Integer Programming

Recall that we defined integer programming problems in our discussion of the Divisibility Assumption in Section 3.1. Simply stated, an *integer programming problem* (IP) is an LP in which some or all of the variables are required to be non-negative integers.†

In this chapter (as for LPs in Chapter 3), we find that many real-life situations may be formulated as IPs. Unfortunately, we will also see that IPs are usually much harder to solve than LPs.

In Section 9.1, we begin with necessary definitions and some introductory comments about IPs. In Section 9.2, we explain how to formulate integer programming models. We also discuss how to solve IPs on the computer with LINDO, LINGO, and Excel Solver. In Sections 9.3–9.8, we discuss other methods used to solve IPs.

9.1 Introduction to Integer Programming

An IP in which all variables are required to be integers is called a **pure integer programming problem.** For example,

$$\begin{aligned} \max z &= 3x_1 + 2x_2 \\ \text{s.t.} \quad x_1 + x_2 &\le 6 \\ x_1, x_2 \ge 0, &\; x_1, x_2 \text{ integer} \end{aligned} \tag{1}$$

is a pure integer programming problem.

An IP in which only some of the variables are required to be integers is called a **mixed integer programming problem.** For example,

$$\begin{aligned} \max z &= 3x_1 + 2x_2 \\ \text{s.t.} \quad x_1 + x_2 &\le 6 \\ x_1, x_2 \ge 0, &\; x_1 \text{ integer} \end{aligned}$$

is a mixed integer programming problem (x_2 is not required to be an integer).

An integer programming problem in which all the variables must equal 0 or 1 is called a 0–1 IP. In Section 9.2, we see that 0–1 IPs occur in surprisingly many situations.‡ The following is an example of a 0–1 IP:

$$\begin{aligned} \max z &= x_1 - x_2 \\ \text{s.t.} \quad x_1 + 2x_2 &\le 2 \\ 2x_1 - x_2 &\le 1 \\ x_1, x_2 &= 0 \text{ or } 1 \end{aligned} \tag{2}$$

Solution procedures especially designed for 0–1 IPs are discussed in Section 9.7.

†A nonlinear integer programming problem is an optimization problem in which either the objective function or the left-hand side of some of the constraints are nonlinear functions and some or all of the variables must be integers. Such problems may be solved with LINGO or Excel Solver.

‡Actually, any pure IP can be reformulated as an equivalent 0–1 IP (Section 9.7).

The concept of LP relaxation of an integer programming problem plays a key role in the solution of IPs.

DEFINITION ■ The LP obtained by omitting all integer or 0–1 constraints on variables is called the **LP relaxation** of the IP. ■

For example, the LP relaxation of (1) is

$$\begin{aligned} \max z &= 3x_1 + 2x_2 \\ \text{s.t.} \quad x_1 + x_2 &\le 6 \\ x_1, x_2 &\ge 0 \end{aligned} \tag{1'}$$

and the LP relaxation of (2) is

$$\begin{aligned} \max z &= x_1 - x_2 \\ \text{s.t.} \quad x_1 + 2x_2 &\le 2 \\ 2x_1 - x_2 &\le 1 \\ x_1, x_2 &\ge 0 \end{aligned} \tag{2'}$$

Any IP may be viewed as the LP relaxation plus additional constraints (the constraints that state which variables must be integers or be 0 or 1). Hence, the LP relaxation is a less constrained, or more relaxed, version of the IP. This means that *the feasible region for any IP must be contained in the feasible region for the corresponding LP relaxation.* For any IP that is a max problem, this implies that

$$\text{Optimal } z\text{-value for LP relaxation} \ge \text{optimal } z\text{-value for IP} \tag{3}$$

This result plays a key role when we discuss the solution of IPs.

To shed more light on the properties of integer programming problems, we consider the following simple IP:

$$\begin{aligned} \max z &= 21x_1 + 11x_2 \\ \text{s.t.} \quad 7x_1 + 4x_2 &\le 13 \\ x_1, x_2 \ge 0;\ & x_1, x_2 \text{ integer} \end{aligned} \tag{4}$$

From Figure 1, we see that the feasible region for this problem consists of the following set of points: $S = \{(0, 0), (0, 1), (0, 2), (0, 3), (1, 0), (1, 1)\}$. Unlike the feasible region for any LP, the one for (4) is not a convex set. By simply computing and comparing the z-values for each of the six points in the feasible region, we find the optimal solution to (4) is $z = 33$, $x_1 = 0$, $x_2 = 3$.

If the feasible region for a pure IP's LP relaxation is bounded, as in (4), then the feasible region for the IP will consist of a finite number of points. In theory, such an IP could be solved (as described in the previous paragraph) by enumerating the z-values for each feasible point and determining the feasible point having the largest z-value. The problem with this approach is that most actual IPs have feasible regions consisting of billions of feasible points. In such cases, a complete enumeration of all feasible points would require a large amount of computer time. As we explain in Section 9.3, IPs often are solved by cleverly enumerating all the points in the IP's feasible region.

Further study of (4) sheds light on other interesting properties of IPs. Suppose that a naive analyst suggests the following approach for solving an IP: First solve the LP relaxation; then round off (to the nearest integer) each variable that is required to be an integer and that assumes a fractional value in the optimal solution to the LP relaxation.

Applying this approach to (4), we first find the optimal solution to the LP relaxation: $x_1 = \frac{13}{7}$, $x_2 = 0$. Rounding this solution yields the solution $x_1 = 2$, $x_2 = 0$ as a possible

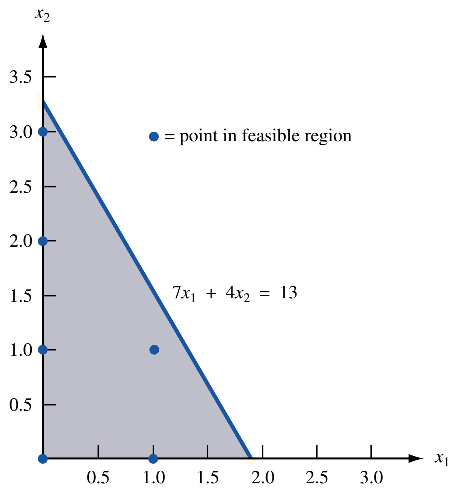


FIGURE 1
Feasible Region for Simple IP (4)

optimal solution to (4). But $x_1 = 2$, $x_2 = 0$ is infeasible for (4), so it cannot possibly be the optimal solution to (4). Even if we round x_1 downward (yielding the candidate solution $x_1 = 1$, $x_2 = 0$), we do not obtain the optimal solution ($x_1 = 0$, $x_2 = 3$ is the optimal solution).

For some IPs, it can even turn out that every roundoff of the optimal solution to the LP relaxation is infeasible. To see this, consider the following IP:

$$\begin{aligned} \max z &= 4x_1 + x_2 \\ \text{s.t.} \quad & 2x_1 + x_2 \le 5 \\ & 2x_1 + 3x_2 = 5 \\ & x_1, x_2 \ge 0;\ x_1, x_2 \text{ integer} \end{aligned}$$

The optimal solution to the LP relaxation for this IP is $z = 10$, $x_1 = \frac{5}{2}$, $x_2 = 0$. Rounding off this solution, we obtain either the candidate $x_1 = 2$, $x_2 = 0$ or the candidate $x_1 = 3$, $x_2 = 0$. Neither candidate is a feasible solution to the IP.

Recall from Chapter 4 that the simplex algorithm allowed us to solve LPs by going from one basic feasible solution to a better one. Also recall that in most cases, the simplex algorithm examines only a small fraction of all basic feasible solutions before the optimal solution is obtained. This property of the simplex algorithm enables us to solve relatively large LPs by expending a surprisingly small amount of computational effort. Analogously, one would hope that an IP could be solved via an algorithm that proceeded from one feasible integer solution to a better feasible integer solution. Unfortunately, no such algorithm is known.

In summary, even though the feasible region for an IP is a subset of the feasible region for the IP's LP relaxation, the IP is usually much more difficult to solve than the IP's LP relaxation.

9.2 Formulating Integer Programming Problems

In this section, we show how practical solutions can be formulated as IPs. After completing this section, the reader should have a good grasp of the art of developing integer programming formulations. We begin with some simple problems and gradually build to more complicated formulations. Our first example is a capital budgeting problem reminiscent of the Star Oil problem of Section 3.6.

EXAMPLE 1 Capital Budgeting IP

Stockco is considering four investments. Investment 1 will yield a net present value (NPV) of \$16,000; investment 2, an NPV of \$22,000; investment 3, an NPV of \$12,000; and investment 4, an NPV of \$8,000. Each investment requires a certain cash outflow at the present time: investment 1, \$5,000; investment 2, \$7,000; investment 3, \$4,000; and investment 4, \$3,000. Currently, \$14,000 is available for investment. Formulate an IP whose solution will tell Stockco how to maximize the NPV obtained from investments 1–4.

Solution As in LP formulations, we begin by defining a variable for each decision that Stockco must make. This leads us to define a 0–1 variable:

$$x_j (j = 1, 2, 3, 4) = \begin{cases} 1 & \text{if investment } j \text{ is made} \\ 0 & \text{otherwise} \end{cases}$$

For example, $x_2 = 1$ if investment 2 is made, and $x_2 = 0$ if investment 2 is not made.

The NPV obtained by Stockco (in thousands of dollars) is

$$\text{Total NPV obtained by Stockco} = 16x_1 + 22x_2 + 12x_3 + 8x_4 \qquad (5)$$

To see this, note that if $x_j = 1$, then (5) includes the NPV of investment j, and if $x_j = 0$, (5) does not include the NPV of investment j. This means that whatever combination of investments is undertaken, (5) gives the NPV of that combination of projects. For example, if Stockco invests in investments 1 and 4, then an NPV of 16,000 + 8,000 = \$24,000 is obtained. This combination of investments corresponds to $x_1 = x_4 = 1$, $x_2 = x_3 = 0$, so (5) indicates that the NPV for this investment combination is 16(1) + 22(0) + 12(0) + 8(1) = \$24 (thousand). This reasoning implies that Stockco's objective function is

$$\max z = 16x_1 + 22x_2 + 12x_3 + 8x_4 \qquad (6)$$

Stockco faces the constraint that at most \$14,000 can be invested. By the same reasoning used to develop (5), we can show that

$$\text{Total amount invested (in thousands of dollars)} = 5x_1 + 7x_2 + 4x_3 + 3x_4 \qquad (7)$$

For example, if $x_1 = 0$, $x_2 = x_3 = x_4 = 1$, then Stockco makes investments 2, 3, and 4. In this case, Stockco must invest 7 + 4 + 3 = \$14 (thousand). Equation (7) yields a total amount invested of 5(0) + 7(1) + 4(1) + 3(1) = \$14 (thousand). Because at most \$14,000 can be invested, x_1, x_2, x_3, and x_4 must satisfy

$$5x_1 + 7x_2 + 4x_3 + 3x_4 \le 14 \qquad (8)$$

Combining (6) and (8) with the constraints $x_j = 0$ or 1 ($j = 1, 2, 3, 4$) yields the following 0–1 IP:

$$\begin{aligned} \max z &= 16x_1 + 22x_2 + 12x_3 + 8x_4 \\ \text{s.t.} \quad & 5x_1 + 7x_2 + 4x_3 + 3x_4 \le 14 \\ & x_j = 0 \text{ or } 1 \quad (j = 1, 2, 3, 4) \end{aligned} \qquad (9)$$

REMARKS 1 In Section 9.5, we show that the optimal solution to (9) is $x_1 = 0$, $x_2 = x_3 = x_4 = 1$, $z =$ \$42,000. Hence, Stockco should make investments 2, 3, and 4, but not 1. Investment 1 yields a higher NPV per dollar invested than any of the others (investment 1 yields \$3.20 per dollar invested, investment 2, \$3.14; investment 3, \$3; and investment 4, \$2.67), so it may seem surprising that investment 1 is not undertaken. To see why the optimal solution to (9) does not involve making the "best" investment, note that any investment combination that includes investment 1 cannot use more than \$12,000. This means that using investment 1 forces Stockco to forgo investing \$2,000. On the other hand, the optimal investment combination uses all \$14,000 of the investment budget. This en-

TABLE 1
Weights and Benefits for Items in Josie's Knapsack

Item	Weight (Pounds)	Benefit
1	5	16
2	7	22
3	4	12
4	3	8

ables the optimal combination to obtain a higher NPV than any combination that includes investment 1. If, as in Chapter 3, fractional investments were allowed, the optimal solution to (9) would be $x_1 = x_2 = 1$, $x_3 = 0.50$, $x_4 = 0$, $z = \$44{,}000$, and investment 1 would be used. This simple example shows that the choice of modeling a capital budgeting problem as a linear programming or as an integer programming problem can significantly affect the optimal solution to the problem.

2 Any IP, such as (9), that has only one constraint is referred to as a **knapsack problem.** Suppose that Josie Camper is going on an overnight hike. There are four items Josie is considering taking along on the trip. The weight of each item and the benefit Josie feels she would obtain from each item are listed in Table 1.

Suppose Josie's knapsack can hold up to 14 lb of items. For $j = 1, 2, 3, 4$, define

$$x_j = \begin{cases} 1 \text{ if Josie takes item } j \text{ on the hike} \\ 0 \text{ otherwise} \end{cases}$$

Then Josie can maximize the total benefit by solving (9).

In the following example, we show how the Stockco formulation can be modified to handle additional constraints.

EXAMPLE 2 Capital Budgeting (Continued)

Modify the Stockco formulation to account for each of the following requirements:

1 Stockco can invest in at most two investments.

2 If Stockco invests in investment 2, they must also invest in investment 1.

3 If Stockco invests in investment 2, they cannot invest in investment 4.

Solution **1** Simply add the constraint

$$x_1 + x_2 + x_3 + x_4 \le 2 \tag{10}$$

to (9). Because any choice of three or four investments will have $x_1 + x_2 + x_3 + x_4 \ge 3$, (10) excludes from consideration all investment combinations involving three or more investments. Thus, (10) eliminates from consideration exactly those combinations of investments that do not satisfy the first requirement.

2 In terms of x_1 and x_2, this requirement states that if $x_2 = 1$, then x_1 must also equal 1. If we add the constraint

$$x_2 \le x_1 \quad \text{or} \quad x_2 - x_1 \le 0 \tag{11}$$

to (9), then we will have taken care of the second requirement. To show that (11) is equivalent to requirement 2, we consider two possibilities: either $x_2 = 1$ or $x_2 = 0$.

Case 1 $x_2 = 1$. If $x_2 = 1$, then the (11) implies that $x_1 \ge 1$. Because x_1 must equal 0 or 1, this implies that $x_1 = 1$, as required by 2.

Case 2 $x_2 = 0$. In this case, (11) reduces to $x_1 \geq 0$, which allows $x_1 = 0$ or $x_1 = 1$. In short, if $x_2 = 0$, (11) does not restrict the value of x_1. This is also consistent with requirement 2.

In summary, for any value of x_2, (11) is equivalent to requirement 2.

3 Simply add the constraint

$$x_2 + x_4 \leq 1 \tag{12}$$

to (9). We now show that for the two cases $x_2 = 1$ and $x_2 = 0$, (12) is equivalent to the third requirement.

Case 1 $x_2 = 1$. In this case, we are investing in investment 2, and requirement 3 implies that Stockco cannot invest in investment 4 (that is, x_4 must equal 0). Note that if $x_2 = 1$, then (12) does imply $1 + x_4 \leq 1$, or $x_4 \leq 0$. Thus, if $x_2 = 1$, then (12) is consistent with requirement 3.

Case 2 $x_2 = 0$. In this case, requirement 3 does not restrict the value of x_4. Note that if $x_2 = 0$, then (12) reduces to $x_4 \leq 1$, which also leaves x_4 free to equal 0 or 1.

Fixed-Charge Problems

Example 3 illustrates an important trick that can be used to formulate many location and production problems as IPs.

EXAMPLE 3 Fixed-Charge IP

Gandhi Cloth Company is capable of manufacturing three types of clothing: shirts, shorts, and pants. The manufacture of each type of clothing requires that Gandhi have the appropriate type of machinery available. The machinery needed to manufacture each type of clothing must be rented at the following rates: shirt machinery, $200 per week; shorts machinery, $150 per week; pants machinery, $100 per week. The manufacture of each type of clothing also requires the amounts of cloth and labor shown in Table 2. Each week, 150 hours of labor and 160 sq yd of cloth are available. The variable unit cost and selling price for each type of clothing are shown in Table 3. Formulate an IP whose solution will maximize Gandhi's weekly profits.

Solution As in LP formulations, we define a decision variable for each decision that Gandhi must make. Clearly, Gandhi must decide how many of each type of clothing should be manufactured each week, so we define

x_1 = number of shirts produced each week
x_2 = number of shorts produced each week
x_3 = number of pants produced each week

TABLE 2
Resource Requirements for Gandhi

Clothing Type	Labor (Hours)	Cloth (Square Yards)
Shirt	3	4
Shorts	2	3
Pants	6	4

TABLE 3
Revenue and Cost Information for Gandhi

Clothing Type	Sales Price ($)	Variable Cost ($)
Shirt	12	6
Shorts	8	4
Pants	15	8

Note that the cost of renting machinery depends only on the types of clothing produced, not on the amount of each type of clothing. This enables us to express the cost of renting machinery by using the following variables:

$$y_1 = \begin{cases} 1 & \text{if any shirts are manufactured} \\ 0 & \text{otherwise} \end{cases}$$

$$y_2 = \begin{cases} 1 & \text{if any shorts are manufactured} \\ 0 & \text{otherwise} \end{cases}$$

$$y_3 = \begin{cases} 1 & \text{if any pants are manufactured} \\ 0 & \text{otherwise} \end{cases}$$

In short, if $x_j > 0$, then $y_j = 1$, and if $x_j = 0$, then $y_j = 0$. Thus, Gandhi's weekly profits = (weekly sales revenue) − (weekly variable costs) − (weekly costs of renting machinery). Also,

$$\text{Weekly cost of renting machinery} = 200y_1 + 150y_2 + 100y_3 \qquad \textbf{(13)}$$

To justify (13), note that it picks up the rental costs only for the machines needed to manufacture those products that Gandhi is actually manufacturing. For example, suppose that shirts and pants are manufactured. Then $y_1 = y_3 = 1$ and $y_2 = 0$, and the total weekly rental cost will be $200 + 100 = \$300$.

Because the cost of renting, say, shirt machinery does not depend on the number of shirts produced, the cost of renting each type of machinery is called a **fixed charge.** A fixed charge for an activity is a cost that is assessed whenever the activity is undertaken at a nonzero level. The presence of fixed charges will make the formulation of the Gandhi problem much more difficult.

We can now express Gandhi's weekly profits as

$$\begin{aligned}\text{Weekly profit} &= (12x_1 + 8x_2 + 15x_3) - (6x_1 + 4x_2 + 8x_3) \\ &\quad - (200y_1 + 150y_2 + 100y_3) \\ &= 6x_1 + 4x_2 + 7x_3 - 200y_1 - 150y_2 - 100y_3\end{aligned}$$

Thus, Gandhi wants to maximize

$$z = 6x_1 + 4x_2 + 7x_3 - 200y_1 - 150y_2 - 100y_3$$

Because its supply of labor and cloth is limited, Gandhi faces the following two constraints:

Constraint 1 At most, 150 hours of labor can be used each week.

Constraint 2 At most, 160 sq yd of cloth can be used each week.

Constraint 1 is expressed by

$$3x_1 + 2x_2 + 6x_3 \le 150 \qquad \text{(Labor constraint)} \qquad \textbf{(14)}$$

Constraint 2 is expressed by

$$4x_1 + 3x_2 + 4x_3 \le 160 \qquad \text{(Cloth constraint)} \tag{15}$$

Observe that $x_j > 0$ and x_j integer ($j = 1, 2, 3$) must hold along with $y_j = 0$ or 1 ($j = 1, 2, 3$). Combining (14) and (15) with these restrictions and the objective function yields the following IP:

$$\begin{aligned} \max z &= 6x_1 + 4x_2 + 7x_3 - 200y_1 - 150y_2 - 100y_3 \\ \text{s.t.} \quad & 3x_1 + 2x_2 + 6x_3 \le 150 \\ & 4x_1 + 3x_2 + 4x_3 \le 160 \\ & x_1, x_2, x_3 \ge 0;\ x_1, x_2, x_3 \text{ integer} \\ & y_1, y_2, y_3 = 0 \text{ or } 1 \end{aligned} \tag{IP 1}$$

The optimal solution to this problem is found to be $x_1 = 30$, $x_3 = 10$, $x_2 = y_1 = y_2 = y_3 = 0$. This cannot be the optimal solution to Gandhi's problem because it indicates that Gandhi can manufacture shirts and pants without incurring the cost of renting the needed machinery. The current formulation is incorrect because the variables y_1, y_2, and y_3 are not present in the constraints. This means that there is nothing to stop us from setting $y_1 = y_2 = y_3 = 0$. Setting $y_i = 0$ is certainly less costly than setting $y_i = 1$, so a minimum-cost solution to (IP 1) will always set $y_i = 0$. Somehow we must modify (IP 1) so that whenever $x_i > 0$, $y_i = 1$ must hold. The following trick will accomplish this goal. Let M_1, M_2, and M_3 be three large positive numbers, and add the following constraints to (IP 1):

$$x_1 \le M_1 y_1 \tag{16}$$

$$x_2 \le M_2 y_2 \tag{17}$$

$$x_3 \le M_3 y_3 \tag{18}$$

Adding (16)–(18) to IP 1 will ensure that if $x_i > 0$, then $y_i = 1$. To illustrate, let us show that (16) ensures that if $x_1 > 0$, then $y_1 = 1$. If $x_1 > 0$, then y_1 cannot be 0. For if $y_1 = 0$, then (16) would imply $x_1 \le 0$ or $x_1 = 0$. Thus, if $x_1 > 0$, $y_1 = 1$ must hold. If any shirts are produced ($x_1 > 0$), (16) ensures that $y_1 = 1$, and the objective function will include the cost of the machinery needed to manufacture shirts. Note that if $y_1 = 1$, then (16) becomes $x_1 \le M_1$, which does not unnecessarily restrict the value of x_1. If M_1 were not chosen large, however (say, $M_1 = 10$), then (16) would unnecessarily restrict the value of x_1. In general, M_i should be set equal to the maximum value that x_i can attain. In the current problem, at most 40 shirts can be produced (if Gandhi produced more than 40 shirts, the company would run out of cloth), so we can safely choose $M_1 = 40$. The reader should verify that we can choose $M_2 = 53$ and $M_3 = 25$.

