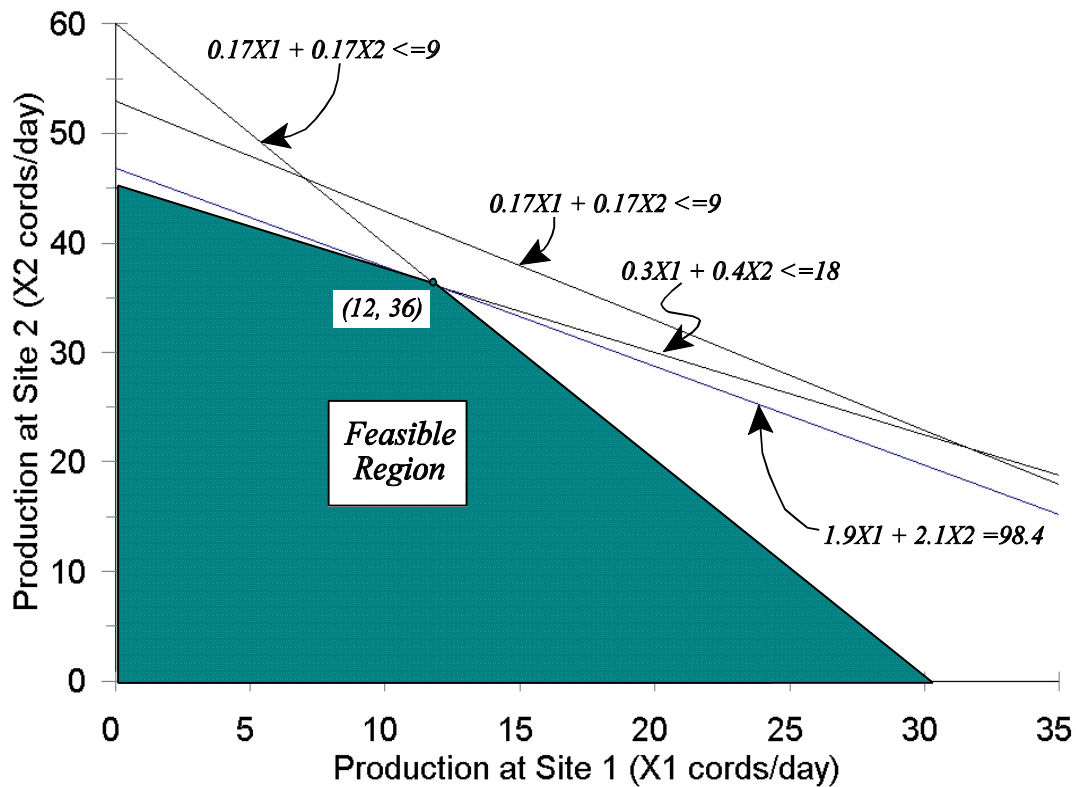
Figure 11.2. Graphical solution of the Logging Problem.

Now, solving for X_1, we get $X_1 = 12$. Substituting this result back into either of the original constraint equations reveals that $X_2 = 36$. The point where the two constraints cross is therefore (12, 36). Plugging these values into the objective function gives an objective function value of 98.4 (1.9×12 + 2.1×36). Since this is higher than the objective function value at the other corners, the solution to this problem is to produce 12 cords/day at site 1 and 36 cords per day at site 2. This will result in a daily net revenue of $98.4. The objective function line corresponding to this value has been graphed in Figure 11.2.

Three key points that you should have learned from the graphical solutions of the lumber mill and logging problems are:

1) the constraints should define a polygon (or, in the case of more than 2 variables, a *n*-dimensional polyhedron) called the feasible region;

2) the objective function defines a set of parallel lines (or with *n* variables, a set of *n*-dimensional hyperplanes) – one for each potential value of the objective function; and

3) the solution is the last corner or face of the feasible region that the objective function touches as the value of the objective function is improved.