If $x_1 = 0$, (16) becomes $0 \le M_1 y_1$. This allows either $y_1 = 0$ or $y_1 = 1$. Because $y_1 = 0$ is less costly than $y_1 = 1$, the optimal solution will choose $y_1 = 0$ if $x_1 = 0$. In summary, we have shown that if (16)–(18) are added to (IP 1), then $x_i > 0$ will imply $y_i = 1$, and $x_i = 0$ will imply $y_i = 0$.

The optimal solution to the Gandhi problem is $z = \$75$, $x_3 = 25$, $y_3 = 1$. Thus, Gandhi should produce 25 pants each week.

The Gandhi problem is an example of a **fixed-charge problem.** In a fixed-charge problem, there is a cost associated with performing an activity at a nonzero level that does not depend on the level of the activity. Thus, in the Gandhi problem, if we make any shirts at all (no matter how many we make), we must pay the fixed charge of \$200 to rent a shirt machine. Problems in which a decision maker must choose where to locate facilities are often fixed-charge problems. The decision maker must choose where to locate various facilities (such as plants, warehouses, or business offices), and a fixed charge is often associated with building or operating a facility. Example 4 is a typical location problem involving the idea of a fixed charge.

EXAMPLE 4 The Lockbox Problem

J. C. Nickles receives credit card payments from four regions of the country (West, Midwest, East, and South). The average daily value of payments mailed by customers from each region is as follows: the West, \$70,000; the Midwest, \$50,000; the East, \$60,000; the South, \$40,000. Nickles must decide where customers should mail their payments. Because Nickles can earn 20% annual interest by investing these revenues, it would like to receive payments as quickly as possible. Nickles is considering setting up operations to process payments (often referred to as lockboxes) in four different cities: Los Angeles, Chicago, New York, and Atlanta. The average number of days (from time payment is sent) until a check clears and Nickles can deposit the money depends on the city to which the payment is mailed, as shown in Table 4. For example, if a check is mailed from the West to Atlanta, it would take an average of 8 days before Nickles could earn interest on the check. The annual cost of running a lockbox in any city is \$50,000. Formulate an IP that Nickles can use to minimize the sum of costs due to lost interest and lockbox operations. Assume that each region must send all its money to a single city and that there is no limit on the amount of money that each lockbox can handle.

Solution Nickles must make two types of decisions. First, Nickles must decide where to operate lockboxes. We define, for $j = 1, 2, 3, 4$,

$$y_j = \begin{cases} 1 & \text{if a lockbox is operated in city } j \\ 0 & \text{otherwise} \end{cases}$$

Thus, $y_2 = 1$ if a lockbox is operated in Chicago, and $y_3 = 0$ if no lockbox is operated in New York. Second, Nickles must determine where each region of the country should send payments. We define (for $i, j = 1, 2, 3, 4$)

$$x_{ij} = \begin{cases} 1 & \text{if region } i \text{ sends payments to city } j \\ 0 & \text{otherwise} \end{cases}$$

For example, $x_{12} = 1$ if the West sends payments to Chicago, and $x_{23} = 0$ if the Midwest does not send payments to New York.

Nickles wants to minimize (total annual cost) = (annual cost of operating lockboxes) + (annual lost interest cost). To determine how much interest Nickles loses annually, we must determine how much revenue would be lost if payments from region i were sent to region j. For example, how much in annual interest would Nickles lose if customers from the West region sent payments to New York? On any given day, 8 days' worth, or $8(70{,}000) = \$560{,}000$ of West payments will be in the mail and will not be earning in-

TABLE 4

Average Number of Days from Mailing of Payment Until Payment Clears

From	To: City 1 (Los Angeles)	City 2 (Chicago)	City 3 (New York)	City 4 (Atlanta)
Region 1 West	2	6	8	8
Region 2 Midwest	6	2	5	5
Region 3 East	8	5	2	5
Region 4 South	8	5	5	2

terest. Because Nickles can earn 20% annually, each year West funds will result in 0.20(560,000) = \$112,000 in lost interest. Similar calculations for the annual cost of lost interest for each possible assignment of a region to a city yield the results shown in Table 5. The lost interest cost from sending region i's payments to city j is only incurred if $x_{ij} = 1$, so Nickles's annual lost interest costs (in thousands) are

$$\begin{aligned}\text{Annual lost interest costs} = {} & 28x_{11} + 84x_{12} + 112x_{13} + 112x_{14} \\ & + 60x_{21} + 20x_{22} + 50x_{23} + 50x_{24} \\ & + 96x_{31} + 60x_{32} + 24x_{33} + 60x_{34} \\ & + 64x_{41} + 40x_{42} + 40x_{43} + 16x_{44}\end{aligned}$$

The cost of operating a lockbox in city i is incurred if and only if $y_i = 1$, so the annual lockbox operating costs (in thousands) are given by

$$\text{Total annual lockbox operating cost} = 50y_1 + 50y_2 + 50y_3 + 50y_4$$

Thus, Nickles's objective function may be written as

$$\begin{aligned}\min z = {} & 28x_{11} + 84x_{12} + 112x_{13} + 112x_{14} \\ & + 60x_{21} + 20x_{22} + 50x_{23} + 50x_{24} \\ & + 96x_{31} + 60x_{32} + 24x_{33} + 60x_{34} \\ & + 64x_{41} + 40x_{42} + 40x_{43} + 16x_{44} \\ & + 50y_1 + 50y_2 + 50y_3 + 50y_4\end{aligned} \tag{19}$$

Nickles faces two types of constraints.

Type 1 Constraint Each region must send its payments to a single city.

Type 2 Constraint If a region is assigned to send its payments to a city, that city must have a lockbox.

TABLE 5
Calculation of Annual Lost Interest

Assignment	Annual Lost Interest Cost (\$)
West to L.A.	0.20(70,000)2 = 28,000
West to Chicago	0.20(70,000)6 = 84,000
West to N.Y.	0.20(70,000)8 = 112,000
West to Atlanta	0.20(70,000)8 = 112,000
Midwest to L.A.	0.20(50,000)6 = 60,000
Midwest to Chicago	0.20(50,000)2 = 20,000
Midwest to N.Y.	0.20(50,000)5 = 50,000
Midwest to Atlanta	0.20(50,000)5 = 50,000
East to L.A.	0.20(60,000)8 = 96,000
East to Chicago	0.20(60,000)5 = 60,000
East to N.Y.	0.20(60,000)2 = 24,000
East to Atlanta	0.20(60,000)5 = 60,000
South to L.A.	0.20(40,000)8 = 64,000
South to Chicago	0.20(40,000)5 = 40,000
South to N.Y	0.20(40,000)5 = 40,000
South to Atlanta	0.20(40,000)2 = 16,000

The type 1 constraints state that for region i ($i = 1, 2, 3, 4$) exactly one of x_{i1}, x_{i2}, x_{i3}, and x_{i4} must equal 1 and the others must equal 0. This can be accomplished by including the following four constraints:

$$x_{11} + x_{12} + x_{13} + x_{14} = 1 \qquad \text{(West region constraint)} \tag{20}$$
$$x_{21} + x_{22} + x_{23} + x_{24} = 1 \qquad \text{(Midwest region constraint)} \tag{21}$$
$$x_{31} + x_{32} + x_{33} + x_{34} = 1 \qquad \text{(East region constraint)} \tag{22}$$
$$x_{41} + x_{42} + x_{43} + x_{44} = 1 \qquad \text{(South region constraint)} \tag{23}$$

The type 2 constraints state that if

$$x_{ij} = 1 \qquad \text{(that is, customers in region } i \text{ send payments to city } j\text{)} \tag{24}$$

then y_j must equal 1. For example, suppose $x_{12} = 1$. Then there must be a lockbox at city 2, so $y_2 = 1$ must hold. This can be ensured by adding 16 constraints of the form

$$x_{ij} \le y_j \qquad (i = 1, 2, 3, 4; j = 1, 2, 3, 4) \tag{25}$$

If $x_{ij} = 1$, then (25) ensures that $y_j = 1$, as desired. Also, if $x_{1j} = x_{2j} = x_{3j} = x_{4j} = 0$, then (25) allows $y_j = 0$ or $y_j = 1$. As in the fixed-charge example, the act of minimizing costs will result in $y_j = 0$. In summary, the constraints in (25) ensure that Nickles pays for a lockbox at city i if it uses a lockbox at city i.

Combining (19)–(23) with the 4(4) = 16 constraints in (25) and the 0–1 restrictions on the variables yields the following formulation:

$$\begin{aligned}
\min z = {} & 28x_{11} + 84x_{12} + 112x_{13} + 112x_{14} + 60x_{21} + 20x_{22} + 50x_{23} + 50x_{24} \\
& + 96x_{31} + 60x_{32} + 24x_{33} + 60x_{34} + 64x_{41} + 40x_{42} + 40x_{43} + 16x_{44} \\
& + 50y_1 + 50y_2 + 50y_3 + 50y_4 \\
\text{s.t.} \quad & x_{11} + x_{12} + x_{13} + x_{14} = 1 \qquad \text{(West region constraint)} \\
& x_{21} + x_{22} + x_{23} + x_{24} = 1 \qquad \text{(Midwest region constraint)} \\
& x_{31} + x_{32} + x_{33} + x_{34} = 1 \qquad \text{(East region constraint)} \\
& x_{41} + x_{42} + x_{43} + x_{44} = 1 \qquad \text{(South region constraint)} \\
& x_{11} \le y_1, x_{21} \le y_1, x_{31} \le y_1, x_{41} \le y_1, x_{12} \le y_2, x_{22} \le y_2, x_{32} \le y_2, x_{42} \le y_2, \\
& x_{13} \le y_3, x_{23} \le y_3, x_{33} \le y_3, x_{43} \le y_3, x_{14} \le y_4, x_{24} \le y_4, x_{34} \le y_4, x_{44} \le y_4 \\
& \text{All } x_{ij} \text{ and } y_j = 0 \text{ or } 1
\end{aligned}$$

The optimal solution is $z = 242$, $y_1 = 1$, $y_3 = 1$, $x_{11} = 1$, $x_{23} = 1$, $x_{33} = 1$, $x_{43} = 1$. Thus, Nickles should have a lockbox operation in Los Angeles and New York. West customers should send payments to Los Angeles, and all other customers should send payments to New York.

There is an alternative way of modeling the Type 2 constraints. Instead of the 16 constraints of the form $x_{ij} \le y_j$, we may include the following four constraints:

$$x_{11} + x_{21} + x_{31} + x_{41} \le 4y_1 \qquad \text{(Los Angeles constraint)}$$
$$x_{12} + x_{22} + x_{32} + x_{42} \le 4y_2 \qquad \text{(Chicago constraint)}$$
$$x_{13} + x_{23} + x_{33} + x_{43} \le 4y_3 \qquad \text{(New York constraint)}$$
$$x_{14} + x_{24} + x_{34} + x_{44} \le 4y_4 \qquad \text{(Atlanta constraint)}$$

For the given city, each constraint ensures that if the lockbox is used, then Nickles must pay for it. For example, consider $x_{14} + x_{24} + x_{34} + x_{44} \le 4y_4$. The lockbox in Atlanta is used if $x_{14} = 1$, $x_{24} = 1$, $x_{34} = 1$, or $x_{44} = 1$. If any of these variables equals 1, then the Atlanta constraint ensures that $y_4 = 1$, and Nickles must pay for the lockbox. If all these variables are 0, then the act of minimizing costs will cause $y_4 = 0$, and the cost of the Atlanta lockbox will not be incurred. Why does the right-hand side of each constraint equal 4? This ensures that for each city, it is possible to send money from all four regions to the city. In Section 9.3, we discuss which of the two alternative formulations of the lockbox problem is easier for a computer to solve. The answer may surprise you!

Set-Covering Problems

The following example is typical of an important class of IPs known as set-covering problems.

EXAMPLE 5 Facility-Location Set-Covering Problem

There are six cities (cities 1–6) in Kilroy County. The county must determine where to build fire stations. The county wants to build the minimum number of fire stations needed to ensure that at least one fire station is within 15 minutes (driving time) of each city. The times (in minutes) required to drive between the cities in Kilroy County are shown in Table 6. Formulate an IP that will tell Kilroy how many fire stations should be built and where they should be located.

Solution For each city, Kilroy must determine whether to build a fire station there. We define the 0–1 variables x_1, x_2, x_3, x_4, x_5, and x_6 by

$$x_i = \begin{cases} 1 & \text{if a fire station is built in city } i \\ 0 & \text{otherwise} \end{cases}$$

Then the total number of fire stations that are built is given by $x_1 + x_2 + x_3 + x_4 + x_5 + x_6$, and Kilroy's objective function is to minimize

$$z = x_1 + x_2 + x_3 + x_4 + x_5 + x_6$$

What are Kilroy's constraints? Kilroy must ensure that there is a fire station within 15 minutes of each city. Table 7 indicates which locations can reach the city in 15 minutes or less. To ensure that at least one fire station is within 15 minutes of city 1, we add the constraint

$$x_1 + x_2 \geq 1 \qquad \text{(City 1 constraint)}$$

This constraint ensures that $x_1 = x_2 = 0$ is impossible, so at least one fire station will be built within 15 minutes of city 1. Similarly the constraint

$$x_1 + x_2 + x_6 \geq 1 \qquad \text{(City 2 constraint)}$$

ensures that at least one fire station will be located within 15 minutes of city 2. In a similar fashion, we obtain constraints for cities 3–6. Combining these six constraints with the

TABLE 6
Time Required to Travel between Cities in Kilroy County

	To					
From	City 1	City 2	City 3	City 4	City 5	City 6
City 1	0	10	20	30	30	20
City 2	10	0	25	35	20	10
City 3	20	25	0	15	30	20
City 4	30	35	15	0	15	25
City 5	30	20	30	15	0	14
City 6	20	10	20	25	14	0

TABLE 7
Cities within 15 Minutes of Given City

City	Within 15 Minutes
1	1, 2
2	1, 2, 6
3	3, 4
4	3, 4, 5
5	4, 5, 6
6	2, 5, 6

objective function (and with the fact that each variable must equal 0 or 1), we obtain the following 0–1 IP:

$$
\begin{aligned}
\min z &= x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \\
\text{s.t.}\quad & x_1 + x_2 \geq 1 \quad \text{(City 1 constraint)} \\
& x_1 + x_2 + x_6 \geq 1 \quad \text{(City 2 constraint)} \\
& x_3 + x_4 \geq 1 \quad \text{(City 3 constraint)} \\
& x_3 + x_4 + x_5 \geq 1 \quad \text{(City 4 constraint)} \\
& x_4 + x_5 + x_6 \geq 1 \quad \text{(City 5 constraint)} \\
& x_2 + x_5 + x_6 \geq 1 \quad \text{(City 6 constraint)} \\
& x_i = 0 \text{ or } 1 \quad (i = 1, 2, 3, 4, 5, 6)
\end{aligned}
$$

One optimal solution to this IP is $z = 2$, $x_2 = x_4 = 1$, $x_1 = x_3 = x_5 = x_6 = 0$. Thus, Kilroy County can build two fire stations: one in city 2 and one in city 4.

As noted, Example 5 represents a class of IPs known as **set-covering problems.** In a set-covering problem, each member of a given set (call it set 1) must be "covered" by an acceptable member of some set (call it set 2). The objective in a set-covering problem is to minimize the number of elements in set 2 that are required to cover all the elements in set 1. In Example 5, set 1 is the cities in Kilroy County, and set 2 is the set of fire stations. The station in city 2 covers cities 1, 2, and 6, and the station in city 4 covers cities 3, 4, and 5. Set-covering problems have many applications in areas such as airline crew scheduling, political districting, airline scheduling, and truck routing.

Either–Or Constraints

The following situation commonly occurs in mathematical programming problems. We are given two constraints of the form

$$f(x_1, x_2, \ldots, x_n) \leq 0 \tag{26}$$

$$g(x_1, x_2, \ldots, x_n) \leq 0 \tag{27}$$

We want to ensure that at least one of (26) and (27) is satisfied, often called **either–or constraints.** Adding the two constraints (26′) and (27′) to the formulation will ensure that at least one of (26) and (27) is satisfied:

$$f(x_1, x_2, \ldots, x_n) \leq My \tag{26′}$$

$$g(x_1, x_2, \ldots, x_n) \leq M(1 - y) \tag{27′}$$

In (26′) and (27′), y is a 0–1 variable, and M is a number chosen large enough to ensure that $f(x_1, x_2, \ldots, x_n) \le M$ and $g(x_1, x_2, \ldots, x_n) \le M$ are satisfied for all values of $x_1, x_2, \ldots, x_n$ that satisfy the other constraints in the problem.

Let us show that the inclusion of constraints (26′) and (27′) is equivalent to at least one of (26) and (27) being satisfied. Either $y = 0$ or $y = 1$. If $y = 0$, then (26′) and (27′) become $f \le 0$ and $g \le M$. Thus, if $y = 0$, then (26) (and possibly (27)) must be satisfied. Similarly, if $y = 1$, then (26′) and (27′) become $f \le M$ and $g \le 0$. Thus, if $y = 1$, then (27) (and possibly (26)) must be satisfied. Therefore, whether $y = 0$ or $y = 1$, (26′) and (27′) ensure that at least one of (26) and (27) is satisfied.

The following example illustrates the use of either–or constraints.

EXAMPLE 6 Either–Or Constraint

Dorian Auto is considering manufacturing three types of autos: compact, midsize, and large. The resources required for, and the profits yielded by, each type of car are shown in Table 8. Currently, 6,000 tons of steel and 60,000 hours of labor are available. For production of a type of car to be economically feasible, at least 1,000 cars of that type must be produced. Formulate an IP to maximize Dorian's profit.

Solution Because Dorian must determine how many cars of each type should be built, we define

$$x_1 = \text{number of compact cars produced}$$
$$x_2 = \text{number of midsize cars produced}$$
$$x_3 = \text{number of large cars produced}$$

Then contribution to profit (in thousands of dollars) is $2x_1 + 3x_2 + 4x_3$, and Dorian's objective function is

$$\max z = 2x_1 + 3x_2 + 4x_3$$

We know that if any cars of a given type are produced, then at least 1,000 cars of that type must be produced. Thus, for $i = 1, 2, 3$, we must have $x_i \le 0$ or $x_i \ge 1{,}000$. Steel and labor are limited, so Dorian must satisfy the following five constraints:

Constraint 1 $x_1 \le 0$ or $x_1 \ge 1{,}000$.

Constraint 2 $x_2 \le 0$ or $x_2 \ge 1{,}000$.

Constraint 3 $x_3 \le 0$ or $x_3 \ge 1{,}000$.

Constraint 4 The cars produced can use at most 6,000 tons of steel.

Constraint 5 The cars produced can use at most 60,000 hours of labor.

TABLE 8
Resources and Profits for Three Types of Cars

	Car Type		
Resource	Compact	Midsize	Large
Steel required	1.5 tons	3 tons	5 tons
Labor required	30 hours	25 hours	40 hours
Profit yielded ($)	2,000	3,000	4,000

From our previous discussion, we see that if we define $f(x_1, x_2, x_3) = x_1$ and $g(x_1, x_2, x_3) = 1{,}000 - x_1$, we can replace Constraint 1 by the following pair of constraints:

$$
\begin{aligned}
x_1 &\le M_1 y_1 \\
1{,}000 - x_1 &\le M_1(1 - y_1) \\
y_1 &= 0 \text{ or } 1
\end{aligned}
$$

To ensure that both x_1 and $1{,}000 - x_1$ will never exceed M_1, it suffices to choose M_1 large enough so that M_1 exceeds 1,000 and x_1 is always less than M_1. Building $\frac{60{,}000}{30} = 2{,}000$ compacts would use all available labor (and still leave some steel), so at most 2,000 compacts can be built. Thus, we may choose $M_1 = 2{,}000$. Similarly, Constraint 2 may be replaced by the following pair of constraints:

$$
\begin{aligned}
x_2 &\le M_2 y_2 \\
1{,}000 - x_2 &\le M_2(1 - y_2) \\
y_2 &= 0 \text{ or } 1
\end{aligned}
$$

You should verify that $M_2 = 2{,}000$ is satisfactory. Similarly, Constraint 3 may be replaced by

$$
\begin{aligned}
x_3 &\le M_3 y_3 \\
1{,}000 - x_3 &\le M_3(1 - y_3) \\
y_3 &= 0 \text{ or } 1
\end{aligned}
$$

Again, you should verify that $M_3 = 1{,}200$ is satisfactory. Constraint 4 is a straightforward resource constraint that reduces to

$$1.5x_1 + 3x_2 + 5x_3 \le 6{,}000 \qquad \text{(Steel constraint)}$$

Constraint 5 is a straightforward resource usage constraint that reduces to

$$30x_1 + 25x_2 + 40x_3 \le 60{,}000 \qquad \text{(Labor constraint)}$$

After noting that $x_i \ge 0$ and that x_i must be an integer, we obtain the following IP:

$$
\begin{aligned}
\max z = 2x_1 &+ 3x_2 + 4x_3 \\
\text{s.t.} \quad x_1 &\le 2{,}000y_1 \\
1{,}000 - x_1 &\le 2{,}000(1 - y_1) \\
x_2 &\le 2{,}000y_2 \\
1{,}000 - x_2 &\le 2{,}000(1 - y_2) \\
x_3 &\le 1{,}200y_3 \\
1{,}000 - x_3 &\le 1{,}200(1 - y_3) \\
1.5x_1 + 3x_2 + 5x_3 &\le 6{,}000 \qquad \text{(Steel constraint)} \\
30x_1 + 25x_2 + 40x_3 &\le 60{,}000 \qquad \text{(Labor constraint)} \\
x_1, x_2, x_3 &\ge 0;\ x_1, x_2, x_3 \text{ integer} \\
y_1, y_2, y_3 &= 0 \text{ or } 1
\end{aligned}
$$

The optimal solution to the IP is $z = 6{,}000$, $x_2 = 2{,}000$, $y_2 = 1$, $y_1 = y_3 = x_1 = x_3 = 0$. Thus, Dorian should produce 2,000 midsize cars. If Dorian had not been required to manufacture at least 1,000 cars of each type, then the optimal solution would have been to produce 570 compacts and 1,715 midsize cars.

If–Then Constraints

In many applications, the following situation occurs: We want to ensure that if a constraint $f(x_1, x_2, \ldots, x_n) > 0$ is satisfied, then the constraint $g(x_1, x_2, \ldots, x_n) \geq 0$ must be satisfied, while if $f(x_1, x_2, \ldots, x_n) > 0$ is not satisfied, then $g(x_1, x_2, \ldots, x_n) \geq 0$ may or may not be satisfied. In short, we want to ensure that $f(x_1, x_2, \ldots, x_n) > 0$ implies $g(x_1, x_2, \ldots, x_n) \geq 0$.

To ensure this, we include the following constraints in the formulation:

$$-g(x_1, x_2, \ldots, x_n) \leq My \tag{28}$$

$$f(x_1, x_2, \ldots, x_n) \leq M(1 - y) \tag{29}$$

$$y = 0 \text{ or } 1$$

As usual, M is a large positive number. (M must be chosen large enough so that $f \leq M$ and $-g \leq M$ hold for all values of $x_1, x_2, \ldots, x_n$ that satisfy the other constraints in the problem.) Observe that if $f > 0$, then (29) can be satisfied only if $y = 0$. Then (28) implies $-g \leq 0$, or $g \geq 0$, which is the desired result. Thus, if $f > 0$, then (28) and (29) ensure that $g \geq 0$. Also, if $f > 0$ is not satisfied, then (29) allows $y = 0$ or $y = 1$. By choosing $y = 1$, (28) is automatically satisfied. Thus, if $f > 0$ is not satisfied, then the values of $x_1, x_2, \ldots, x_n$ are unrestricted and $g < 0$ or $g \geq 0$ are both possible.

To illustrate the use of this idea, suppose we add the following constraint to the Nickles lockbox problem: If customers in region 1 send their payments to city 1, then no other customers may send their payments to city 1. Mathematically, this restriction may be expressed by

$$\text{If } x_{11} = 1, \qquad \text{then} \qquad x_{21} = x_{31} = x_{41} = 0 \tag{30}$$

Because all x_{ij} must equal 0 or 1, (30) may be written as

$$\text{If } x_{11} > 0, \qquad \text{then} \qquad x_{21} + x_{31} + x_{41} \leq 0, \qquad \text{or} \qquad -x_{21} - x_{31} - x_{41} \geq 0 \tag{30$'$}$$

If we define $f = x_{11}$ and $g = -x_{21} - x_{31} - x_{41}$, we can use (28) and (29) to express (30′) [and therefore (30)] by the following two constraints:

$$x_{21} + x_{31} + x_{41} \leq My$$

$$x_{11} \leq M(1 - y)$$

$$y = 0 \text{ or } 1$$

Because $-g$ and f can never exceed 3, we can choose $M = 3$ and add the following constraints to the original lockbox formulation:

$$x_{21} + x_{31} + x_{41} \leq 3y$$

$$x_{11} \leq 3(1 - y)$$

$$y = 0 \text{ or } 1$$

Integer Programming and Piecewise Linear Functions†

The next example shows how 0–1 variables can be used to model optimization problems involving piecewise linear functions. A **piecewise linear function** consists of several straight-line segments. The piecewise linear function in Figure 2 is made of four straight-line segments. The points where the slope of the piecewise linear function changes (or the range of definition of the function ends) are called the **break points** of the function. Thus, 0, 10, 30, 40, and 50 are the break points of the function pictured in Figure 2.

†This section covers topics that may be omitted with no loss of continuity.

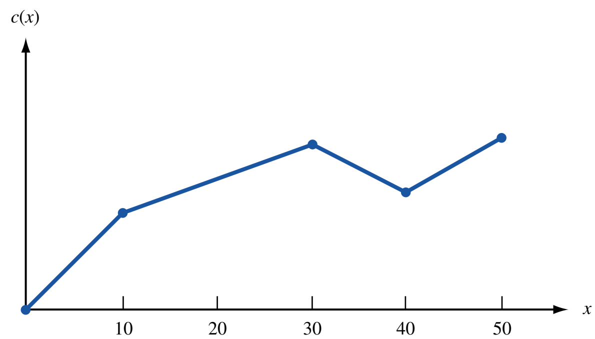


FIGURE 2
A Piecewise Linear Function

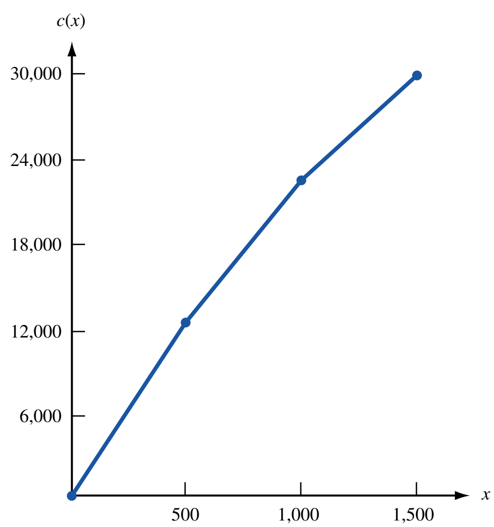


FIGURE 3
Cost of Purchasing Oil

To illustrate why piecewise linear functions can occur in applications, suppose we manufacture gasoline from oil. In purchasing oil from our supplier, we receive a quantity discount. The first 500 gallons of oil purchased cost 25¢ per gallon; the next 500 gallons cost 20¢ per gallon; and the next 500 gallons cost 15¢ per gallon. At most, 1,500 gallons of oil can be purchased. Let x be the number of gallons of oil purchased and $c(x)$ be the cost (in cents) of purchasing x gallons of oil. For $x \leq 0$, $c(x) = 0$. Then for $0 \leq x \leq 500$, $c(x) = 25x$. For $500 \leq x \leq 1{,}000$, $c(x) =$ (cost of purchasing first 500 gallons at 25¢ per gallon) + (cost of purchasing next $x - 500$ gallons at 20¢ per gallon) $= 25(500) + 20(x - 500) = 20x + 2{,}500$. For $1{,}000 \leq x \leq 1{,}500$, $c(x) =$ (cost of purchasing first 1,000 gallons) + (cost of purchasing next $x - 1{,}000$ gallons at 15¢ per gallon) $= c(1{,}000) + 15(x - 1{,}000) = 7{,}500 + 15x$. Thus, $c(x)$ has break points 0, 500, 1,000, and 1,500 and is graphed in Figure 3.

A piecewise linear function is not a linear function, so one might think that linear programming could not be used to solve optimization problems involving these functions. By using 0–1 variables, however, piecewise linear functions can be represented in linear form. Suppose that a piecewise linear function $f(x)$ has break points $b_1, b_2, \ldots, b_n$. For some k ($k = 1, 2, \ldots, n - 1$), $b_k \leq x \leq b_{k+1}$. Then, for some number z_k ($0 \leq z_k \leq 1$), x may be written as

$$x = z_k b_k + (1 - z_k) b_{k+1}$$

Because $f(x)$ is linear for $b_k \leq x \leq b_{k+1}$, we may write

$$f(x) = z_k f(b_k) + (1 - z_k) f(b_{k+1})$$

To illustrate the idea, take $x = 800$ in our oil example. Then we have $b_2 = 500 \le 800 \le 1{,}000 = b_3$, and we may write

$$\begin{aligned} x &= \tfrac{2}{5}(500) + \tfrac{3}{5}(1{,}000) \\ f(x) &= f(800) = \tfrac{2}{5}f(500) + \tfrac{3}{5}f(1{,}000) \\ &= \tfrac{2}{5}(12{,}500) + \tfrac{3}{5}(22{,}500) = 18{,}500 \end{aligned}$$

We are now ready to describe the method used to express a piecewise linear function via linear constraints and 0–1 variables:

Step 1 Wherever $f(x)$ occurs in the optimization problem, replace $f(x)$ by $z_1 f(b_1) + z_2 f(b_2) + \cdots + z_n f(b_n)$.

Step 2 Add the following constraints to the problem:

$$z_1 \le y_1,\ z_2 \le y_1 + y_2,\ z_3 \le y_2 + y_3, \ldots, z_{n-1} \le y_{n-2} + y_{n-1},\ z_n \le y_{n-1}$$
$$y_1 + y_2 + \cdots + y_{n-1} = 1$$
$$z_1 + z_2 + \cdots + z_n = 1$$
$$x = z_1 b_1 + z_2 b_2 + \cdots + z_n b_n$$
$$y_i = 0 \text{ or } 1 \quad (i = 1, 2, \ldots, n-1); \qquad z_i \ge 0 \quad (i = 1, 2, \ldots, n)$$

EXAMPLE 7 IP with Piecewise Linear Functions

Euing Gas produces two types of gasoline (gas 1 and gas 2) from two types of oil (oil 1 and oil 2). Each gallon of gas 1 must contain at least 50 percent oil 1, and each gallon of gas 2 must contain at least 60 percent oil 1. Each gallon of gas 1 can be sold for 12¢, and each gallon of gas 2 can be sold for 14¢. Currently, 500 gallons of oil 1 and 1,000 gallons of oil 2 are available. As many as 1,500 more gallons of oil 1 can be purchased at the following prices: first 500 gallons, 25¢ per gallon; next 500 gallons, 20¢ per gallon; next 500 gallons, 15¢ per gallon. Formulate an IP that will maximize Euing's profits (revenues − purchasing costs).

Solution Except for the fact that the cost of purchasing additional oil 1 is a piecewise linear function, this is a straightforward blending problem. With this in mind, we define

$$\begin{aligned} x &= \text{amount of oil 1 purchased} \\ x_{ij} &= \text{amount of oil } i \text{ used to produce gas } j \qquad (i, j = 1, 2) \end{aligned}$$

Then (in cents)

$$\text{Total revenue} - \text{cost of purchasing oil 1} = 12(x_{11} + x_{21}) + 14(x_{12} + x_{22}) - c(x)$$

As we have seen previously,

$$c(x) = \begin{cases} 25x & (0 \le x \le 500) \\ 20x + 2{,}500 & (500 \le x \le 1{,}000) \\ 15x + 7{,}500 & (1{,}000 \le x \le 1{,}500) \end{cases}$$

Thus, Euing's objective function is to maximize

$$z = 12x_{11} + 12x_{21} + 14x_{12} + 14x_{22} - c(x)$$

Euing faces the following constraints:

Constraint 1 Euing can use at most $x + 500$ gallons of oil 1.

Constraint 2 Euing can use at most 1,000 gallons of oil 2.

Constraint 3 The oil mixed to make gas 1 must be at least 50% oil 1.

Constraint 4 The oil mixed to make gas 2 must be at least 60% oil 1.

Constraint 1 yields

$$x_{11} + x_{12} \leq x + 500$$

Constraint 2 yields

$$x_{21} + x_{22} \leq 1{,}000$$

Constraint 3 yields

$$\frac{x_{11}}{x_{11} + x_{21}} \geq 0.5 \quad \text{or} \quad 0.5x_{11} - 0.5x_{21} \geq 0$$

Constraint 4 yields

$$\frac{x_{12}}{x_{12} + x_{22}} \geq 0.6 \quad \text{or} \quad 0.4x_{12} - 0.6x_{22} \geq 0$$

Also all variables must be nonnegative. Thus, Euing Gas must solve the following optimization problem:

$$\begin{aligned} \max z = 12x_{11} + 12x_{21} + 14x_{12} + 14x_{22} - c(x) \\ \text{s.t.} \quad x_{11} + x_{12} &\leq x + 500 \\ x_{21} + x_{22} &\leq 1{,}000 \\ 0.5x_{11} - 0.5x_{21} &\geq 0 \\ 0.4x_{12} - 0.6x_{22} &\geq 0 \\ x_{ij} \geq 0, 0 \leq x \leq 1{,}500 \end{aligned}$$

Because $c(x)$ is a piecewise linear function, the objective function is not a linear function of x, and this optimization is not an LP. By using the method described earlier, however, we can transform this problem into an IP. After recalling that the break points for $c(x)$ are 0, 500, 1,000, and 1,500, we proceed as follows:

Step 1 Replace $c(x)$ by $c(x) = z_1c(0) + z_2c(500) + z_3c(1{,}000) + z_4c(1{,}500)$.

Step 2 Add the following constraints:

$$\begin{aligned} x &= 0z_1 + 500z_2 + 1{,}000z_3 + 1{,}500z_4 \\ z_1 &\leq y_1, z_2 \leq y_1 + y_2, z_3 \leq y_2 + y_3, z_4 \leq y_3 \\ z_1 &+ z_2 + z_3 + z_4 = 1, \quad y_1 + y_2 + y_3 = 1 \\ y_i &= 0 \text{ or } 1\ (i = 1, 2, 3);\ z_i \geq 0\ (i = 1, 2, 3, 4) \end{aligned}$$

Our new formulation is the following IP:

$$\begin{aligned} \max z = 12x_{11} + 12x_{21} + 14x_{12} + 14x_{22} - z_1c(0) - z_2c(500) \\ - z_3c(1{,}000) - z_4c(1{,}500) \\ \text{s.t.} \quad x_{11} + x_{12} &\leq x + 500 \\ x_{21} + x_{22} &\leq 1{,}000 \\ 0.5x_{11} - 0.5x_{21} &\geq 0 \\ 0.4x_{12} - 0.6x_{22} &\geq 0 \end{aligned}$$

$$x = 0z_1 + 500z_2 + 1{,}000z_3 + 1{,}500z_4 \tag{31}$$

$$z_1 \leq y_1 \tag{32}$$

$$z_2 \leq y_1 + y_2 \tag{33}$$

$$z_3 \leq y_2 + y_3 \tag{34}$$

$$z_4 \le y_3 \qquad (35)$$
$$y_1 + y_2 + y_3 = 1 \qquad (36)$$
$$z_1 + z_2 + z_3 + z_4 = 1 \qquad (37)$$
$$y_i = 0 \text{ or } 1 \qquad (i = 1, 2, 3);\ z_i \ge 0 \qquad (i = 1, 2, 3, 4)$$
$$x_{ij} \ge 0$$

To see why this formulation works, observe that because $y_1 + y_2 + y_3 = 1$ and $y_i = 0$ or 1, exactly one of the y_i's will equal 1, and the others will equal 0. Now, (32)–(37) imply that if $y_i = 1$, then z_i and z_{i+1} may be positive, but all the other z_i's must equal 0. For instance, if $y_2 = 1$, then $y_1 = y_3 = 0$. Then (32)–(35) become $z_1 \le 0$, $z_2 \le 1$, $z_3 \le 1$, and $z_4 \le 0$. These constraints force $z_1 = z_4 = 0$ and allow z_2 and z_3 to be any nonnegative number less than or equal to 1. We can now show that (31)–(37) correctly represent the piecewise linear function $c(x)$. Choose any value of x, say $x = 800$. Note that $b_2 = 500 \le 800 \le 1{,}000 = b_3$. For $x = 800$, what values do our constraints assign to y_1, y_2, and y_3? The value $y_1 = 1$ is impossible, because if $y_1 = 1$, then $y_2 = y_3 = 0$. Then (34)–(35) force $z_3 = z_4 = 0$. Then (31) reduces to $800 = x = 500z_2$, which cannot be satisfied by $z_2 \le 1$. Similarly, $y_3 = 1$ is impossible. If we try $y_2 = 1$ (32) and (35) force $z_1 = z_4 = 0$. Then (33) and (34) imply $z_2 \le 1$ and $z_3 \le 1$. Now (31) becomes $800 = x = 500z_2 + 1{,}000z_3$. Because $z_2 + z_3 = 1$, we obtain $z_2 = \frac{2}{5}$ and $z_3 = \frac{3}{5}$. Now the objective function reduces to

$$12x_{11} + 12x_{21} + 14x_{21} + 14x_{22} - \frac{2c(500)}{5} - \frac{3c(1{,}000)}{5}$$

Because

$$c(800) = \frac{2c(500)}{5} + \frac{3c(1{,}000)}{5}$$

our objective function yields the correct value of Euing's profits!

The optimal solution to Euing's problem is $z = 12{,}500$, $x = 1{,}000$, $x_{12} = 1{,}500$, $x_{22} = 1{,}000$, $y_3 = z_3 = 1$. Thus, Euing should purchase 1,000 gallons of oil 1 and produce 2,500 gallons of gas 2.

In general, constraints of the form (31)–(37) ensure that if $b_i \le x \le b_{i+1}$, then $y_i = 1$ and only z_i and z_{i+1} can be positive. Because $c(x)$ is linear for $b_i \le x \le b_{i+1}$, the objective function will assign the correct value to $c(x)$.

If a piecewise linear function $f(x)$ involved in a formulation has the property that the slope of $f(x)$ becomes less favorable to the decision maker as x increases, then the tedious IP formulation we have just described is unnecessary.

EXAMPLE 8 Media Selection with Piecewise Linear Functions

Dorian Auto has a \$20,000 advertising budget. Dorian can purchase full-page ads in two magazines: *Inside Jocks* (IJ) and *Family Square* (FS). An exposure occurs when a person reads a Dorian Auto ad for the first time. The number of exposures generated by each ad in IJ is as follows: ads 1–6, 10,000 exposures; ads 7–10, 3,000 exposures; ads 11–15, 2,500 exposures; ads 16+, 0 exposures. For example, 8 ads in IJ would generate 6(10,000) + 2(3,000) = 66,000 exposures. The number of exposures generated by each ad in FS is as follows: ads 1–4, 8,000 exposures; ads 5–12, 6,000 exposures; ads 13–15, 2,000 exposures; ads 16+, 0 exposures. Thus, 13 ads in FS would generate 4(8,000) +

8(6,000) + 1(2,000) = 82,000 exposures. Each full-page ad in either magazine costs \$1,000. Assume there is no overlap in the readership of the two magazines. Formulate an IP to maximize the number of exposures that Dorian can obtain with limited advertising funds.

Solution If we define

$$\begin{aligned}
x_1 &= \text{number of IJ ads yielding 10,000 exposures}\\
x_2 &= \text{number of IJ ads yielding 3,000 exposures}\\
x_3 &= \text{number of IJ ads yielding 2,500 exposures}\\
y_1 &= \text{number of FS ads yielding 8,000 exposures}\\
y_2 &= \text{number of FS ads yielding 6,000 exposures}\\
y_3 &= \text{number of FS ads yielding 2,000 exposures}
\end{aligned}$$

then the total number of exposures (in thousands) is given by

$$10x_1 + 3x_2 + 2.5x_3 + 8y_1 + 6y_2 + 2y_3$$

Thus, Dorian wants to maximize

$$z = 10x_1 + 3x_2 + 2.5x_3 + 8y_1 + 6y_2 + 2y_3$$

Because the total amount spent (in thousands) is just the toal number of ads placed in both magazines, Dorian's budget constraint may be written as

$$x_1 + x_2 + x_3 + y_1 + y_2 + y_3 \le 20$$

The statement of the problem implies that $x_1 \le 6$, $x_2 \le 4$, $x_3 \le 5$, $y_1 \le 4$, $y_2 \le 8$, and $y_3 \le 3$ all must hold. Adding the sign restrictions on each variable and noting that each variable must be an integer, we obtain the following IP:

$$\begin{aligned}
\max z = 10x_1 &+ 3x_2 + 2.5x_3 + 8y_1 + 6y_2 + 2y_3\\
\text{s.t.} \quad x_1 &+ x_2 + x_3 + y_1 + y_2 + y_3 \le 20\\
x_1 &\le 6\\
x_2 &\le 4\\
x_3 &\le 5\\
y_1 &\le 4\\
y_2 &\le 8\\
y_3 &\le 3\\
x_i, y_i &\text{ integer} \quad (i = 1, 2, 3)\\
x_i, y_i &\ge 0 \quad (i = 1, 2, 3)
\end{aligned}$$

Observe that the statement of the problem implies that x_2 cannot be positive unless x_1 assumes its maximum value of 6. Similarly, x_3 cannot be positive unless x_2 assumes its maximum value of 4. Because x_1 ads generate more exposures than x_2 ads, however, the act of maximizing ensures that x_2 will be positive only if x_1 has been made as large as possible. Similarly, because x_3 ads generate fewer exposures than x_2 ads, x_3 will be positive only if x_2 assumes its maximum value. (Also, y_2 will be positive only if $y_1 = 4$, and y_3 will be positive only if $y_2 = 8$.)