This third point implies two important facts. First, the solution to a LP problem always includes at least one corner. Second, the solution is not always just a single point. If more than one corner point is optimal, then the face between those points is also optimal. The fact that the solution always includes a corner is used by the solution algorithm for solving LP problems. The algorithm searches from corner to corner, always looking for an adjacent corner that is better than the current corner. When a corner is found which has no superior adjacent corners, then that is reported as the solution. Some of the adjacent corners may be equally good, however.

An important concept introduced in the logging problem example is whether a constraint is *binding*. A constraint is said to be binding at points where it holds as an equality. For example, in the case of a less-than-or-equal constraint representing a resource limitation, the constraint is binding when all of the resource is being used. In this example, the truck constraint would never be binding because the skidders and the forwarder could not be operated enough to keep the truck fully occupied. At the optimal solution, the truck will be utilized for only 8.17 (0.17×12 + 0.17×36) hours each day. It will be idle for the remaining 0.83 hours. In the lumber mill problem, the kiln has the capacity to dry 200 mbf per day, but this capacity would not be fully utilized at the optimal solution because only 178.6 mbf will be produced per day. There is 21.4 mbf/day of excess kiln capacity. In LP terminology, this excess capacity is called *slack* capacity.

4. Interpreting Computer Solutions of Linear Programming Problems

The graphical solution method can only be applied to LP problems with two variables. For problems that are larger than this, we will rely on the computer to provide solutions. A variety of programs have been written to solve linear programming problems. This section discusses the output that a computer program called LINDO gives when it solves a linear program. You can also solve small linear programs with a spreadsheet, such as Excel. For larger linear programming problems, you will need a more specialized program, like LINDO. Appendix 11.2 discusses setting up, solving, and interpreting LP problems with Excel.

As discussed earlier, the solution to an LP problem is a set of optimal values for each of the variables. However, the output that comes with the solution to a LP problem usually contains much more information than just this. In addition to the optimal values of the variables, the output will typically include *reduced cost* values, *slack or surplus* values, and *dual prices* (also known as *shadow prices*).

Reduced Cost

Associated with each variable is a *reduced cost value*. However, the reduced cost value is only non-zero when the optimal value of a variable is zero. A somewhat intuitive way to think about the reduced cost variable is to think of it as indicating how much the cost of the activity represented by the variable must be reduced before any of that activity will be done. More precisely,

> ... the *reduced cost value* indicates how much the objective function coefficient on the corresponding variable must be improved before the value of the variable will be positive in the optimal solution.

In the case of a minimization problem, "improved" means "reduced." So, in the case of a cost-minimization problem, where the objective function coefficients represent the per-unit cost of the activities represented by the variables, the "reduced cost" coefficients indicate how much each cost coefficient would have to be reduced before the activity represented by the corresponding variable would be cost-effective. In the case of a maximization problem, "improved" means "increased." In this case, where, for example, the objective function coefficient might represent the net profit per unit of the activity, the reduced cost value indicates how much the profitability of the activity would have to increase in order for the activity to occur in the optimal solution. The units of the reduced cost values are the same as the units of the corresponding objective function coefficients.

If the optimal value of a variable is positive (not zero), then the reduced cost is always zero. If the optimal value of a variable is zero and the reduced cost corresponding to the variable is also zero, then there is at least one other corner that is also in the optimal solution. The value of this variable will be positive at one of the other optimal corners.

Slack or Surplus

A *slack* or *surplus* value is reported for each of the constraints. The term "slack" applies to less than or equal constraints, and the term "surplus" applies to greater than or equal constraints. If a constraint is binding, then the corresponding slack or surplus value will equal zero. When a less-than-or-equal constraint is not binding, then there is some un-utilized, or slack, resource.

> The *slack* value is the amount of a resource, as represented by a less-than-or-equal constraint, that is not being used. When a greater-than-or-equal constraint is not binding, then the *surplus* is the extra amount over the constraint that is being produced or utilized.

The units of the slack or surplus values are the same as the units of the corresponding constraints.