The optimal solution to Dorian's IP is $z = 146{,}000$, $x_1 = 6$, $x_2 = 2$, $y_1 = 4$, $y_2 = 8$, $x_3 = 0$, $y_3 = 0$. Thus, Dorian will place $x_1 + x_2 = 8$ ads in IJ and $y_1 + y_2 = 12$ ads in FS.

In Example 8, additional advertising in a magazine yielded diminishing returns. This ensured that x_i (y_i) would be positive only if x_{i-1} (y_{i-1}) assumed its maximum value. If additional advertising generated increasing returns, then this formulation would not yield the correct solution. For example, suppose that the number of exposures generated by each IJ ad was as follows: ads 1–6, 2,500 exposures; ads 7–10, 3,000 exposures; ads 11–15, 10,000 exposures. Suppose also that the number of exposures generated by each FS is as follows: ads 1–4, 2,000 exposures; ads 5–12, 6,000 exposures; ads 13–15, 8,000 exposures.

If we define

$$
\begin{aligned}
x_1 &= \text{number of IJ ads generating 2,500 exposures} \\
x_2 &= \text{number of IJ ads generating 3,000 exposures} \\
x_3 &= \text{number of IJ ads generating 10,000 exposures} \\
y_1 &= \text{number of FS ads generating 2,000 exposures} \\
y_2 &= \text{number of FS ads generating 6,000 exposures} \\
y_3 &= \text{number of FS ads generating 8,000 exposures}
\end{aligned}
$$

the reasoning used in the previous example would lead to the following formulation:

$$
\begin{aligned}
\max z = {} & 2.5x_1 + 3x_2 + 10x_3 + 2y_1 + 6y_2 + 8y_3 \\
\text{s.t.} \quad & x_1 + x_2 + x_3 + y_1 + y_2 + y_3 \le 20 \\
& x_1 \le 6 \\
& x_2 \le 4 \\
& x_3 \le 5 \\
& y_1 \le 4 \\
& y_2 \le 8 \\
& y_3 \le 3 \\
& x_i, y_i \text{ integer} \quad (i = 1, 2, 3) \\
& x_i, y_i \le 0 \quad (i = 1, 2, 3)
\end{aligned}
$$

The optimal solution to this IP is $x_3 = 5$, $y_3 = 3$, $y_2 = 8$, $x_2 = 4$, $x_1 = 0$, $y_1 = 0$, which cannot be correct. According to this solution, $x_1 + x_2 + x_3 = 9$ ads should be placed in IJ. If 9 ads were placed in IJ, however, then it must be that $x_1 = 6$ and $x_2 = 3$. Therefore, we see that the type of formulation used in the Dorian Auto example is correct only if the piecewise linear objective function has a less favorable slope for larger values of x. In our second example, the effectiveness of an ad increased as the number of ads in a magazine increased, and the act of maximizing will not ensure that x_i can be positive only if x_{i-1} assumes its maximum value. In this case, the approach used in the Euing Gas example would yield a correct formulation (see Problem 8).

Solving IPs with LINDO

LINDO can be used to solve pure or mixed IPs. In addition to the optimal solution, the LINDO output for an IP gives shadow prices and reduced costs. Unfortunately, the shadow prices and reduced costs refer to subproblems generated during the branch-and-bound solution—*not* to the IP. Unlike linear programming, there is no well-developed theory of sensitivity analysis for integer programming. The reader interested in a discussion of sensitivity analysis for IPs should consult Williams (1985).

To use LINDO to solve an IP, begin by entering the problem as if it were an LP. After typing in the **END** statement (to designate the end of the LP constraints), type for each 0–1 variable x the following statement:

INTE x

Thus, for an IP in which x and y are 0–1 variables, the following statements would be typed after the **END** statement:

INTE x

INTE y

A variable (say, w) that can assume any non-negative integer value is indicated by the **GIN** statement. Thus, if w may assume the values 0, 1, 2, . . . , we would type the following statement after the **END** statement:

GIN w

To tell LINDO that the first n variables appearing in the formulation must be 0–1 variables, use the command **INT** n.

To tell LINDO that the first n variables appearing in the formulation may assume any non-negative integer value, use the command **GIN** n.

Gandhi

To illustrate how to use LINDO to solve IPs, we show how to solve Example 3 with LINDO. We typed the following input (file Gandhi):

```
MAX        6 X1 + 4 X2 + 7 X3 - 200 Y1 - 150 Y2 - 100 Y3
SUBJECT TO
        2)    3 X1 + 2 X2 + 6 X3 <= 150
        3)    4 X1 + 3 X2 + 4 X3 <= 160
        4)    X1 - 40 Y1 <= 0
        5)    X2 - 53 Y2 <= 0
        6)    X3 - 25 Y3 <= 0
END
GIN         X1
GIN         X2
GIN         X3
INTE        Y1
INTE        Y2
INTE        Y3
```

Thus we see that X1, X2, and X3 can be any nonnegative integer, while Y1, Y2, and Y3 must equal 0 or 1. By the way, we could have typed GIN 3 to ensure that X1, X2, and X3 must be nonnegative integers. The optimal solution found by LINDO is given in Figure 4.

Solving IPs with LINGO

Lock.lng

LINGO can also be used to solve IPs. To indicate that a variable must equal 0 or 1 use the **@BIN** operator (see the following example). To indicate that a variable must equal a non-negative integer, use the **@GIN** operator. We illustrate how LINGO is used to solve IPs with Example 4 (the Lockbox Problem). The following LINGO program (file Lock.lng) can be used to solve Example 4 (or any reasonably sized lockbox program).

```
MODEL:
  SETS:
  REGIONS/W,MW,E,S/:DEMAND;
  CITIES/LA,CHIC,NY,ATL/:Y;
  LINKS(REGIONS,CITIES):DAYS,COST,ASSIGN;
  ENDSETS
  MIN=@SUM(CITIES:50000*Y)+@SUM(LINKS:COST*ASSIGN);
  @FOR(LINKS(I,J):ASSIGN(I,J)  < Y(J));
  @FOR(REGIONS(I):
  @SUM(CITIES(J):ASSIGN(I,J))=1);
  @FOR(CITIES(I):@BIN(Y(I)););
```

```
MAX     6 X1 + 4 X2 + 7 X3 - 200 Y1 - 150 Y2 - 100 Y3
SUBJECT TO
       2)   3 X1 + 2 X2 + 6 X3 <=   150
       3)   4 X1 + 3 X2 + 4 X3 <=   160
       4)   X1 - 40 Y1 <=   0
       5)   X2 - 53 Y2 <=   0
       6)   X3 - 25 Y3 <=   0
END
GIN        X1
GIN        X2
GIN        X3
INTE       Y1
INTE       Y2
INTE       Y3


      OBJECTIVE FUNCTION VALUE

      1)     75.000000

VARIABLE        VALUE          REDUCED COST
      X1          .000000         -6.000000
      X2          .000000         -4.000000
      X3        25.000000         -7.000000
      Y1          .000000        200.000000
      Y2          .000000        150.000000
      Y3         1.000000        100.000000

     ROW   SLACK OR SURPLUS     DUAL PRICES
      2)          .000000          .000000
      3)        60.000000          .000000
      4)          .000000          .000000
      5)          .000000          .000000
      6)          .000000          .000000

NO. ITERATIONS=      11
BRANCHES=     1 DETERM.=  1.000E    0
```

FIGURE 4

```
  11]@FOR(LINKS(I,J):@BIN(ASSIGN(I,J)););
  12]@FOR(LINKS(I,J):COST(I,J)=.20*DEMAND(I)*DAYS(I,J));
  13]DATA:
  14]DAYS=2,6,8,8,
  15]6,2,5,5,
  16]8,5,2,5,
  17]8,5,5,2;
  18]DEMAND=70000,50000,60000,40000;
  19]ENDDATA
END
```

In line 2, we define the four regions of the country and associate a daily demand for cash payments from each region. Line 3 specifies the four cities where a lockbox may be built. With each city I, we associate a 0–1 variable (Y(I)) that equals 1 if a lockbox is built in the city or 0 otherwise. In line 4, we create a “link” (LINK(I,J)) between each region of the country and each potential lockbox site. Associated with each link are the following quantities:

1 The average number of days (DAYS) it takes a check to clear when mailed from region I to city J. This information is given in the DATA section.

2 The annual lost interest cost for funds sent from region *i* (COST) incurred if region I sends its money to city J.

3 A 0–1 variable ASSIGN(I,J) which equals 1 if region I sends its money to city J and 0 otherwise.

In line 6, we compute the total cost by summing 50000*Y(I) over all cities. This computes the total annual cost of running lockboxes. Then we sum COST*ASSIGN over all links. This picks up the total annual lost interest cost. The line 7 constraints ensure that (for all combinations of I and J) if region I sends its money to city J, then Y(J) = 1. This forces us to pay for lockboxes we use. Lines 8–9 ensure that each region of the country sends its money to some city. Line 10 ensures that each Y(I) equals 0 or 1. Line 11 ensures that each ASSIGN(I,J) equals 0 or 1 (actually we do not need this statement; see Problem 44). We compute the lost annual interest cost if region I sends its money to city J in line 12. This duplicates the calculations in Table 5. Note that an * is needed to ensure that multiplications are performed.

In lines 14–17, we input the average number of days required for a check to clear when it is sent from region I to city J. In line 18, we input the daily demand for each region.

Note that to obtain the objective function and constraints we selected the Model window and then chose LINDO, Generate, Display Model. See Figure 8.

Using the Excel Solver to Solve IP Problems

Gandhi.xls

It is easy to use the Excel Solver to solve integer programming problems. The file Gandhi.xls contains a spreadsheet solution to Example 3. See Figure 7 for the optimal solution. In our spreadsheet, the changing cells J4:J6 (the number of each product produced) must be integers. To tell the Solver that these changing cells must be integers, just select Add Constraint and point to the cells J4:J6. Then select int from the drop-down arrow in the middle.

The changing cells K4:K6 are the binary fixed charge variables. To tell the Solver that these changing cells must be binary, select Add Constraint and point to cells K4:K6. Then select bin from the drop-down arrow. See Figure 6.

From Figure 7, we find that the optimal solution (as found with LINDO) is to make 25 pairs of pants.

FIGURE 5

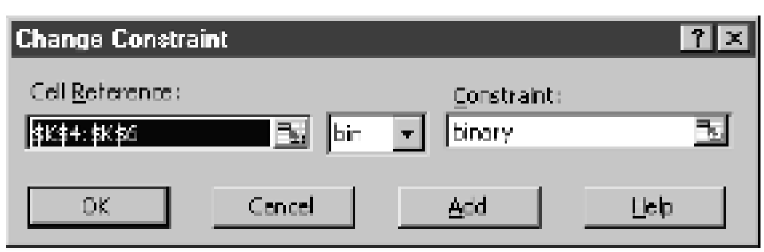


FIGURE 6

FIGURE 7

	A	B	C	D	E	F	G	H	I	J	K
1	**Gandhi**										
2											
3				Labor hours used	Cloth yards used	Unit price	Unit cost	Unit profit	Fixed Cost	Number Made	Binary variable
4			Shirt	3	4	$ 12.00	$ 6.00	$ 6.00	$ 200.00	0	0
5			Shorts	2	3	$ 8.00	$ 4.00	$ 4.00	$ 150.00	0	0
6			Pants	6	4	$ 15.00	$ 8.00	$ 7.00	$ 100.00	25	1
7		**Resource Constraints**									
8			Used		Available				Fixed charge	$ 100.00	
9		Labor	150	<=	150				Variable cost	$ 200.00	
10		Cloth	100	<=	160				Revenue	$ 375.00	
11									Profit	$ 75.00	
12		**Fixed Charge Constraints**	Number Made		Logical Upper Bound	Max possible to make					
13		Shirts	0	<=	0	40					
14		Shorts	0	<=	0	53.33333					
15		Pants	25	<=	25	25					

```
MIN     50000 Y(ATL + 50000 Y(NY + 50000 Y(CHIC + 50000 Y(LA + 16000 ASSIGNSA
     + 40000 ASSIGNSN + 40000 ASSIGNSC + 64000 ASSIGNSL + 60000 ASSIGNEA
     + 24000 ASSIGNEN + 60000 ASSIGNEC + 96000 ASSIGNEL + 50000 ASSIGNMW
     + 50000 ASSIGNMW + 20000 ASSIGNMW + 60000 ASSIGNMW + 112000 ASSIGNWA
     + 112000 ASSIGNWN + 84000 ASSIGNWC + 28000 ASSIGNWL
SUBJECT TO
2)- Y(LA + ASSIGNWL <=   0
3)- Y(CHIC + ASSIGNWC <=   0
4)- Y(NY + ASSIGNWN <=   0
5)- Y(ATL + ASSIGNWA <=   0
6)- Y(LA + ASSIGNMW <=   0
7)- Y(CHIC + ASSIGNMW <=   0
8)- Y(NY + ASSIGNMW <=   0
9)- Y(ATL + ASSIGNMW <=   0
10)- Y(LA + ASSIGNEL <=   0
11)- Y(CHIC + ASSIGNEC <=   0
12)- Y(NY + ASSIGNEN <=   0
13)- Y(ATL + ASSIGNEA <=   0
14)- Y(LA + ASSIGNSL <=   0
15)- Y(CHIC + ASSIGNSC <=   0
16)- Y(NY + ASSIGNSN <=   0
17)- Y(ATL + ASSIGNSA <=   0
18)  ASSIGNWA + ASSIGNWN + ASSIGNWC + ASSIGNWL =    1
19)  ASSIGNMW + ASSIGNMW + ASSIGNMW + ASSIGNMW =    1
20)  ASSIGNEA + ASSIGNEN + ASSIGNEC + ASSIGNEL =    1
21)  ASSIGNSA + ASSIGNSN + ASSIGNSC + ASSIGNSL =    1
END
INTE    20


[ERROR CODE: 96]
WARNING: SEVERAL LINGO NAMES MAY HAVE BEEN TRANSFORMED INTO A
SINGLE LINDO NAME.

LP OPTIMUM FOUND AT STEP      14
OBJECTIVE VALUE =    242000.000
ENUMERATION COMPLETE. BRANCHES=     0 PIVOTS=     14
```

FIGURE 8

```
LAST INTEGER SOLUTION IS THE BEST FOUND
RE-INSTALLING BEST SOLUTION...
```

VARIABLE	VALUE	REDUCED COST
DEMAND(W)	70000.00	0.0000000E+00
DEMAND(MW)	50000.00	0.0000000E+00
DEMAND(E)	60000.00	0.0000000E+00
DEMAND(S)	40000.00	0.0000000E+00
Y(LA)	1.000000	50000.00
Y(CHIC)	0.0000000E+00	50000.00
Y(NY)	1.000000	50000.00
Y(ATL)	0.0000000E+00	50000.00
DAYS(W, LA)	2.000000	0.0000000E+00
DAYS(W, CHIC)	6.000000	0.0000000E+00
DAYS(W, NY)	8.000000	0.0000000E+00
DAYS(W, ATL)	8.000000	0.0000000E+00
DAYS(MW, LA)	6.000000	0.0000000E+00
DAYS(MW, CHIC)	2.000000	0.0000000E+00
DAYS(MW, NY)	5.000000	0.0000000E+00
DAYS(MW, ATL)	5.000000	0.0000000E+00
DAYS(E, LA)	8.000000	0.0000000E+00
DAYS(E, CHIC)	5.000000	0.0000000E+00
DAYS(E, NY)	2.000000	0.0000000E+00
DAYS(E, ATL)	5.000000	0.0000000E+00
DAYS(S, LA)	8.000000	0.0000000E+00
DAYS(S, CHIC)	5.000000	0.0000000E+00
DAYS(S, NY)	5.000000	0.0000000E+00
DAYS(S, ATL)	2.000000	0.0000000E+00
COST(W, LA)	28000.00	0.0000000E+00
COST(W, CHIC)	84000.00	0.0000000E+00
COST(W, NY)	112000.0	0.0000000E+00
COST(W, ATL)	112000.0	0.0000000E+00
COST(MW, LA)	60000.00	0.0000000E+00
COST(MW, CHIC)	20000.00	0.0000000E+00
COST(MW, NY)	50000.00	0.0000000E+00
COST(MW, ATL)	50000.00	0.0000000E+00
COST(E, LA)	96000.00	0.0000000E+00
COST(E, CHIC)	60000.00	0.0000000E+00
COST(E, NY)	24000.00	0.0000000E+00
COST(E, ATL)	60000.00	0.0000000E+00
COST(S, LA)	64000.00	0.0000000E+00
COST(S, CHIC)	40000.00	0.0000000E+00
COST(S, NY)	40000.00	0.0000000E+00
COST(S, ATL)	16000.00	0.0000000E+00
ASSIGN(W, LA)	1.000000	28000.00
ASSIGN(W, CHIC)	0.0000000E+00	84000.00
ASSIGN(W, NY)	0.0000000E+00	112000.0
ASSIGN(W, ATL)	0.0000000E+00	112000.0
ASSIGN(MW, LA)	0.0000000E+00	60000.00
ASSIGN(MW, CHIC)	0.0000000E+00	20000.00
ASSIGN(MW, NY)	1.000000	50000.00
ASSIGN(MW, ATL)	0.0000000E+00	50000.00
ASSIGN(E, LA)	0.0000000E+00	96000.00
ASSIGN(E, CHIC)	0.0000000E+00	60000.00
ASSIGN(E, NY)	1.000000	24000.00
ASSIGN(E, ATL)	0.0000000E+00	60000.00
ASSIGN(S, LA)	0.0000000E+00	64000.00
ASSIGN(S, CHIC)	0.0000000E+00	40000.00
ASSIGN(S, NY)	1.000000	40000.00
ASSIGN(S, ATL)	0.0000000E+00	16000.00

FIGURE 8
(Continued)

ROW	SLACK OR SURPLUS	DUAL PRICE
1	242000.0	-1.000000
2	0.0000000E+00	0.0000000E+00
3	0.0000000E+00	0.0000000E+00
4	1.000000	0.0000000E+00
5	0.0000000E+00	0.0000000E+00
6	1.000000	0.0000000E+00
7	0.0000000E+00	0.0000000E+00
8	0.0000000E+00	0.0000000E+00
9	0.0000000E+00	0.0000000E+00
10	1.000000	0.0000000E+00
11	0.0000000E+00	0.0000000E+00
12	0.0000000E+00	0.0000000E+00
13	0.0000000E+00	0.0000000E+00
14	1.000000	0.0000000E+00
15	0.0000000E+00	0.0000000E+00
16	0.0000000E+00	0.0000000E+00
17	0.0000000E+00	0.0000000E+00
18	0.0000000E+00	0.0000000E+00
19	0.0000000E+00	0.0000000E+00
20	0.0000000E+00	0.0000000E+00
21	0.0000000E+00	0.0000000E+00
22	0.0000000E+00	-1.000000
23	0.0000000E+00	0.0000000E+00
24	0.0000000E+00	0.0000000E+00
25	0.0000000E+00	0.0000000E+00
26	0.0000000E+00	0.0000000E+00
27	0.0000000E+00	0.0000000E+00
28	0.0000000E+00	-1.000000
29	0.0000000E+00	0.0000000E+00
30	0.0000000E+00	0.0000000E+00
31	0.0000000E+00	0.0000000E+00
32	0.0000000E+00	-1.000000
33	0.0000000E+00	0.0000000E+00
34	0.0000000E+00	0.0000000E+00
35	0.0000000E+00	0.0000000E+00
36	0.0000000E+00	-1.000000
37	0.0000000E+00	0.0000000E+00

FIGURE 8
(Continued)

PROBLEMS

Group A

1 Coach Night is trying to choose the starting lineup for the basketball team. The team consists of seven players who have been rated (on a scale of 1 = poor to 3 = excellent) according to their ball-handling, shooting, rebounding, and defensive abilities. The positions that each player is allowed to play and the player's abilities are listed in Table 9.

The five-player starting lineup must satisfy the following restrictions:

1 At least 4 members must be able to play guard, at least 2 members must be able to play forward, and at least 1 member must be able to play center.

2 The average ball-handling, shooting, and rebounding level of the starting lineup must be at least 2.

3 If player 3 starts, then player 6 cannot start.

4 If player 1 starts, then players 4 and 5 must both start.

5 Either player 2 or player 3 must start.

Given these constraints, Coach Night wants to maximize the total defensive ability of the starting team. Formulate an IP that will help him choose his starting team.

TABLE 9

Player	Position	Ball-Handling	Shooting	Rebounding	Defense
1	G	3	3	1	3
2	C	2	1	3	2
3	G-F	2	3	2	2
4	F-C	1	3	3	1
5	G-F	3	3	3	3
6	F-C	3	1	2	3
7	G-F	3	2	2	1

2 Because of excessive pollution on the Momiss River, the state of Momiss is going to build pollution control stations. Three sites (1, 2, and 3) are under consideration. Momiss is interested in controlling the pollution levels of two pollutants (1 and 2). The state legislature requires that at least 80,000 tons of pollutant 1 and at least 50,000 tons of pollutant 2 be removed from the river. The relevant data for this problem are shown in Table 10. Formulate an IP to minimize the cost of meeting the state legislature's goals.

3 A manufacturer can sell product 1 at a profit of \$2/unit and product 2 at a profit of \$5/unit. Three units of raw material are needed to manufacture 1 unit of product 1, and

TABLE 10

Site	Cost of Building Station ($)	Cost of Treating 1 Ton Water ($)	Amount Removed per Ton of Water	
			Pollutant 1	Pollutant 2
1	100,000	20	0.40	0.30
2	60,000	30	0.25	0.20
3	40,000	40	0.20	0.25

TABLE 11

From	To ($)		
	Region 1	Region 2	Region 3
New York	20	40	50
Los Angeles	48	15	26
Chicago	26	35	18
Atlanta	24	50	35

6 units of raw material are needed to manufacture 1 unit of product 2. A total of 120 units of raw material are available. If any of product 1 is produced, a setup cost of $10 is incurred, and if any of product 2 is produced, a setup cost of $20 is incurred. Formulate an IP to maximize profits.

4 Suppose we add the following restriction to Example 1 (Stockco): If investments 2 and 3 are chosen, then investment 4 must be chosen. What constraints would be added to the formulation given in the text?

5 How would the following restrictions modify the formulation of Example 6 (Dorian car sizes)? (Do each part separately.)

a If midsize cars are produced, then compacts must also be produced.

b Either compacts or large cars must be manufactured.

6 To graduate from Basketweavers University with a major in operations research, a student must complete at least two math courses, at least two OR courses, and at least two computer courses. Some courses can be used to fulfill more than one requirement: Calculus can fulfill the math requirement; operations research, math and OR requirements; data structures, computer and math requirements; business statistics, math and OR requirements; computer simulation, OR and computer requirements; introduction to computer programming, computer requirement; and forecasting, OR and math requirements.

Some courses are prerequisites for others: Calculus is a prerequisite for business statistics; introduction to computer programming is a prerequisite for computer simulation and for data structures; and business statistics is a prerequisite for forecasting. Formulate an IP that minimizes the number of courses needed to satisfy the major requirements.

7 In Example 7 (Euing Gas), suppose that $x = 300$. What would be the values of y_1, y_2, y_3, z_1, z_2, z_3, and z_4? How about if $x = 1{,}200$?

8 Formulate an IP to solve the Dorian Auto problem for the advertising data that exhibit increasing returns as more ads are placed in a magazine (pages 495–496).

9 How can integer programming be used to ensure that the variable x can assume only the values 1, 2, 3, and 4?

10 If x and y are integers, how could you ensure that $x + y \leq 3$, $2x + 5y \leq 12$, or both are satisfied by x and y?

11 If x and y are both integers, how would you ensure that whenever $x \leq 2$, then $y \leq 3$?

12 A company is considering opening warehouses in four cities: New York, Los Angeles, Chicago, and Atlanta. Each warehouse can ship 100 units per week. The weekly fixed cost of keeping each warehouse open is $400 for New York, $500 for Los Angeles, $300 for Chicago, and $150 for Atlanta. Region 1 of the country requires 80 units per week, region 2 requires 70 units per week, and region 3 requires 40 units per week. The costs (including production and shipping costs) of sending one unit from a plant to a region are shown in Table 11. We want to meet weekly demands at minimum cost, subject to the preceding information and the following restrictions:

1 If the New York warehouse is opened, then the Los Angeles warehouse must be opened.

2 At most two warehouses can be opened.

3 Either the Atlanta or the Los Angeles warehouse must be opened.

Formulate an IP that can be used to minimize the weekly costs of meeting demand.

13 Glueco produces three types of glue on two different production lines. Each line can be utilized by up to seven workers at a time. Workers are paid $500 per week on production line 1, and $900 per week on production line 2. A week of production costs $1,000 to set up production line 1 and $2,000 to set up production line 2. During a week on a production line, each worker produces the number of units of glue shown in Table 12. Each week, at least 120 units of glue 1, at least 150 units of glue 2, and at least 200 units of glue 3 must be produced. Formulate an IP to minimize the total cost of meeting weekly demands.

14† The manager of State University's DED computer wants to be able to access five different files. These files are scattered on 10 disks as shown in Table 13. The amount of storage required by each disk is as follows: disk 1, 3K; disk 2, 5K; disk 3, 1K; disk 4, 2K; disk 5, 1K; disk 6, 4K; disk 7, 3K; disk 8, 1K; disk 9, 2K; disk 10, 2K.

a Formulate an IP that determines a set of disks requiring the minimum amount of storage such that each

TABLE 12

Production Line	Glue		
	1	2	3
1	20	30	40
2	50	35	45

†Based on Day (1965).

TABLE 13

File	Disk 1	2	3	4	5	6	7	8	9	10
1	x	x		x	x			x	x	
2	x		x							
3		x			x		x			x
4			x			x		x		
5	x	x		x		x	x		x	x

file is on at least one of the disks. For a given disk, we must either store the entire disk or store none of the disk; we cannot store part of a disk.

b Modify your formulation so that if disk 3 or disk 5 is used, then disk 2 must also be used.

15 Fruit Computer produces two types of computers: Pear computers and Apricot computers. Relevant data are given in Table 14. A total of 3,000 chips and 1,200 hours of labor are available. Formulate an IP to help Fruit maximize profits.

16 The Lotus Point Condo Project will contain both homes and apartments. The site can accommodate up to 10,000 dwelling units. The project must contain a recreation project: either a swimming–tennis complex or a sailboat marina, but not both. If a marina is built, then the number of homes in the project must be at least triple the number of apartments in the project. A marina will cost $1.2 million, and a swimming–tennis complex will cost $2.8 million. The developers believe that each apartment will yield revenues with an NPV of $48,000, and each home will yield revenues with an NPV of $46,000. Each home (or apartment) costs $40,000 to build. Formulate an IP to help Lotus Point maximize profits.

17 A product can be produced on four different machines. Each machine has a fixed setup cost, variable production costs per-unit-processed, and a production capacity given in Table 15. A total of 2,000 units of the product must be produced. Formulate an IP whose solution will tell us how to minimize total costs.

TABLE 14

Computer	Labor	Chips	Equipment Costs ($)	Selling Price ($)
Pear	1 hour	2	5,000	400
Apricot	2 hours	5	7,000	900

TABLE 15

Machine	Fixed Cost ($)	Variable Cost per Unit ($)	Capacity
1	1,000	20	900
2	920	24	1,000
3	800	16	1,200
4	700	28	1,600

TABLE 16

	Book 1	2	3	4	5
Maximum Demand	5,000	4,000	3,000	4,000	3,000
Variable Cost ($)	25	20	15	18	22
Sales Price ($)	50	40	38	32	40
Fixed Cost ($ Thousands)	80	50	60	30	40

18 Use LINDO, LINGO, or Excel Solver to find the optimal solution to the following IP:

Bookco Publishers is considering publishing five textbooks. The maximum number of copies of each textbook that can be sold, the variable cost of producing each textbook, the sales price of each textbook, and the fixed cost of a production run for each book are given in Table 16. Thus, for example, producing 2,000 copies of book 1 brings in a revenue of $2,000(50) = \$100,000$ but costs $80,000 + 25(2,000) = \$130,000$. Bookco can produce at most 10,000 books if it wants to maximize profit.

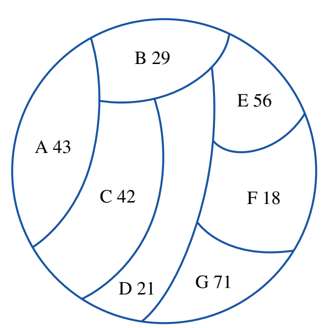
19 Comquat owns four production plants at which personal computers are produced. Comquat can sell up to 20,000 computers per year at a price of $3,500 per computer. For each plant the production capacity, the production cost per computer, and the fixed cost of operating a plant for a year are given in Table 17. Determine how Comquat can maximize its yearly profit from computer production.

20 WSP Publishing sells textbooks to college students. WSP has two sales reps available to assign to the A–G state area. The number of college students (in thousands) in each state is given in Figure 9. Each sales rep must be assigned to two adjacent states. For example, a sales rep could be assigned to A and B, but not A and D. WSP's goal is to

TABLE 17

Plant	Production Capacity	Plant Fixed Cost ($ Million)	Cost per Computer ($)
1	10,000	9	1,000
2	8,000	5	1,700
3	9,000	3	2,300
4	6,000	1	2,900

FIGURE 9



maximize the number of total students in the states assigned to the sales reps. Formulate an IP whose solution will tell you where to assign the sales reps. Then use LINDO to solve your IP.

21 Eastinghouse sells air conditioners. The annual demand for air conditioners in each region of the country is as follows: East, 100,000; South, 150,000; Midwest, 110,000; West, 90,000. Eastinghouse is considering building the air conditioners in four different cities: New York, Atlanta, Chicago, and Los Angeles. The cost of producing an air conditioner in a city and shipping it to a region of the country is given in Table 18. Any factory can produce as many as 150,000 air conditioners per year. The annual fixed cost of operating a factory in each city is given in Table 19. At least 50,000 units of the Midwest demand for air conditioners must come from New York, or at least 50,000 units of the Midwest demand must come from Atlanta. Formulate an IP whose solution will tell Eastinghouse how to minimize the annual cost of meeting demand for air conditioners.

22 Consider the following puzzle. You are to pick out 4 three-letter "words" from the following list:

DBA DEG ADI FFD GHI BCD FDF BAI

For each word, you earn a score equal to the position that the word's third letter appears in the alphabet. For example, DBA earns a score of 1, DEG earns a score of 7, and so on. Your goal is to choose the four words that maximize your total score, subject to the following constraint: The sum of the positions in the alphabet for the first letter of each word chosen must be at least as large as the sum of the positions in the alphabet for the second letter of each word chosen. Formulate an IP to solve this problem.

23 At a machine tool plant, five jobs must be completed each day. The time it takes to do each job depends on the machine used to do the job. If a machine is used at all, there is a setup time required. The relevant times are given in Table 20. The company's goal is to minimize the sum of the setup and machine operation times needed to complete all jobs. Formulate and solve (with LINDO, LINGO, or Excel Solver) an IP whose solution will do this.

TABLE 18

City	Price by Region ($)			
	East	South	Midwest	West
New York	206	225	230	290
Atlanta	225	206	221	270
Chicago	230	221	208	262
Los Angeles	290	270	262	215

TABLE 19

City	Annual Fixed Cost ($ Million)
New York	6
Atlanta	5.5
Chicago	5.8
Los Angeles	6.2

TABLE 20

Machine	Job					Machine Setup Time (Minutes)
	1	2	3	4	5	
1	42	70	93	X	X	30
2	X	85	45	X	X	40
3	58	X	X	37	X	50
4	58	X	55	X	38	60
5	X	60	X	54	X	20

Group B

24[†] Breadco Bakeries is a new bakery chain that sells bread to customers throughout the state of Indiana. Breadco is considering building bakeries in three locations: Evansville, Indianapolis, and South Bend. Each bakery can bake as many as 900,000 loaves of bread each year. The cost of building a bakery at each site is $5 million in Evansville, $4 million in Indianapolis, and $4.5 million in South Bend. To simplify the problem, we assume that Breadco has only three customers, whose demands each year are 700,000 loaves (customer 1); 400,000 loaves (customer 2); and 300,000 loaves (customer 3). The total cost of baking and shipping a loaf of bread to a customer is given in Table 21.

Assume that future shipping and production costs are discounted at a rate of $11\frac{1}{9}$% per year. Assume that once built, a bakery lasts forever. Formulate an IP to minimize Breadco's total cost of meeting demand (present and future). (*Hint:* You will need the fact that for $x < 1$, $a + ax + ax^2 + ax^3 + \cdots = a/(1 - x)$.) How would you modify the formulation if either Evansville or South Bend must produce at least 800,000 loaves per year?

25[‡] Speaker's Clearinghouse must disburse sweepstakes checks to winners in four different regions of the country: Southeast (SE), Northeast (NE), Far West (FW), and Midwest (MW). The average daily amount of the checks written to winners in each region of the country is as follows: SE, $40,000; NE, $60,000; FW, $30,000; MW, $50,000. Speaker's must issue the checks the day they find out a customer has won. They can delay winners from quickly cashing their checks by giving a winner a check drawn on an out-of-the-way bank (this will cause the check to clear

TABLE 21

From	To		
	Customer 1	Customer 2	Customer 3
Evansville	16¢	34¢	26¢
Indianapolis	40¢	30¢	35¢
South Bend	45¢	45¢	23¢

[†]Based on Efroymson and Ray (1966).
[‡]Based on Shanker and Zoltners (1972).

slowly). Four bank sites are under consideration: Frosbite Falls, Montana (FF), Redville, South Carolina (R), Painted Forest, Arizona (PF), and Beanville, Maine (B). The annual cost of maintaining an account at each bank is as follows: FF, \$50,000; R, \$40,000; PF, \$30,000; B, \$20,000. Each bank has a requirement that the average daily amount of checks written cannot exceed \$90,000. The average number of days it takes a check to clear is given in Table 22. Assuming that money invested by Speaker's earns 15% per year, where should the company have bank accounts, and from which bank should a given customer's check be written?

26[†] Governor Blue of the state of Berry is attempting to get the state legislature to gerrymander Berry's congressional districts. The state consists of 10 cities, and the numbers of registered Republicans and Democrats (in thousands) in each city are shown in Table 23. Berry has five congressional representatives. To form congressional districts, cities must be grouped according to the following restrictions:

1 All voters in a city must be in the same district.

2 Each district must contain between 150,000 and 250,000 voters (there are no independent voters).

Governor Blue is a Democrat. Assume that each voter always votes a straight party ticket. Formulate an IP to help Governor Blue maximize the number of Democrats who will win congressional seats.

27[‡] The Father Domino Company sells copying machines. A major factor in making a sale is Domino's quick service. Domino sells copiers in six cities: Boston, New York, Philadelphia, Washington, Providence, and Atlantic City. The annual sales of copiers projected depend on whether a service representative is within 150 miles of a city (see Table 24).

Each copier costs \$500 to produce and sells for \$1,000. The annual cost per service representative is \$80,000. Domino must determine in which of its markets to base a service representative. Only Boston, New York, Philadelphia, and Washington are under consideration as bases for service representative. The distance (in miles) between the cities is shown in Table 25. Formulate an IP that will help Domino maximize annual profits.

28[§] Thailand inducts naval draftees at three drafting centers. Then the draftees must each be sent to one of three naval bases for training. The cost of transporting a draftee from a drafting center to a base is given in Table 26. Each year, 1,000 men are inducted at center 1; 600 at center 2; and 700 at center 3. Base 1 can train 1,000 men a year, base 2, 800 men; and base 3, 700 men. After the inductees are trained, they are sent to Thailand's main naval base (B). They may be transported on either a small ship or a large ship. It costs \$5,000 plus \$2 per mile to use a small ship. A small ship can transport up to 200 men to the main base and may visit up to two bases on its way to the main base. Seven small and five large ships are available. It costs \$10,000 plus \$3 per mile to use a large ship. A large ship may visit up to

TABLE 22

Region	FF	R	PF	B
SE	7	2	6	5
NE	8	4	5	3
FW	4	8	2	11
MW	5	4	7	5

TABLE 23

City	Republicans	Democrats
1	80	34
2	60	44
3	40	44
4	20	24
5	40	114
6	40	64
7	70	14
8	50	44
9	70	54
10	70	64

TABLE 24

Representative Within 150 Miles?	Sales					
	Boston	N.Y.	Phila.	Wash.	Prov.	Atl. City
Yes	700	1,000	900	800	400	450
No	500	750	700	450	200	300

TABLE 25

	Boston	N.Y.	Phila.	Wash.
Boston	0	222	310	441
New York	222	0	89	241
Philadelphia	310	89	0	146
Washington	441	241	146	0
Providence	47	186	255	376
Atlantic City	350	123	82	178

TABLE 26

	To (\$)		
From	Base 1	Base 2	Base 3
Center 1	200	200	300
Center 2	300	400	220
Center 3	300	400	250

[†]Based on Garfinkel and Nemhauser (1970).
[‡]Based on Gelb and Khumawala (1984).
[§]Based on Choypeng, Puakpong, and Rosenthal (1986).

three bases on its way to the main base and may transport up to 500 men. The possible "tours" for each type of ship are given in Table 27.

Assume that the assignment of draftees to training bases is done using the transportation method. Then formulate an IP that will minimize the total cost incurred in sending the men from the training bases to the main base. (*Hint:* Let y_{ij} = number of men sent by tour i from base j to main base (B) on a small ship, x_{ij} = number of men sent by tour i from base j to B on a large ship, S_i = number of times tour i is used by a small ship, and L_i = number of times tour i is used by a large ship.)

29 You have been assigned to arrange the songs on the cassette version of Madonna's latest album. A cassette tape has two sides (1 and 2). The songs on each side of the cassette must total between 14 and 16 minutes in length. The length and type of each song are given in Table 28. The assignment of songs to the tape must satisfy the following conditions:

1 Each side must have exactly two ballads.

2 Side 1 must have at least three hit songs.

3 Either song 5 or song 6 must be on side 1.

4 If songs 2 and 4 are on side 1, then song 5 must be on side 2.

Explain how you could use an integer programming formulation to determine whether there is an arrangement of songs satisfying these restrictions.

30 Cousin Bruzie of radio station WABC schedules radio commercials in 60-second blocks. This hour, the station has sold commercial time for commercials of 15, 16, 20, 25, 30, 35, 40, and 50 seconds. Formulate an integer programming model that can be used to determine the minimum number of 60-second blocks of commercials that must be scheduled to fit in all the current hour's commercials. (*Hint:* Certainly no more than eight blocks of time are needed. Let $y_i = 1$ if block i is used and $y_i = 0$ otherwise).

31[†] A Sunco oil delivery truck contains five compartments, holding up to 2,700, 2,800, 1,100, 1,800, and 3,400 gallons of fuel, respectively. The company must deliver three types of fuel (super, regular, and unleaded) to a customer. The demands, penalty per gallon short, and the maximum allowed shortage are given in Table 29. Each compartment of the truck can carry only one type of gasoline. Formulate an IP whose solution will tell Sunco how to load the truck in a way that minimizes shortage costs.

32[‡] Simon's Mall has 10,000 sq ft of space to rent and wants to determine the types of stores that should occupy the mall. The minimum number and maximum number of each type of store (along with the square footage of each type) is given in Table 30. The annual profit made by each type of store will, of course, depend on how many stores of that type are in the mall. This dependence is given in Table 31 (all profits are in units of $10,000). Thus, if there are two department stores in the mall, each department store earns $210,000 profit per year. Each store pays 5% of its annual profit as rent to Simon's. Formulate an IP whose solution will tell Simon's how to maximize rental income from the mall.

33[§] Boris Milkem's financial firm owns six assets. The expected sales price (in millions of dollars) for each asset is given in Table 32. If asset 1 is sold in year 2, the firm receives $20 million. To maintain a regular cash flow, Milkem must sell at least $20 million of assets during year 1, at least $30 million worth during year 2, and at least $35 million worth during year 3. Set up an IP that Milkem can

TABLE 27

Tour Number	Locations Visited	Miles Traveled
1	B–1–B	370
2	B–1–2–B	515
3	B–2–3–B	665
4	B–2–B	460
5	B–3–B	600
6	B–1–3–B	640
7	B–1–2–3–B	720

TABLE 28

Song	Type	Length (in minutes)
1	Ballad	4
2	Hit	5
3	Ballad	3
4	Hit	2
5	Ballad	4
6	Hit	3
7		5
8	Ballad and hit	4

TABLE 29

Type of Gasoline	Demand	Cost per Gallon Short ($)	Maximum Allowed Shortage
Super	2,900	10	500
Regular	4,000	8	500
Unleaded	4,900	6	500

TABLE 30

Store Type	Square Footage	Minimum	Maximum
Jewelry	500	1	3
Shoe	600	1	3
Department	1,500	1	3
Book	700	0	3
Clothing	900	1	3

[†]Based on Brown (1987).
[‡]Based on Bean et al. (1988).
[§]Based on Bean, Noon, and Salton (1987).

TABLE 31

Type of Store	Number of Stores 1	2	3
Jewelry	9	8	7
Shoe	10	9	5
Department	27	21	20
Book	16	9	7
Clothing	17	13	10

TABLE 32

Asset	Sold In Year 1	Year 2	Year 3
1	15	20	24
2	16	18	21
3	22	30	36
4	10	20	30
5	17	19	22
6	19	25	29

use to determine how to maximize total revenue from assets sold during the next three years. In implementing this model, how could the idea of a rolling planning horizon be used?