Dual Prices (a.k.a. Shadow Prices)

The *dual prices* are some of the most interesting values in the solution to a linear program. A dual price is reported for each constraint. The dual price is only positive when a constraint is binding.

> The *dual price* gives the improvement in the objective function if the constraint is relaxed by one unit.

In the case of a less-than-or-equal constraint, such as a resource constraint, the dual price gives the value of having one more unit of the resource represented by that constraint. In the case of a greater-than-or-equal constraint, such as a minimum production level constraint, the dual price gives the cost of meeting the last unit of the minimum production target.

The units of the dual prices are the units of the objective function divided by the units of the constraint. Knowing the units of the dual prices can be useful when you are trying to interpret what the dual prices mean.

> **Example** — The Lumber Mill Problem
>
> The LINDO input file for the lumber mill problem looks like this.

```
Batch
Max 3P + 10L
ST
L <=  200
P <=  600
0.25P + 1.4L <= 400
end
```

> The solution will look something like this:

```
LP OPTIMUM FOUND AT STEP        2

        OBJECTIVE FUNCTION VALUE

        1)       3585.714

    VARIABLE        VALUE          REDUCED COST
          P       600.000000            .000000
          L       178.571400            .000000


        ROW     SLACK OR SURPLUS      DUAL PRICES
         2)          21.428570            .000000
         3)            .000000           1.214286
         4)            .000000           7.142857

    NO. ITERATIONS=        2
```

> The first line of the solution indicates how many iterations of the solution process were needed to find the solution. An iteration corresponds to a corner

of the feasible region that LINDO checks to see if it is the optimal solution. (For our purposes, it doesn't matter much how many iterations it takes LINDO to solve a problem.) The third line of the output gives the value of the objective function at the optimal solution. In this case, it is $3,585.714/day, as we already determined from our graphical solution.

The next block of information in the output gives the optimal values and the reduced cost values for each variable. It tells us that the optimal solution is to produce 600 pallets and 178.5714 mbf per day. Since both variable values are positive, the reduced cost values are zero.

The last block of the output lists the slack or surplus values and the dual prices for each constraint. Note that the slack value for the first constraint — the kiln capacity constraint — is 21.429. This is the excess capacity of the kiln in mbf per day. Since both of the next two constraints are binding, their slack values are zero. However, their dual prices are non-zero. The dual price on the second constraint is $1.2143/day per pallet. It says that the objective function value — the net revenue per day — could be increased by this amount if one more pallet could be produced each day. Another way of looking at this value is that the marginal net revenue on the last pallet produced was $1.2143. Similarly, the dual price on the third constraint says the objective function value could be increased by $7.1429 if one more log could be processed.

Example — The Logging Problem

The LINDO input file for the logging problem looks like this.

```
batch
Max 1.90X1 + 2.10X2
St
0.30 X1 + 0.40 X2 <=  18
0.30 X1 + 0.15 X2 <=  9
0.17 X1 + 0.17 X2 <=  9
end
```

The solution will look something like this:

```
LP OPTIMUM FOUND AT STEP      2

        OBJECTIVE FUNCTION VALUE

    1)       98.39999

   VARIABLE        VALUE          REDUCED COST
        X1       12.000000           .000000
        X2       36.000000           .000000


      ROW    SLACK OR SURPLUS      DUAL PRICES
       2)           .000000         4.599999
       3)           .000000         1.733333
       4)           .840000          .000000

   NO. ITERATIONS=       2
```

It took two iterations of the solution process to find the solution to this problem. The objective function at the optimal solution is $98.4/day. The optimal solution is to produce 12 cords at site 1 and 36 cords at site 2 each day. Again, since both variable values are positive, the reduced cost values are zero. Both the skidder and the forwarder constraints are binding. The dual price on the skidder constraint says that one more skidder-hour per day would allow the logger to increase his daily net revenue by $4.60. The dual price on the forwarder says that one more forwarder-hour would allow the logger to increase his daily net revenue by $1.733. Finally, as we have already discussed, the truck constraint is not binding, and the truck will be idle for 0.84 hours each day.