34† The Smalltown Fire Department currently has seven conventional ladder companies and seven alarm boxes. The two closest ladder companies to each alarm box are given in Table 33. The city fathers want to maximize the number of conventional ladder companies that can be replaced with tower ladder companies. Unfortunately, political considerations dictate that a conventional company can be replaced only if, after replacement, at least one of the two closest companies to each alarm box is still a conventional company.

a Formulate an IP that can be used to maximize the number of conventional companies that can be replaced by tower companies.

b Suppose $y_k = 1$ if conventional company k is replaced. Show that if we let $z_k = 1 - y_k$, the answer in part (a) is equivalent to a set-covering problem.

35‡ A power plant has three boilers. If a given boiler is operated, it can be used to produce a quantity of steam (in tons) between the minimum and maximum given in Table 34. The cost of producing a ton of steam on each boiler is also given. Steam from the boilers is used to produce power on three turbines. If operated, each turbine can process an amount of steam (in tons) between the minimum and maximum given in Table 35. The cost of processing a ton of steam and the power produced by each turbine is also given. Formulate an IP that can be used to minimize the cost of producing 8,000 kwh of power.

†Based on Walker (1974).

‡Based on Cavalieri, Roversi, and Ruggeri (1971).

TABLE 33

Alarm Box	Two Closest Ladder Companies
1	2, 3
2	3, 4
3	1, 5
4	2, 6
5	3, 6
6	4, 7
7	5, 7

TABLE 34

Boiler Number	Minimum Steam	Maximum Steam	Cost/Ton ($)
1	500	1,000	10
2	300	900	8
3	400	800	6

TABLE 35

Turbine Number	Minimum	Maximum	Kwh per Ton of Steam	Processing Cost per Ton ($)
1	300	600	4	2
2	500	800	5	3
3	600	900	6	4

36§ An Ohio company, Clevcinn, consists of three subsidiaries. Each has the respective average payroll, unemployment reserve fund, and estimated payroll given in Table 36. (All figures are in millions of dollars.) Any employer in the state of Ohio whose reserve/average payroll ratio is less than 1 must pay 20% of its estimated payroll in unemployment insurance premiums or 10% if the ratio is at least 1. Clevcinn can aggregate its subsidiaries and label them as separate employers. For instance, if subsidiaries 2 and 3 are aggregated, they must pay 20% of their combined payroll in unemployment insurance premiums. Formulate an IP that can be used to determine which subsidiaries should be aggregated.

37 The Indiana University Business School has two rooms that each seat 50 students, one room that seats 100 students, and one room that seats 150 students. Classes are held five hours a day. The four types of requests for rooms are listed in Table 37. The business school must decide how many requests of each type should be assigned to each type of room. Penalties for each type of assignment are given in Table 38. An X means that a request must be satisfied by a room of adequate size. Formulate an IP whose solution will tell the business school how to assign classes to rooms in a way that minimizes total penalties.

§Based on Salkin (1979).

TABLE 36

Subsidiary	Average Payroll	Reserve	Estimated Payroll
1	300	400	350
2	600	510	400
3	800	600	500

TABLE 37

Type	Size Room Requested (Seats)	Hours Requested	Number of Requests
1	50	2, 3, 4	3
2	150	1, 2, 3	1
3	100	5	1
4	50	1, 2	2

TABLE 38

Size Requested	Sizes Used to Satisfy Request 50	100	150	Penalty
50	0	2	4	100* (Hours requested)
100	X	0	1	100* (Hours requested)
150	X	X	0	100* (Hours requested)

38 A company sells seven types of boxes, ranging in volume from 17 to 33 cubic feet. The demand and size of each box are given in Table 39. The variable cost (in dollars) of producing each box is equal to the box's volume. A fixed cost of $1,000 is incurred to produce any of a particular box. If the company desires, demand for a box may be satisfied by a box of larger size. Formulate and solve (with LINDO, LINGO, or Excel Solver) an IP whose solution will minimize the cost of meeting the demand for boxes.

39 Huntco produces tomato sauce at five different plants. The capacity (in tons) of each plant is given in Table 40. The tomato sauce is stored at one of three warehouses. The per-ton cost (in hundreds of dollars) of producing tomato sauce at each plant and shipping it to each warehouse is given in Table 41. Huntco has four customers. The cost of shipping a ton of sauce from each warehouse to each customer is as given in Table 42. Each customer must be delivered the amount (in tons) of sauce given in Table 43.

TABLE 39

	Box 1	2	3	4	5	6	7
Size	33	30	26	24	19	18	17
Demand	400	300	500	700	200	400	200

TABLE 40

	Plant 1	2	3	4	5
Tons	300	200	300	200	400

TABLE 41

From	To Warehouse 1	Warehouse 2	Warehouse 3
Plant 1	8	10	12
Plant 2	7	5	7
Plant 3	8	6	5
Plant 4	5	6	7
Plant 5	7	6	5

TABLE 42

From	To Customer 1	Customer 2	Customer 3	Customer 4
Warehouse 1	40	80	90	50
Warehouse 2	70	70	60	80
Warehouse 3	80	30	50	60

TABLE 43

	Customer 1	2	3	4
Demand	200	300	150	250

a Formulate a balanced transportation problem whose solution will tell us how to minimize the cost of meeting the customer demands.

b Modify this problem if these are annual demands and there is a fixed annual cost of operating each plant and warehouse. These costs (in thousands) are given in Table 44.

40 To satisfy telecommunication needs for the next 20 years, Telstar Corporation estimates that the number of circuits required between the United States and Germany, France, Switzerland, and the United Kingdom will be as given in Table 45.

Two types of circuits may be created: cable and satellite. Two types of cable circuits (TA7 and TA8) are available. The fixed cost of building each type of cable and the circuit capacity of each type are as given in Table 46.

TA7 and TA8 cable go underseas from the United States to the English Channel. Thus, it costs an additional amount to extend these circuits to other European countries. The annual variable cost per circuit is given in Table 47.

TABLE 44[†]

Facility	Fixed Annual Cost (in Thousands) $
Plant 1	35
Plant 2	45
Plant 3	40
Plant 4	42
Plant 5	40
Warehouse 1	30
Warehouse 2	40
Warehouse 3	30

[†]Based on Geoffrion and Graves (1974).

TABLE 45

Country	Required Circuits
France	20,000
Germany	60,000
Switzerland	16,000
United Kingdom	60,000

TABLE 46

Cable Type	Fixed Operating Cost ($ Billion)	Capacity
TA7	1.6	8,500
TA8	2.3	37,800

TABLE 47

Country	Variable Cost per Circuit ($)
France	0
Germany	310
Switzerland	290
United Kingdom	0

To create and use a satellite circuit, Telstar must launch a satellite, and each country using the satellite must have an earth station(s) to receive the signal. It costs $3 billion to launch a satellite. Each launched satellite can handle up to 140,000 circuits. All earth stations have a maximum capacity of 190 circuits and cost $6,000 per year to operate. Formulate an integer programming model to help determine how to supply the needed circuits and minimize total cost incurred during the next 20 years.

Then use LINDO (or LINGO) to find a near optimal solution. LINDO after 300 pivots did not think it had an optimal solution! By the way, do not require that the number of cable or satellite circuits in a country be integers, or your model will never get solved! For some variables, however, the integer requirement is vital![†]

41 A large drug company must determine how many sales representatives to assign to each of four sales districts. The cost of having n representatives in a district is ($88,000 + $80,000$n$) per year. If a rep is based in a given district, the time it takes to complete a call on a doctor is given in Table 48 (times are in hours).

Each sales rep can work up to 160 hours per month. Each month the number of calls given in Table 49 must be made in each district. A fractional number of representatives in a district is not permissible. Determine how many representatives should be assigned to each district.

42[‡] In this assignment, we will use integer programming and the concept of bond duration to show how Wall Street firms can select an optimal bond portfolio. The *duration* of a bond (or any stream of payments) is defined as follows: Let $C(t)$ be the payment of the bond at time t ($t = 1, 2, \ldots, n$). Let r = market interest rate. If the time-weighted average of the bond's payments is given by:

$$\sum_{t=1}^{t=n} tC(t)/(1 + r)^t$$

and the market price P of the bond is given by:

$$\sum_{t=1}^{t=n} C(t)/(1 + r)^t$$

then the duration of the bond D is given by:

$$D = (1/P) \sum_{t=1}^{n} \frac{tC(t)}{}$$

Thus, the duration of a bond measures the "average" time (in years) at which a randomly chosen $1 of NPV is received. Suppose an insurance company needs to make payments of $20,000 every six months for the next 10 years. If the market

TABLE 48

	Actual Sales Call District			
Rep's Base District	**1**	**2**	**3**	**4**
1	1	4	5	7
2	4	1	3	5
3	5	3	1	2
4	7	5	2	1

TABLE 49

District	Number of Calls
1	50
2	80
3	100
4	60

[†]Based on Calloway, Cummins, and Freeland (1990).
[‡]Based on Strong (1989).

rate of interest is 10% per year, then this stream of payments has an NPV of $251,780 and a duration of 4.47 years. If we want to minimize the sensitivity of our bond portfolio to interest risk and still meet our payment obligations, then it has been shown that we should invest $251,780 at the beginning of year 1 in a bond portfolio having a duration equal to the duration of the payment stream.

Suppose the only cost of owning a bond portfolio is the transaction cost associated with the cost of purchasing the bonds. Let's suppose six bonds are available. The payment streams for these six bonds are given in Table 50. The transaction cost of purchasing any units of bond i equals $500 + $5 per bond purchased. Thus, purchasing one unit of bond 1 costs $505 and purchasing 10 units of bond 1 costs $550. Assume that a fractional number of bond i unit purchases is permissible, but in the interests of diversification at most 100 units of any bond can be purchased. Treasury bonds may also be purchased (with no transaction cost). A treasury bond costs $980 and has a duration of .25 year (90 days).

After computing the price and duration for each bond, use integer programming to determine the immunized bond portfolio that incurs the smallest transaction costs. You may assume the duration of your portfolio is a weighted average of the durations of the bonds included in the portfolio, where the weight associated with each bond is equal to the money invested in that bond.

43 Ford has four automobile plants. Each is capable of producing the Taurus, Lincoln, or Escort, but it can only produce one of these cars. The fixed cost of operating each plant for a year and the variable cost of producing a car of each type at each plant are given in Table 51.

Ford faces the following restrictions:

a Each plant can produce only one type of car.

b The total production of each type of car must be at a single plant; that is, for example, if any Tauruses are made at plant 1, then all Tauruses must be made there.

c If plants 3 and 4 are used, then plant 1 must also be used.

TABLE 50

Year	Available Bonds					
	Bond 1	Bond 2	Bond 3	Bond 4	Bond 5	Bond 6
1	50	100	130	20	100	120
2	60	90	130	20	100	100
3	70	80	130	20	100	80
4	80	70	130	20	100	140
5	90	60	130	20	100	100
6	100	50	130	80	100	90
7	110	40	130	40	100	110
8	120	30	130	150	100	130
9	130	20	130	200	100	180
10	1,010	1,040	1,130	1,200	1,100	950

TABLE 51

Plant	Fixed Cost ($)	Variable Cost ($)		
		Taurus	Lincoln	Escort
1	7 billion	12,000	16,000	9,000
2	6 billion	15,000	18,000	11,000
3	4 billion	17,000	19,000	12,000
4	2 billion	19,000	22,000	14,000

Each year, Ford must produce 500,000 of each type of car. Formulate an IP whose solution will tell Ford how to minimize the annual cost of producing cars.

44 Venture capital firm JD is trying to determine in which of 10 projects it should invest. It knows how much money is available for investment each of the next N years, the NPV of each project, and the cash required by each project during each of the next N years (see Table 52).

a Write a LINGO program to determine the projects in which JD should invest.

b Use your LINGO program to determine which of the 10 projects should be selected. Each project requires cash investment during the next three years. During year 1, $80 million is available for investment. During year 2, $60 million is available for investment. During year 3, $70 million is available for investment. (All figures are in millions of dollars.)

45 Write a LINGO program that can solve a fixed-charge problem of the type described in Example 3. Assume there is a limited demand for each product. Then use your program to solve a four-product, three-resource fixed-charge problem with the parameters shown in Tables 53, 54, and 55.

TABLE 52

Investment ($ Million)	Project									
	1	2	3	4	5	6	7	8	9	10
Year 1	6	9	12	15	18	21	24	27	30	35
Year 2	3	5	7	9	11	13	15	17	19	21
Year 3	5	7	9	12	12	14	16	11	20	24
NPV	20	30	40	50	60	70	80	90	100	130

TABLE 53

Resource	Resource Availability
1	40
2	60
3	80

TABLE 54

Product	Demand	Unit Profit Contribution ($)	Fixed Charge ($)
1	40	2	30
2	60	5	40
3	65	6	50
4	70	7	60

TABLE 55

	Product			
Resource Usage	1	2	3	4
1	1	2	3.5	4
2	5	6	7	9
3	3	4	5	6

9.3 The Branch-and-Bound Method for Solving Pure Integer Programming Problems

In practice, most IPs are solved by using the technique of branch-and-bound. Branch-and-bound methods find the optimal solution to an IP by efficiently enumerating the points in a subproblem's feasible region. Before explaining how branch-and-bound works, we need to make the following elementary but important observation: *If you solve the LP relaxation of a pure IP and obtain a solution in which all variables are integers, then the optimal solution to the LP relaxation is also the optimal solution to the IP.*

To see why this observation is true, consider the following IP:

$$\max z = 3x_1 + 2x_2$$
$$\text{s.t.} \quad 2x_1 + x_2 \le 6$$
$$x_1, x_2 \ge 0; x_1, x_2 \text{ integer}$$

The optimal solution to the LP relaxation of this pure IP is $x_1 = 0$, $x_2 = 6$, $z = 12$. Because this solution gives integer values to all variables, the preceding observation implies that $x_1 = 0$, $x_2 = 6$, $z = 12$ is also the optimal solution to the IP. Observe that the feasible region for the IP is a subset of the points in the LP relaxation's feasible region (see Figure 10). Thus, the optimal z-value for the IP cannot be larger than the optimal z-value for the LP relaxation. This means that the optimal z-value for the IP must be ≤ 12. But the point $x_1 = 0$, $x_2 = 6$, $z = 12$ is feasible for the IP and has $z = 12$. Thus, $x_1 = 0$, $x_2 = 6$, $z = 12$ must be optimal for the IP.

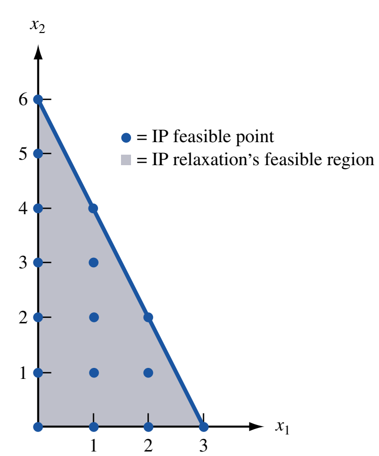


FIGURE 10
Feasible Region for an IP and Its LP Relaxation

EXAMPLE 9 Branch-and-Bound Method

The Telfa Corporation manufactures tables and chairs. A table requires 1 hour of labor and 9 square board feet of wood, and a chair requires 1 hour of labor and 5 square board feet of wood. Currently, 6 hours of labor and 45 square board feet of wood are available. Each table contributes \$8 to profit, and each chair contributes \$5 to profit. Formulate and solve an IP to maximize Telfa's profit.

Solution Let

$$x_1 = \text{number of tables manufactured}$$
$$x_2 = \text{number of chairs manufactured}$$

Because x_1 and x_2 must be integers, Telfa wants to solve the following IP:

$$\begin{aligned} \max z &= 8x_1 + 5x_2 \\ \text{s.t.} \quad x_1 + x_2 &\le 6 \quad \text{(Labor constraint)} \\ 9x_1 + 5x_2 &\le 45 \quad \text{(Wood constraint)} \\ x_1, x_2 &\ge 0;\ x_1, x_2 \text{ integer} \end{aligned}$$

The branch-and-bound method begins by solving the LP relaxation of the IP. If all the decision variables assume integer values in the optimal solution to the LP relaxation, then the optimal solution to the LP relaxation will be the optimal solution to the IP. We call the LP relaxation subproblem 1. Unfortunately, the optimal solution to the LP relaxation is $z = \frac{165}{4}$, $x_1 = \frac{15}{4}$, $x_2 = \frac{9}{4}$ (see Figure 11). From Section 9.1, we know that (optimal z-value for IP) $\le$ (optimal z-value for LP relaxation). This implies that the optimal z-value for the IP cannot exceed $\frac{165}{4}$. Thus, the optimal z-value for the LP relaxation is an **upper bound** for Telfa's profit.

Our next step is to partition the feasible region for the LP relaxation in an attempt to find out more about the location of the IP's optimal solution. We arbitrarily choose a variable that is fractional in the optimal solution to the LP relaxation—say, x_1. Now observe that every point in the feasible region for the IP must have either $x_1 \le 3$ or $x_1 \ge 4$. (Why can't a feasible solution to the IP have $3 < x_1 < 4$?) With this in mind, we "branch" on the variable x_1 and create the following two additional subproblems:

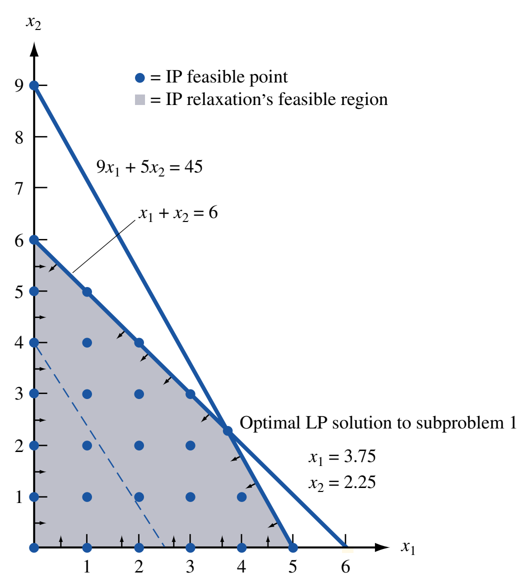


FIGURE 11
Feasible Region for Telfa Problem

Subproblem 2 Subproblem 1 + Constraint $x_1 \geq 4$.

Subproblem 3 Subproblem 1 + Constraint $x_1 \leq 3$.

Observe that neither subproblem 2 nor subproblem 3 includes any points with $x_1 = \frac{15}{4}$. This means that the optimal solution to the LP relaxation cannot recur when we solve subproblem 2 or subproblem 3.

From Figure 12, we see that every point in the feasible region for the Telfa IP is included in the feasible region for subproblem 2 or subproblem 3. Also, the feasible regions for subproblems 2 and 3 have no points in common. Because subproblems 2 and 3 were created by adding constraints involving x_1, we say that subproblems 2 and 3 were created by **branching** on x_1.

We now choose any subproblem that has not yet been solved as an LP. We arbitrarily choose to solve subproblem 2. From Figure 12, we see that the optimal solution to subproblem 2 is $z = 41$, $x_1 = 4$, $x_2 = \frac{9}{5}$ (point C). Our accomplishments to date are summarized in Figure 13.

A display of all subproblems that have been created is called a **tree.** Each subproblem is referred to as a **node** of the tree, and each line connecting two nodes of the tree is called an **arc.** The constraints associated with any node of the tree are the constraints for the LP relaxation plus the constraints associated with the arcs leading from subproblem 1 to the node. The label t indicates the chronological order in which the subproblems are solved.

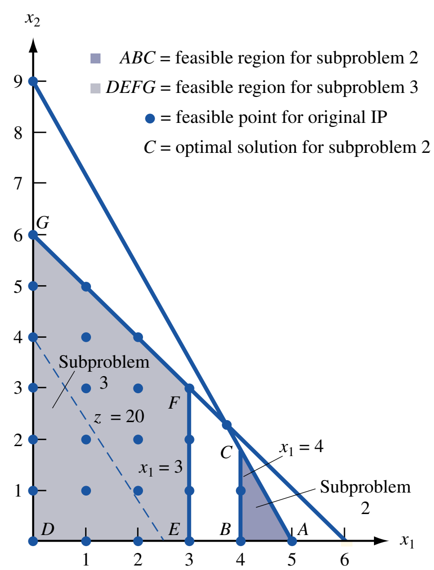


FIGURE 12
Feasible Region for Subproblems 2 and 3 of Telfa Problem

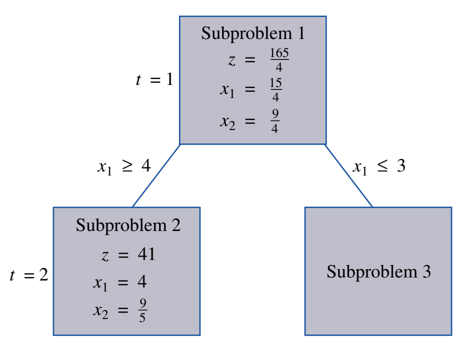


FIGURE 13
Telfa Subproblems 1 and 2 Solved

The optimal solution to subproblem 2 did not yield an all-integer solution, so we choose to use subproblem 2 to create two new subproblems. We choose a fractional-valued variable in the optimal solution to subproblem 2 and then branch on that variable. Because x_2 is the only fractional variable in the optimal solution to subproblem 2, we branch on x_2. We partition the feasible region for subproblem 2 into those points having $x_2 \geq 2$ and $x_2 \leq 1$. This creates the following two subproblems:

Subproblem 4 Subproblem 1 + Constraints $x_1 \geq 4$ and $x_2 \geq 2$ = subproblem 2 + Constraint $x_2 \geq 2$.