5. The Fundamental Assumptions of Linear Programming

Now that you have seen how some simple problems can be formulated and solved as linear programs, it is useful to reconsider the question of when a problem can be realistically represented as a linear programming problem. A problem can be realistically represented as a linear program if the following assumptions hold:

1. The constraints and objective function are <u>linear</u>.

    a) This requires that the value of the objective function and the response of each resource expressed by the constraints is <u>proportional</u> to the level of each activity expressed in the variables.

    b) Linearity also requires that the effects of the value of each variable on the values of the objective function and the constraints are <u>additive</u>. In other

words, there can be no interactions between the effects of different activities; i.e., the level of activity X_1 should not affect the costs or benefits associated with the level of activity X_2.

2. <u>Divisibility</u> -- the values of decision variables can be fractions. Sometimes these values only make sense if they are integers; then we need an extension of linear programming called integer programming.

3. <u>Certainty</u> -- the model assumes that the responses to the values of the variables are exactly equal to the responses represented by the coefficients.

4. <u>Data</u> -- formulating a linear program to solve a problem assumes that data are available to specify the problem.

6. <u>Study Questions for Linear Programming</u>

1. What is a linear program? What are the elements of a linear program?

2. What is the difference between a parameter and a variable?

3. What is the difference between formulating and solving a linear programming problem?

4. What does it mean to *formulate* a linear program (... as opposed to solving one)?

5. What are the basic steps in formulating a linear program?

6. Why are units so important in linear programming?

7. What does it mean to say that a solution optimal? What does it mean to say that a solution is feasible?

8. What are two possible reasons why a LP does not have a solution?

9. What is the *feasible region* of an LP?

10. How can it happen that a linear program will have more than one optimal solution?

11. What does it mean when a constraint is not binding?

12. What is the interpretation of the reduced cost values? . . . the dual prices?

13. Why is the reduced cost always zero when the optimal value of the corresponding variable is positive?

14. Why is the dual price only positive when the corresponding constraint is binding?

15. What does it mean when the reported optimal value of a variable is zero and the reduced cost value for that variable is also zero?

7. Exercises

*1.   A logger has timber rights to two stands. The logger has contracted to provide a mill with 12 units of jack pine, 8 units of birch, and 24 units of aspen. It costs $300 /day to operate in stand 1 and $160/day to operate in stand 2. In a day's operation in stand 1, 6 units of jack pine, 2 units of birch and 4 units of aspen can be harvested and delivered. Similar numbers for stand 2 are 2 units of jack pine, 2 units of birch and 12 units of aspen.

a. Describe, in words only, the logger's objective.

b. Define the decision variables you will use in formulating this problem as a linear program. Be sure to identify the units of the decision variables.

c. Write out the linear programming formulation of the problem. Include the units of the variables and the coefficients in your description. Don't forget the non-negativity constraints.

d. On a piece of graph paper, use the graphic solution method (and a ruler) to solve the linear programming model developed in part **c**. Be sure to label all axes, constraints, and your objective function. Write the solution to the problem and the value of the objective function here:

e. Solve the problem from part **c** using Excel. Attach a printout of your Excel spreadsheet (formulation page, answer report and sensitivity report).

f. Solve the problem from part **c** using LINDO. Basic LINDO commands are described in an appendix at the back of this chapter. Attach a printout of your LINDO input and output files.

g. How much excess of each timber type is produced under the optimal solution?

h. How much could total costs be reduced if the logger could sell excess jack pine for $25/unit?

i. Suppose the logger could purchase birch from an external source for $40/unit. Should the logger purchase any units? How many? How will this affect the logger's costs?

j. How much could net costs (costs minus aspen revenues) be reduced if the logger could sell excess aspen for $10/unit? What will the solution be under this scenario?

k. What would be the optimal number of days to operate in each stand if one unit of birch is purchased by the logger to help meet the contracted demands?

l. Interpret the coefficient "+35" for excess jack pine in the final set of equations.

m. Write out the dual to the problem you formulated in part a.

n. Give a verbal interpretation of the dual to the problem in part a.

*2. The Primal Paper Company can produce low, medium and high grades of paper. Each carload of low grade paper produces a net revenue of $700, each carload of medium grade paper produces a net revenue of $840, and each carload of high grade paper produces a net revenue of $900. For the present planning period Primal Paper has 92 units of aspen and 36 units of pine to process. To produce a carload of low grade paper, 3 units of aspen and 1 unit of pine are required. To produce a carload of medium grade paper, 2 units of aspen and 2 units of pine are required. To produce a carload of high grade paper, 1 units of aspen and 3 units of pine are required. Primal Paper wants to determine how many carloads or partial carloads of each paper grade to produce so that net revenue is maximized.

a. Define the decision variables you will use in formulating this problem as a linear program. Be sure to identify the units of the decision variables.

b. On a separate piece of paper, write out the linear programming formulation of the problem. <u>Include the units of the variables and the coefficients in your description</u>. Don't forget the non-negativity constraints.

c. Solve the problem from part b using Excel. Attach a printout of your Excel spreadsheet (formulation page, answer report and sensitivity report).

d. What is the solution to Primal Paper's problem?

3. A logger wants to maximize the net revenues per hour that she earns with her four feller-bunchers and six skidders. From her records, she estimates her net revenue per hour of operation for a feller-buncher at $3, and for a skidder at $6. She currently has only 18 people trained to run logging equipment. This limits her operations, since it takes two people to run a skidder and three to run a feller-buncher.

a. Define the decision variables you will use in formulating this problem as a linear program. Be sure to identify the units of the decision variables.

b. Formulate this problem as a linear program. Include the units of the variables <u>and the coefficients</u> in your description. Don't forget the nonnegativity constraints.

c. On a piece of graph paper, use the graphic solution method to solve the linear programming model developed in part a. Be sure and label all axes, constraints, the objective function, and the solution.

d. Write out the mathematical formulation of the dual to the logger's problem.

e. Describe in words the dual to the logger's problem; i.e., provide a verbal interpretation of the dual problem.

4. The ABC furniture company manufactures tables, chairs, desks, and bookcases. A table requires 5 feet of softwood and 2 feet of hardwood, and takes 3 hours of labor to produce. A chair requires 1 foot of softwood, 3 feet of hardwood, and 2 hours of labor; a desk requires 9 feet of softwood, 4 feet of hardwood, and 5 hours of labor; finally, a bookcase requires 12 feet of softwood, 1 foot of hardwood, and 10 hours of labor to produce.

The company has available only 1500 feet of softwood, and 1000 feet of hardwood. The factory employs 10 people, each of which works 8 hours per day; overtime is not permitted. The firm plans its production for a 10-day period, according to the estimated demands projected by the sales manager. The sales manager has asked for at least 40 tables, 130 chairs, and 30 desks, and says he can sell any amount in excess of these minimum requirements. There is little demand for bookcases, so there is no minimum requirement, but the sales manager does not think he could accept more than 10 for sale.

Softwood costs $2 per foot, hardwood $5 per foot, and labor $10 per hour. These products can be sold for the following prices: table, $62/unit; chair, $42/unit; desk, $103/unit; and bookcase, $139/unit.

a. Formulate this problem up as a linear program. Be sure to define all of your variables and their units.

b. Create a LINDO input file for the ABC Furniture Company's problem that you formulated in part a. Attach a print-out of your file to your homework.

c. Use LINDO to solve the problem. Print a copy of your output file and attach it to this homework.

d. How much more money could the company make if the sales manager could live with only 80 chairs?

e. How much more money could the company make if it could hire 3 new workers at $10 per hour? (Assume that they must produce 130 chairs.)

f. Would it be a good idea for the company to hire 4 new workers? Why or why not?

g. How much would bookcases have to sell for before it would be profitable for the company to make some bookcases?