Subproblem 5 Subproblem 1 + Constraints $x_1 \geq 4$ and $x_2 \leq 1$ = subproblem 2 + Constraint $x_2 \leq 1$.

The feasible regions for subproblems 4 and 5 are displayed in Figure 14. The set of unsolved subproblems consists of subproblems 3, 4, and 5. We now choose a subproblem to solve. For reasons that are discussed later, we choose to solve the most recently created subproblem. (This is called the LIFO, or last-in-first-out, rule.) The LIFO rule implies that we should next solve subproblem 4 or subproblem 5. We arbitrarily choose to solve subproblem 4. From Figure 14 we see that subproblem 4 is infeasible. Thus, subproblem 4 cannot yield the optimal solution to the IP. To indicate this fact, we place an × by subproblem 4 (see Figure 15). Because any branches emanating from subproblem 4 will yield no useful information, it is fruitless to create them. When further branching on a subproblem cannot yield any useful information, we say that the subproblem (or node) is **fathomed.** Our results to date are displayed in Figure 15.

Now the only unsolved subproblems are subproblems 3 and 5. The LIFO rule implies that subproblem 5 should be solved next. From Figure 14, we see that the optimal solution to subproblem 5 is point I in Figure 14: $z = \frac{365}{9}$, $x_1 = \frac{40}{9}$, $x_2 = 1$. This solution does not yield any immediately useful information, so we choose to partition subproblem 5's feasible region by branching on the fractional-valued variable x_1. This yields two new subproblems (see Figure 16).

Subproblem 6 Subproblem 5 + Constraint $x_1 \geq 5$.

Subproblem 7 Subproblem 5 + Constraint $x_1 \leq 4$.

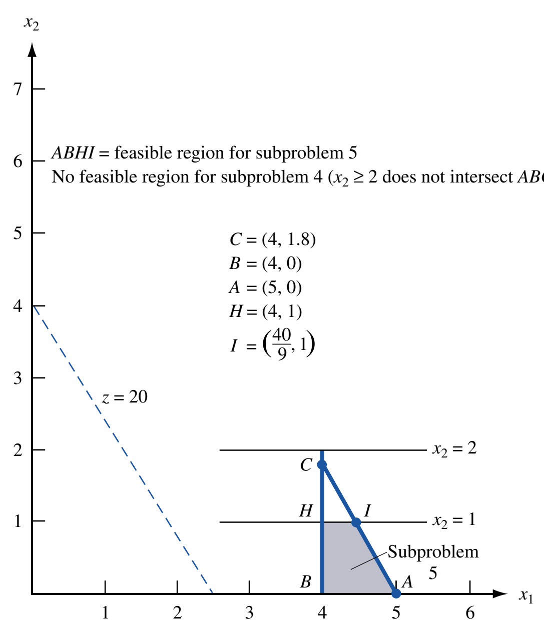


FIGURE 14
Feasible Regions for Subproblems 4 and 5 of Telfa Problem

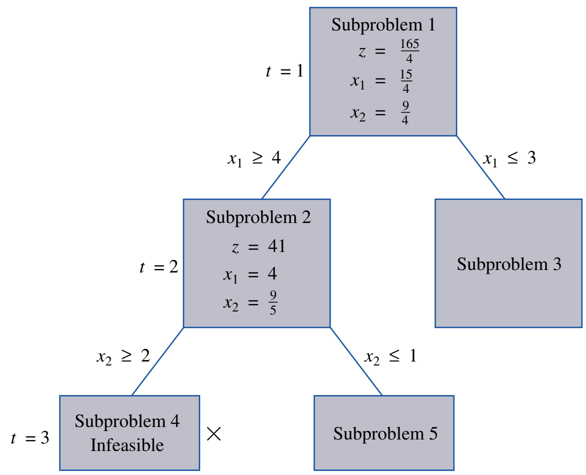


FIGURE 15
Telfa Subproblems 1, 2, and 4 Solved

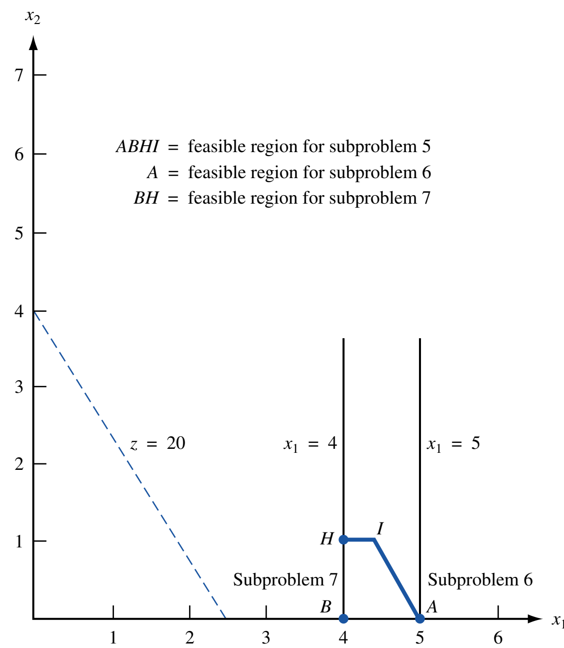


FIGURE 16
Feasible Regions for Subproblems 6 and 7 of Telfa Problem

Together, subproblems 6 and 7 include all integer points that were included in the feasible region for subproblem 5. Also, no point having $x_1 = \frac{40}{9}$ can be in the feasible region for subproblem 6 or subproblem 7. Thus, the optimal solution to subproblem 5 will not recur when we solve subproblems 6 and 7. Our tree now looks as shown in Figure 17.

Subproblems 3, 6, and 7 are now unsolved. The LIFO rule implies that we next solve subproblem 6 or subproblem 7. We arbitrarily choose to solve subproblem 7. From Figure 16, we see that the optimal solution to subproblem 7 is point H: $z = 37$, $x_1 = 4$, $x_2 = 1$. Both x_1 and x_2 assume integer values, so this solution is feasible for the original IP. We now know that subproblem 7 yields a feasible integer solution with $z = 37$. We also know that subproblem 7 cannot yield a feasible integer solution having $z > 37$. Thus, further branching on subproblem 7 will yield no new information about the optimal solution to the IP, and subproblem has been fathomed. The tree to date is pictured in Figure 18.

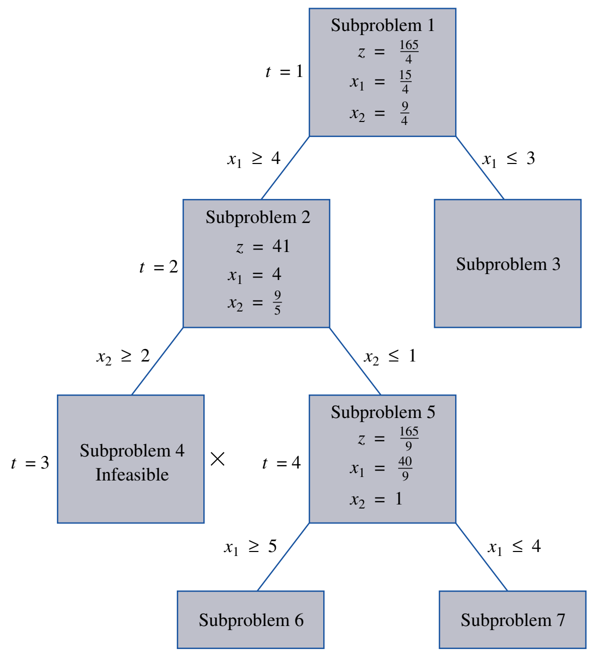


FIGURE 17
Telfa Subproblems 1, 2, 4, and 5 Solved

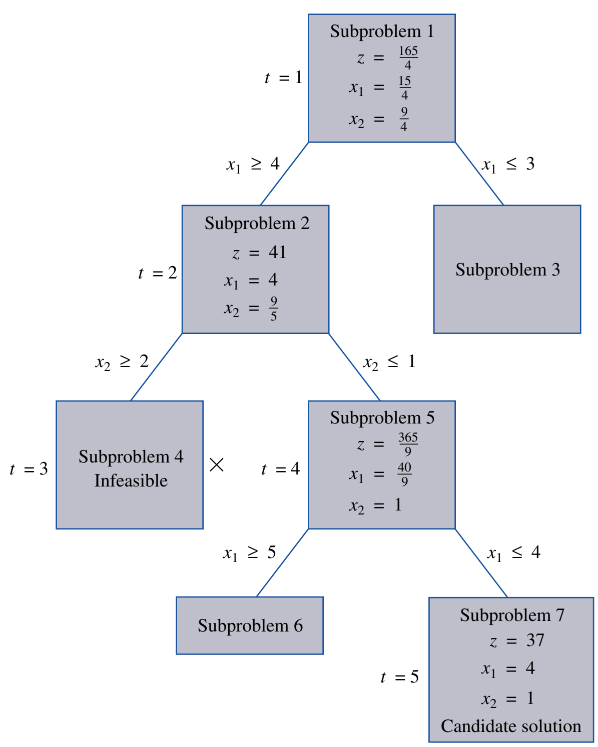


FIGURE 18
Branch-and-Bound Tree After Five Subproblems Have Been Solved

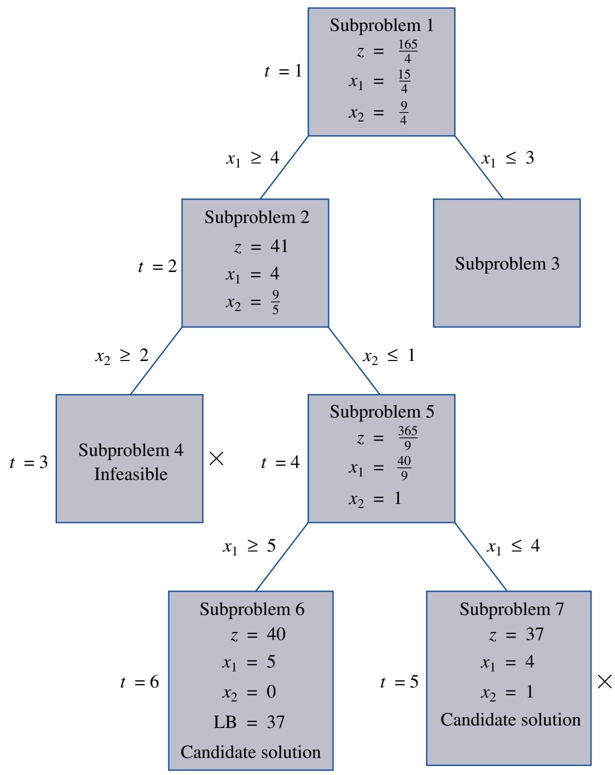


FIGURE 19
Branch-and-Bound Tree After Six Subproblems Have Been Solved

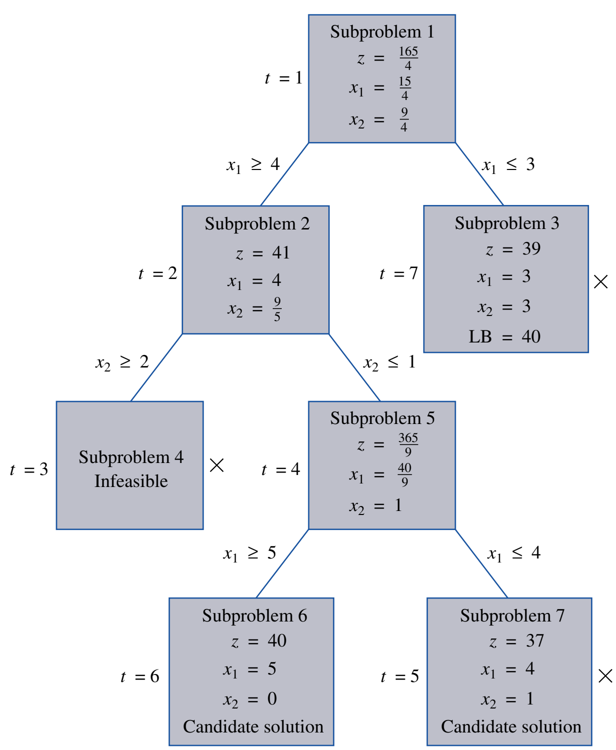


FIGURE 20
Final Branch-and-Bound Tree for Telfa Problem

A solution obtained by solving a subproblem in which all variables have integer values is a **candidate solution.** Because the candidate solution may be optimal, we must keep a candidate solution until a better feasible solution to the IP (if any exists) is found. We have a feasible solution to the original IP with $z = 37$, so we may conclude that the optimal z-value for the IP ≥ 37. Thus, the z-value for the candidate solution is a **lower bound** on the optimal z-value for the original IP. We note this by placing the notation LB = 37 in the box corresponding to the *next* solved subproblem (see Figure 19).

The only remaining unsolved subproblems are 6 and 3. Following the LIFO rule, we next solve subproblem 6. From Figure 16, we find that the optimal solution to subproblem 6 is point A: $z = 40$, $x_1 = 5$, $x_2 = 0$. All decision variables have integer values, so this is a candidate solution. Its z-value of 40 is larger than the z-value of the best previous candidate (candidate 7 with $z = 37$). Thus, subproblem 7 cannot yield the optimal solution of the IP (we denote this fact by placing an × by subproblem 7). We also update our LB to 40. Our progress to date is summarized in Figure 20.

Subproblem 3 is the only remaining unsolved problem. From Figure 12, we find that the optimal solution to subproblem 3 is point F: $z = 39$, $x_1 = x_2 = 3$. Subproblem 3 cannot yield a z-value exceeding the current lower bound of 40, so it cannot yield the optimal solution to the original IP. Therefore, we place an × by it in Figure 20. From Figure 20, we see that there are no remaining unsolved subproblems, and that only subproblem 6 can yield the optimal solution to the IP. Thus, the optimal solution to the IP is for Telfa to manufacture 5 tables and 0 chairs. This solution will contribute $40 to profits.

In using the branch-and-bound method to solve the Telfa problem, we have implicitly enumerated all points in the IP's feasible region. Eventually, all such points (except for the optimal solution) are eliminated from consideration, and the branch-and-bound procedure is complete. To show that the branch-and-bound procedure actually does consider all points in the IP's feasible region, we examine several possible solutions to the Telfa problem and show how the procedure found these points to be nonoptimal. For example, how do we know that $x_1 = 2$, $x_2 = 3$ is not optimal? This point is in the feasible region for subproblem 3, and we know that all points in the feasible region for subproblem 3 have $z \leq 39$. Thus, our analysis of subproblem 3 shows that $x_1 = 2$, $x_2 = 3$ cannot beat $z = 40$ and cannot be optimal. As another example, why isn't $x_1 = 4$, $x_2 = 2$ optimal? Following the branches of the tree, we find that $x_1 = 4$, $x_2 = 2$ is associated with subproblem 4. Because no point associated with subproblem 4 is feasible, $x_1 = 4$, $x_2 = 2$ must fail to satisfy the constraints for the original IP and thus cannot be optimal for the Telfa problem. In a similar fashion, the branch-and-bound analysis has eliminated all points x_1, x_2 (except for the optimal solution) from consideration.

For the simple Telfa problem, the use of the branch-and-bound method may seem like using a cannon to kill a fly, but for an IP in which the feasible region contains a large number of integer points, the procedure can be very efficient for eliminating nonoptimal points from consideration. For example, suppose we are applying the branch-and-bound method and our current LB = 42. Suppose we solve a subproblem that contains 1 million feasible points for the IP. If the optimal solution to this subproblem has $z < 42$, then we have eliminated 1 million nonoptimal points by solving a single LP!

The key aspects of the branch-and-bound method for solving pure IPs (mixed IPs are considered in the next section) may be summarized as follows:

Step 1 If it is unnecessary to branch on a subproblem, then it is fathomed. The following three situations result in a subproblem being fathomed: (1) The subproblem is infeasible; (2) the subproblem yields an optimal solution in which all variables have integer values; and (3) the optimal z-value for the subproblem does not exceed (in a max problem) the current LB.

Step 2 A subproblem may be eliminated from consideration in the following situations: (1) The subproblem is infeasible (in the Telfa problem, subproblem 4 was eliminated for this reason); (2) the LB (representing the z-value of the best candidate to date) is at least as large as the z-value for the subproblem (in the Telfa problem, subproblems 3 and 7 were eliminated for this reason).

Recall that in solving the Telfa problem by the branch-and-bound procedure, many seemingly arbitrary choices were made. For example, when x_1 and x_2 were both fractional in the optimal solution to subproblem 1, how did we determine the branching variable? Or how did we determine which subproblem should next be solved? The manner in which these questions are answered can result in trees that differ greatly in size and in the computer time required to find an optimal solution. Through experience and ingenuity, practitioners of the procedure have developed guidelines on how to make the necessary decisions.

Two general approaches are commonly used to determine which subproblems should be solved next. The most widely used is the LIFO rule, which chooses to solve the most recently created subproblem.† LIFO leads us down one side of the branch-and-bound tree (as in the Telfa problem) and quickly finds a candidate solution. Then we backtrack our way up to the top of the other side of the tree. For this reason, the LIFO approach is often called **backtracking.**

The second commonly used method is **jumptracking.** When branching on a node, the jumptracking approach solves all the problems created by the branching. Then it branches again on the node with the best z-value. Jumptracking often jumps from one side of the tree to the other. It usually creates more subproblems and requires more computer storage than backtracking. The idea behind jumptracking is that moving toward the subproblems with good z-values should lead us more quickly to the best z-value.

If two or more variables are fractional in a subproblem's optimal solution, then on which variable should we branch? Branching on the fractional-valued variable that has the greatest economic importance is often the best strategy. In the Nickles example, suppose the optimal solution to a subproblem had y_1 and x_{12} fractional. Our rule would say to branch on y_1 because y_1 represents the decision to operate (or not operate) a lockbox in city 1, and this is presumably a more important decision than whether region 1 payments should be sent to city 2. When more than one variable is fractional in a subproblem solution, many computer codes will branch on the lowest-numbered fractional variable. Thus, if an integer programming computer code requires that variables be numbered, they should be numbered in order of their economic importance (1 = most important).

REMARKS

1 For some IP's, the optimal solution to the LP relaxation will also be the optimal solution to the IP. Suppose the constraints of the IP are written as $A\mathbf{x} = \mathbf{b}$. If the determinant‡ of every square submatrix of A is $+1$, -1, or 0, we say that the matrix A is **unimodular.** If A is unimodular and each element of $\mathbf{b}$ is an integer, then the optimal solution to the LP relaxation will assign all variables integer values [see Shapiro (1979) for a proof] and will therefore be the optimal solution to the IP. It can be shown that the constraint matrix of any MCNFP is unimodular. Thus, as was discussed in Chapter 8, any MCNFP in which each node's net outflow and each arc's capacity are integers will have an integer-valued solution.

2 As a general rule, the more an IP looks like an MCNFP, the easier the problem is to solve by branch-and-bound methods. Thus, in formulating an IP, it is good to choose a formulation in which as many variables as possible have coefficients of $+1$, -1, and 0. To illustrate this idea, recall that the formulation of the Nickles (lockbox) problem given in Section 9.2 contained 16 constraints of the following form:

Formulation 1 $$x_{ij} \le y_j \ (i = 1, 2, 3, 4; j = 1, 2, 3, 4) \quad (25)$$

†For two subproblems created at the same time, many sophisticated methods have been developed to determine which one should be solved first. See Taha (1975) for details.

‡The determinant of a matrix is defined in Section 2.6.

As we have already seen in Section 9.2, if the 16 constraints in (25) are replaced by the following 4 constraints, then an equivalent formulation results:

Formulation 2

$$x_{11} + x_{21} + x_{31} + x_{41} \le 4y_1$$
$$x_{12} + x_{22} + x_{32} + x_{42} \le 4y_2$$
$$x_{13} + x_{23} + x_{33} + x_{43} \le 4y_3$$
$$x_{14} + x_{24} + x_{34} + x_{44} \le 4y_4$$

Because formulation 2 has $16 - 4 = 12$ fewer constraints than formulation 1, one might think that formulation 2 would require less computer time to find the optimal solution. This turns out to be untrue. To see why, recall that the branch-and-bound method begins by solving the LP relaxation of the IP. The feasible region of the LP relaxation of formulation 2 contains many more noninteger points than the feasible region of formulation 1. For example, the point $y_1 = y_2 = y_3 = y_4 = \frac{1}{4}$, $x_{11} = x_{22} = x_{33} = x_{44} = 1$ (all other x_{ij}'s equal 0) is in the feasible region for the LP relaxation of formulation 2, but not for formulation 1. The branch-and-bound method must eliminate all noninteger points before obtaining the optimal solution to the IP, so it seems reasonable that formulation 2 will require more computer time than formulation 1. Indeed, when the LINDO package was used to find the optimal solution to formulation 1, the LP relaxation yielded the optimal solution. But 17 subproblems were solved before the optimal solution was found for formulation 2. Note that formulation 2 contains the terms $4y_1$, $4y_2$, $4y_3$, and $4y_4$. These terms "disturb" the network-like structure of the lockbox problem and cause the branch-and-bound method to be less efficient.