5. The forestry club is planning its fund raising activities for the year. Two major activities are being considered: 1) selling t-shirts and hats and 2) selling Christmas trees. The club wants to make as much income as it can. T-shirts, hats and trees must be contracted for in advance, so current-year revenues cannot be spent on purchasing this year's trees, t-shirts and hats. However, the club has $2,610 left over after last year's conclave that can be used. Both t-shirts and hats can be purchased for $6 and can be sold for $14. Trees can be purchased for $9 each and can be sold for $30. The trees will have to be cut down and shipped to the university, where they will be sold. Only 2 weekends are available for cutting trees, and the maximum number of trees that can be harvested with the available labor is 240 trees. The club members also estimate that they can sell at most 100 t-shirts and 50 hats. The club members will have to volunteer time for selling both Christmas trees and t-shirts and hats. On average, it takes about 12 minutes of sales time per tree sold and 20 minutes of sales time per t-shirt or hat. Club members will be able to volunteer a maximum of 80 hours of sales time.

    a. Define the decision variables you will use in formulating this problem as a linear program. Be sure to identify the units of the decision variables.

    b. Write out the linear programming formulation of the problem. Include the units of the variables and the coefficients in your description. Don't forget the non-negativity constraints.

    c. On a piece of graph paper, use the graphic solution method (and a ruler) to solve the linear programming model developed in part b. Be sure to label all axes, constraints, and your objective function. Write the solution to the problem and the value of the objective function here:

    d. How would the solution change if one of the club members decided to lend the club $150 (with no interest) until after the Christmas tree sale? (What would the new optimal solution be? What would the value of the objective function be?)

6. The Pacific International Corporation produces two type of paper, both newsprint and magazine quality. Both softwood and hardwood are used as raw materials in different combinations to produce a ton of each quality paper. To produce one ton of newsprint, 1 ton of hardwood and 2 tons of softwood are required. Similarly, to produce one ton of magazine grade paper, 2 tons of hardwood and 1 ton of softwood are necessary. The sale price per ton of newsprint is $600

    a. Define the decision variables you will use in formulating this problem as a linear program. Be sure to identify the units of the decision variables.

    b. Write out the linear programming formulation of the problem. Include the units of the variables and the coefficients in your description. Don't forget the non-negativity constraints.

c. On a piece of graph paper, use the graphic solution method (and a ruler) to solve the linear programming model developed in part b. Be sure to label all axes, constraints, and your objective function. Write the solution to the problem and the value of the objective function here:

d. How would the solution change if one of the club members decided to lend the club $150 (with no interest) until after the Christmas tree sale? (What would the new optimal solution be? What would the value of the objective function be?)

APPENDIX 11.1: Using LINDO to Solve LP Problems.

LINDO can be instructed to read a text file like the ones presented earlier by typing the TAKE command at LINDO's prompt line. The file must start with the command BATCH, which tells LINDO to read all of the file and that it contains an LP formulation. The rest of the file is written pretty much the way you would write the problem formulation. The second line gives the objective function; it can be broken up over several lines if it gets too long. The ST command ("subject to") following the objective function indicates that the remaining lines are constraints. As with the objective function, a constraint can be broken up onto more than one line if it gets too long. Finally, the END command tells LINDO that the whole problem has been read.

After LINDO has read the file, you can have LINDO solve the problem by typing GO at LINDO's prompt. LINDO will then write the solution to the problem to the screen. At this point, LINDO will ask you whether you want to "do a range-sensitivity analysis." While this analysis gives some useful information, we will not cover the type of output presented in the range/sensitivity analysis in this class. You should generally answer "no" to this question.

You can have LINDO write the solution to a problem to a file by typing the DIVE command at the prompt, followed by a file name. It is a good idea to use the LOOK ALL command to have LINDO write the problem formulation to this file. Then issue the GO command to have LINDO write the solution in the file.

APPENDIX 11.2: Using Excel to Solve LP Problems.

This section explains how to solve a problem using Microsoft Excel's linear programming solver. The lumber mill problem will be used as an example.

Set up a table in an Excel workbook with the following rows (there will be one column in your table for each variable, one column for your right-hand-side coefficients, and one column for equations):

1. Variable names (1 row),
2. Variable values (1 row),
3. Objective function coefficients (1 row), and
4. Constraint coefficients (1 row for each constraint).

The table for the sawmill problem would look like this:

Table 11.A2.1. Excel workbook with LP tableau for the lumber mill problem.

	A	B	C	D	E
1	Variable Names	P	L	RHS	Equations
2	Variable Values	0	0		
3	Obj. Fn. Coefficients	3	10		=B3*B$2+C3*C$2
4	Kiln Constr. Coeff.	0	1	200	=B4*B$2+C4*C$2
5	Pallet Constr. Coeff.	1	0	600	=B5*B$2+C5*C$2
6	Log Cap. Constr. Coeff.	0.25	1.4	400	=B6*B$2+C6*C$2

Once you have your data set up, you call up Excel's *Solver* under the *Tools* menu. Within the *Solver* form, you will need to specify the following information:

1. the target cell (in our example, this would be E3, which gives the objective function value);
2. whether the problem is a maximization or minimization problem (min, in this case);
3. the variable values (under By changing cells – B2 to C2 in this case); and
4. the constraints.

To specify the constraints for the problem, click the Add button. For the "Cell reference," specify the location of the constraint equation. (Our first constraint equation is in cell E4.) Then, specify whether the constraint is a less-than-or-equal-to, equal-to, or greater-than-or-equal-to constraint. (All of ours are less-than-or-equal-to constraints.) Finally, under "Constraint," specify the cell with the right-hand-side coefficient for that constraint. (For our first constraint, this would be D4.) After a constraint has been specified, press the Add key to enter the next constraint. When all of the constraints have been specified, press the "OK" button.

One last important step that you should perform before asking Excel to solve the problem requires opening the Options dialog box. Under options, check the "Assume Linear Model" and "Assume Non-Negative" check-boxes. Click "OK" to close the Options dialog box, and click on Solve to have Excel solve the problem.

Hopefully, the Solver will tell you that it has found a solution. If not, you will need to figure out what went wrong. It would be impossible to cover all of the possibilities here. Assuming that the Solver did find a solution, make sure that the "Keep Solver Solution" radio button is selected and click on the "Answer" and "Sensitivity" reports.

The "Answer Report" given by Excel for this problem is shown in Table 11.A2.1, and the "Sensitivity Report" given by Excel is shown in Table 11.A2.2. The Answer Report gives the optimal values of the variables under the heading "Final Value." The slack or surplus values are also reported in the Answer Report. The reduced cost values and dual prices are found in the Sensitivity Report. Note, however, that Excel calls these "shadow prices," rather than dual prices.

Table 11.A2.2. Excel "Answer Report" for the Lumber Mill Problem
Microsoft Excel 8.0 Answer Report
Worksheet: [Book2]Sheet1
Report Created: 11/5/98 4:05:16 PM

Target Cell (Max)

Cell	Name	Original Value	Final Value
$E$4		2300	3585.714286

Adjustable Cells

Cell	Name	Original Value	Final Value
$B$3	P	100	600
$C$3	L	200	178.5714286

Constraints

Cell	Name	Cell Value	Formula	Status	Slack
$E$5		178.5714286	$E$5<=$D$5	Not Binding	21.42857143
$E$6		600	$E$6<=$D$6	Binding	0
$E$7		400	$E$7<=$D$7	Binding	0

Table 11.A2.3. Excel "Sensitivity Report" for the Lumber Mill Problem

Microsoft Excel 8.0 Sensitivity Report
Worksheet: [Book2]Sheet1
Report Created: 11/5/98 4:05:16 PM

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$3	P	600	0	3	1E+30	1.214285714
$C$3	L	178.5714286	0	10	6.8	10

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$E$5		178.5714286	0	200	1E+30	21.42857143
$E$6		600	1.214285714	600	1000	120
$E$7		400	7.142857143	400	30	250