3 When solving an IP in the real world, we are usually happy with a near-optimal solution. For example, suppose that we are solving a lockbox problem and the LP relaxation yields a cost of $200,000. This means that the optimal solution to the lockbox IP will certainly have a cost of at least $200,000. If we find a candidate solution during the course of the branch-and-bound procedure that has a cost of, say, $205,000, why bother to continue with the branch-and-bound procedure? Even if we found the optimal solution to the IP, it could not save more than $5,000 in costs over the candidate solution with $z = 205{,}000$. It might even cost more than $5,000 in computer time to find the optimal lockbox solution. For this reason, the branch-and-bound procedure is often terminated when a candidate solution is found with a z-value close to the z-value of the LP relaxation.

4 Subproblems for branch-and-bound problems are often solved using some variant of the dual simplex algorithm. To illustrate this, we return to the Telfa example. The optimal tableau for the LP relaxation of the Telfa problem is

$$z + 1.25s_1 + 0.75s_2 = 41.25$$
$$x_2 + 2.25s_1 - 0.25s_2 = 2.25$$
$$x_1 - 1.25s_1 + 0.25s_2 = 3.75$$

After solving the LP relaxation, we solved subproblem 2, which is just subproblem 1 plus the constraint $x_1 \ge 4$. Recall that the dual simplex is an efficient method for finding the new optimal solution to an LP when we know the optimal tableau and a new constraint is added to the LP. We have added the constraint $x_1 \ge 4$ (which may be written as $x_1 - e_3 = 4$). To utilize the dual simplex, we must eliminate the basic variable x_1 from this constraint and use e_3 as a basic variable for $x_1 - e_3 = 4$. Adding $-$(second row of optimal tableau) to the constraint $x_1 - e_3 = 4$, we obtain the constraint $1.25s_1 - 0.25s_2 - e_3 = 0.25$. Multiplying this constraint through by -1, we obtain $-1.25s_1 + 0.25s_2 + e_3 = -0.25$. After adding this constraint to subproblem 1's optimal tableau, we obtain the tableau in Table 56. The dual simplex method states that we should enter a variable from row 3 into the basis. Because s_1 is the only variable with a negative coefficient in row 3, s_1 will enter the basis in row 3. After the pivot, we obtain the (optimal) tableau in Table 57. Thus, the optimal solution to subproblem 2 is $z = 41$, $x_2 = 1.8$, $x_1 = 4$, $s_1 = 0.20$.

TABLE 56

Initial Tableau for Solving Subproblem 2 by Dual Simplex

	Basic Variable
$z + 1.25s_1 + 0.75s_2 = 41.25$	$z = 41.25$
$x_2 + 2.25s_1 - 0.25s_2 = 2.25$	$x_2 = 2.25$
$x_1 - 1.25s_1 + 0.25s_2 = 3.75$	$x_1 = 3.75$
$- 1.25s_1 + 0.25s_2 + e_3 = -0.25$	$e_3 = -0.25$

TABLE 57

Optimal Tableau for Solving Subproblem 2 by Dual Simplex

	Basic Variable
$z \quad + 0.20s_2 + 0.80e_3 = 41$	$z = 41$
$x_2 + 0.20s_2 + 1.8e_3 = 1.8$	$x_2 = 1.8$
$x_1 \quad - 0.80e_3 = 4$	$x_1 = 4$
$s_1 - 0.20s_2 - 0.80e_3 = 0.20$	$s_1 = 0.20$

5 In Problem 8, we show that if we create two subproblems by adding the constraints $x_k \leq i$ and $x_k \geq i + 1$, then the optimal solution to the first subproblem will have $x_k = i$ and the optimal solution to the second subproblem will have $x_k = i + 1$. This observation is very helpful when we graphically solve subproblems. For example, we know the optimal solution to subproblem 5 of Example 9 will have $x_2 = 1$. Then we can find the value of x_1 that solves subproblem 5 by choosing x_1 to be the largest integer satisfying all constraints when $x_2 = 1$.

Solver Tolerance Option for Solving IPs

When solving integer programming problems with the Excel Solver, you may go to Options and set a tolerance. A tolerance value of, say, .20, causes the Excel Solver to stop when a feasible solution is found that has an objective function value within 20% of the optimal z-value for the problem's LP relaxation. For instance, in Example 9, the optimal z-value for the LP relaxation was 41.25. With a tolerance of .20, the Solver would stop whenever a feasible integer solution is found with a z-value exceeding $(1 - .2)(41.25) = 33$. Thus, if we solved Example 9 with the Excel Solver and found a feasible integer solution having $z = 35$, then the Solver would stop because this solution would be within 20% of the LP relaxation bound.

Why set a nonzero tolerance? For many large IP problems, it might take a long time (weeks or months!) to find an optimal solution. It might take much less time to find a near-optimal solution (say, within 5% of the optimal LP relaxation). In this case, we would be much better off with a near-optimal solution, and use of the tolerance option might be appropriate.

PROBLEMS

Group A

Use branch-and-bound to solve the following IPs:

1
$$\begin{aligned} \max z &= 5x_1 + 2x_2 \\ \text{s.t.} \quad 3x_1 + x_2 &\leq 12 \\ x_1 + x_2 &\leq 5 \\ x_1, x_2 \geq 0;\ &x_1, x_2 \text{ integer} \end{aligned}$$

2 The Dorian Auto example of Section 3.2.

3
$$\begin{aligned} \max z &= 2x_1 + 3x_2 \\ \text{s.t.} \quad x_1 + 2x_2 &\leq 10 \\ 3x_1 + 4x_2 &\leq 25 \\ x_1, x_2 \geq 0;\ &x_1, x_2 \text{ integer} \end{aligned}$$

4
$$\begin{aligned} \max z &= 4x_1 + 3x_2 \\ \text{s.t.} \quad 4x_1 + 9x_2 &\leq 26 \\ 8x_1 + 5x_2 &\leq 17 \\ x_1, x_2 \geq 0;\ &x_1, x_2 \text{ integer} \end{aligned}$$

5
$$\begin{aligned} \max z &= 4x_1 + 5x_2 \\ \text{s.t.} \quad x_1 + 4x_2 &\geq 5 \\ 3x_1 + 2x_2 &\geq 7 \\ x_1, x_2 \geq 0;\ &x_1, x_2 \text{ integer} \end{aligned}$$

6
$$\begin{aligned} \max z &= 4x_1 + 5x_2 \\ \text{s.t.} \quad 3x_1 + 2x_2 &\leq 10 \\ x_1 + 4x_2 &\leq 11 \\ 3x_1 + 3x_2 &\leq 13 \\ x_1, x_2 \geq 0;\ &x_1, x_2 \text{ integer} \end{aligned}$$

7 Use the branch-and-bound method to find the optimal solution to the following IP:

$$\begin{aligned} \max z &= 7x_1 + 3x_2 \\ \text{s.t.} \quad 2x_1 + x_2 &\leq 9 \\ 3x_1 + 2x_2 &\leq 13 \\ x_1, x_2 \geq 0;\ &x_1, x_2 \text{ integer} \end{aligned}$$

Group B

8 Suppose we have branched on a subproblem (call it subproblem 0, having optimal solution SOL0) and have obtained the following two subproblems:

Subproblem 1 Subproblem 0 + Constraint $x_1 \le i$.
Subproblem 2 Subproblem 0 + Constraint $x_1 \ge i + 1$ (i is some integer).

Prove that there will exist at least one optimal solution to subproblem 1 having $x_1 = i$ and at least one optimal solution to subproblem 2 having $x_1 = i + 1$. [*Hint:* Suppose an optimal solution to subproblem 1 (call it SOL1) has $x_1 = \bar{x}_1$, where $\bar{x}_1 < i$. For some number c ($0 < c < 1$), c(SOL0) + $(1 - c)$SOL1 will have the following three properties:

a The value of x_1 in c(SOL0) + $(1 - c)$SOL1 will equal i.

b c(SOL0) + $(1 - c)$SOL1 will be feasible in subproblem 1.

c The z-value for c(SOL0) + $(1 - c)$SOL1 will be at least as good as the z-value for SOL1.

Explain how this result can help when we graphically solve branch-and-bound problems.]

9 During the next five periods, the demands in Table 58 must be met on time. At the beginning of period 1, the inventory level is 0. Each period that production occurs a setup cost of \$250 and a per-unit production cost of \$2 are incurred. At the end of each period a per-unit holding cost of \$1 is incurred.

a Solve for the cost-minimizing production schedule using the following decision variables: x_t = units produced during month t and $y_t = 1$ if any units are produced during period t, $y_t = 0$ otherwise.

b Solve for the cost-minimizing production schedule using the following variables: y_t's defined in part (a) and x_{it} = number of units produced during period i to satisfy period t demand.

c Which formulation took LINDO or LINGO less time to solve?

d Give an intuitive explanation of why the part (b) formulation is solved faster than the part (a) formulation.

TABLE 58

	Period				
	1	2	3	4	5
Demand	220	280	360	140	270

9.4 The Branch-and-Bound Method for Solving Mixed Integer Programming Problems

Recall that, in a mixed IP, some variables are required to be integers and others are allowed to be either integers or nonintegers. To solve a mixed IP by the branch-and-bound method, modify the method described in Section 9.3 by branching only on variables that are required to be integers. Also, for a solution to a subproblem to be a candidate solution, it need only assign integer values to those variables that are required to be integers. To illustrate, let us solve the following mixed IP:

$$
\begin{aligned}
\max z &= 2x_1 + x_2 \\
\text{s.t.} \quad & 5x_1 + 2x_2 \le 8 \\
& x_1 + x_2 \le 3 \\
& x_1, x_2 \ge 0;\ x_1 \text{ integer}
\end{aligned}
$$

As before, we begin by solving the LP relaxation of the IP. The optimal solution of the LP relaxation is $z = \frac{11}{3}$, $x_1 = \frac{2}{3}$, $x_2 = \frac{7}{3}$. Because x_2 is allowed to be fractional, we do not branch on x_2; if we did so, we would be excluding points having x_2 values between 2 and 3, and we don't want to do that. Thus, we must branch on x_1. This yields subproblems 2 and 3 in Figure 21.

We next choose to solve subproblem 2. The optimal solution to subproblem 2 is the candidate solution $z = 3$, $x_1 = 0$, $x_2 = 3$. We now solve subproblem 3 and obtain the candidate solution $z = \frac{7}{2}$, $x_1 = 1$, $x_2 = \frac{3}{2}$. The z-value from the subproblem 3 candidate exceeds the z-value for the subproblem 2 candidate, so subproblem 2 can be eliminated from consideration, and the subproblem 3 candidate ($z = \frac{7}{2}$, $x_1 = 1$, $x_2 = \frac{3}{2}$) is the optimal solution to the mixed IP.

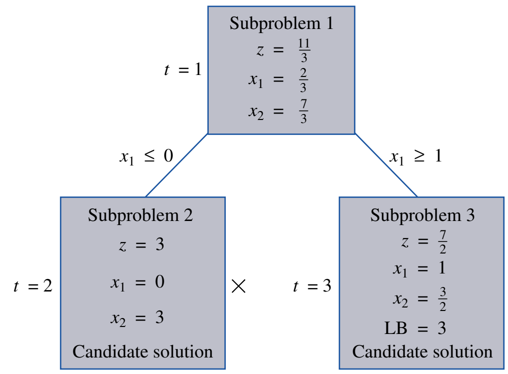


FIGURE 21
Branch-and-Bound Tree for Mixed IP

PROBLEMS

Group A

Use the branch-and-bound method to solve the following IPs:

1

$$\begin{aligned} \max z &= 3x_1 + x_2 \\ \text{s.t. } \quad 5x_1 + 2x_2 &\le 10 \\ 4x_1 + x_2 &\le 7 \\ x_1, x_2 \ge 0; &\ x_2 \text{ integer} \end{aligned}$$

2

$$\begin{aligned} \min z &= 3x_1 + x_2 \\ \text{s.t. } \quad x_1 + 5x_2 &\ge 8 \\ x_1 + 2x_2 &\ge 4 \\ x_1, x_2 \ge 0; &\ x_1 \text{ integer} \end{aligned}$$

3

$$\begin{aligned} \max z &= 4x_1 + 3x_2 + x_3 \\ \text{s.t. } \quad 3x_1 + 2x_2 + x_3 &\le 7 \\ 2x_1 + x_2 + 2x_3 &\le 11 \\ x_2, x_3 \text{ integer}, &\ x_1, x_2, x_3 \ge 0 \end{aligned}$$

9.5 Solving Knapsack Problems by the Branch-and-Bound Method

In Section 9.2, we learned that a knapsack problem is an IP with a single constraint. In this section, we discuss knapsack problems in which each variable must equal 0 or 1 (see Problem 1 at the end of this section for an explanation of how any knapsack problem can be reformulated so that each variable must equal 0 or 1). A knapsack problem in which each variable must equal 0 or 1 may be written as

$$\begin{aligned} \max z &= c_1x_1 + c_2x_2 + \cdots + c_nx_n \\ \text{s.t. } \quad a_1x_1 + a_2x_2 &+ \cdots + a_nx_n \le b \\ x_i = 0 \text{ or } 1 &\quad (i = 1, 2, \ldots, n) \end{aligned} \tag{38}$$

Recall that c_i is the benefit obtained if item i is chosen, b is the amount of an available resource, and a_i is the amount of the available resource used by item i.

When knapsack problems are solved by the branch-and-bound method, two aspects of the method greatly simplify. Because each variable must equal 0 or 1, branching on x_i will yield an $x_i = 0$ and an $x_i = 1$ branch. Also, the LP relaxation (and other subproblems) may be solved by inspection. To see this, observe that $\frac{c_i}{a_i}$ may be interpreted as the benefit item i earns for each unit of the resource used by item i. Thus, the best items have the largest values of $\frac{c_i}{a_i}$, and the worst items have the smallest values of $\frac{c_i}{a_i}$. To solve any subproblem resulting from a knapsack problem, compute all the ratios $\frac{c_i}{a_i}$. Then put the best item in the knapsack. Then put the second-best item in the knapsack. Continue in this fashion until the best remaining item will overfill the knapsack. Then fill the knapsack with as much of this item as possible.

To illustrate, we solve the LP relaxation of

$$\begin{aligned} \max z &= 40x_1 + 80x_2 + 10x_3 + 10x_4 + 4x_5 + 20x_6 + 60x_7 \\ \text{s.t.} \quad & 40x_1 + 50x_2 + 30x_3 + 10x_4 + 10x_5 + 40x_6 + 30x_7 \le 100 \\ & x_i = 0 \text{ or } 1 \quad (i = 1, 2, \ldots, 7) \end{aligned} \tag{39}$$

We begin by computing the $\frac{c_i}{a_i}$ ratios and ordering the variables from best to worst (see Table 59). To solve the LP relaxation of (39), we first choose item 7 ($x_7 = 1$). Then $100 - 30 = 70$ units of the resource remain. Now we include the second-best item (item 2) in the knapsack by setting $x_2 = 1$. Now $70 - 50 = 20$ units of the resource remain. Item 4 and item 1 have the same $\frac{c_i}{a_i}$ ratio, so we can next choose either of these items. We arbitrarily choose to set $x_4 = 1$. Then $20 - 10 = 10$ units of the resource remain. The best remaining item is item 1. We now fill the knapsack with as much of item 1 as we can. Because only 10 units of the resource remain, we set $x_1 = \frac{10}{40} = \frac{1}{4}$. Thus an optimal solution to the LP relaxation of (39) is $z = 80 + 60 + 10 + (\frac{1}{4})(40) = 160$, $x_2 = x_7 = x_4 = 1$, $x_1 = \frac{1}{4}$, $x_3 = x_5 = x_6 = 0$.

To show how the branch-and-bound method can be used to solve a knapsack problem, let us find the optimal solution to the Stockco capital budgeting problem (Example 1). Recall that this problem was

$$\begin{aligned} \max z &= 16x_1 + 22x_2 + 12x_3 + 8x_4 \\ \text{s.t.} \quad & 5x_1 + 7x_2 + 4x_3 + 3x_4 \le 14 \\ & x_j = 0 \text{ or } 1 \end{aligned}$$

The branch-and-bound tree for this problem is shown in Figure 22. From the tree, we find that the optimal solution to Example 1 is $z = 42$, $x_1 = 0$, $x_2 = x_3 = x_4 = 1$. Thus, we should invest in investments 2, 3, and 4 and earn an NPV of \$42,000. As discussed in Section 9.2, the "best" investment is not used.

REMARKS The method we used in traversing the tree of Figure 22 is as follows:

1 We used the LIFO approach to determine which subproblem should be solved.

2 We arbitrarily chose to solve subproblem 3 before subproblem 2. To solve subproblem 3, we first set $x_3 = 1$ and then solved the resulting knapsack problem. After setting $x_3 = 1$, $14 - 4 = \$10$ million was still available for investment. Applying the technique used to solve the LP relaxation of a knapsack problem yielded the following optimal solution to subproblem 3: $x_3 = 1$, $x_1 = 1$, $x_2 = \frac{5}{7}$, $x_4 = 0$, $z = 16 + (\frac{5}{7})(22) + 12 = \frac{306}{7}$. Other subproblems were solved similarly; of course, if a subproblem specified $x_i = 0$, the optimal solution to that subproblem could not use investment i.

TABLE 59

Ordering Items from Best to Worst in a Knapsack Problem

Item	$\frac{c_i}{a_i}$	Ranking (1 = best, 7 = worst)
1	1	3.5 (tie for third or fourth)
2	$\frac{8}{5}$	2
3	$\frac{1}{3}$	7
4	1	3.5
5	$\frac{4}{10}$	6
6	$\frac{1}{2}$	5
7	2	1

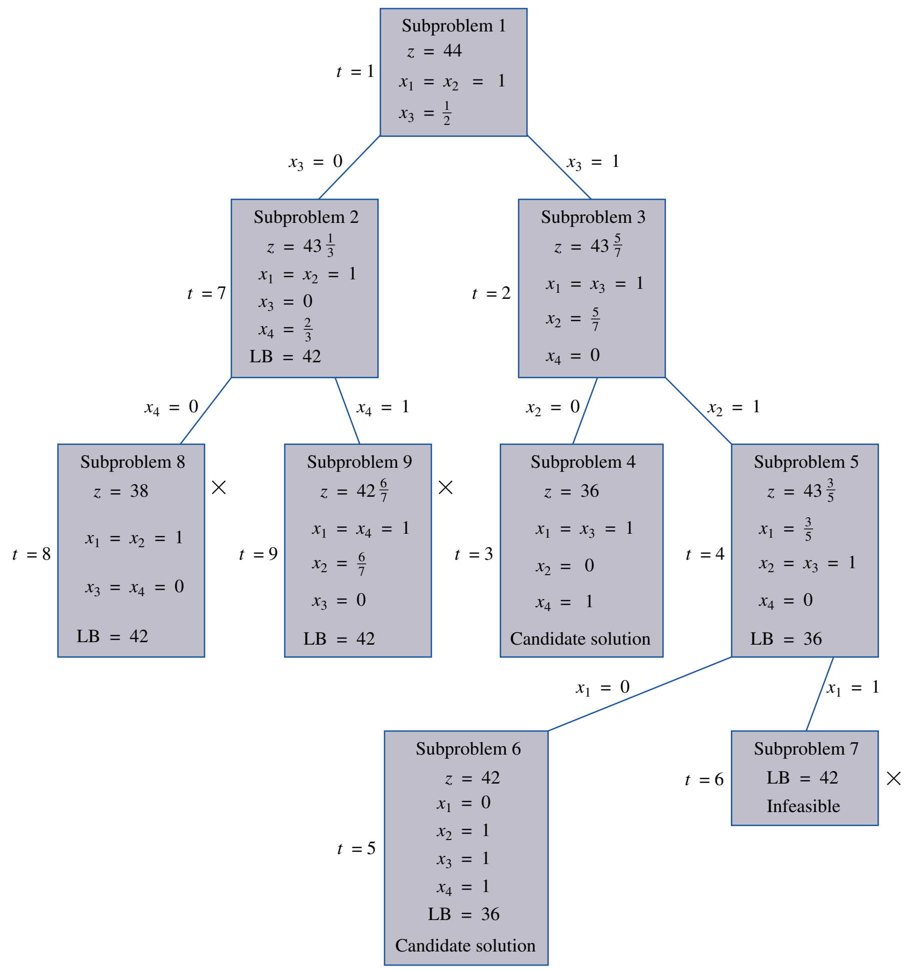


FIGURE 22
Branch-and-Bound Tree for Stockco Knapsack Problem

3 Subproblem 4 yielded the candidate solution $x_1 = x_3 = x_4 = 1$, $z = 36$. We then set LB $= 36$.
4 Subproblem 6 yielded a candidate solution with $z = 42$. Thus, subproblem 4 was eliminated from consideration, and the LB was updated to 42.
5 Subproblem 7 was infeasible because it required $x_1 = x_2 = x_3 = 1$, and such a solution requires at least \$16 million.
6 Subproblem 8 was eliminated because its z-value ($z = 38$) did not exceed the current LB of 42.
7 Subproblem 9 had a z-value of $42\frac{6}{7}$. Because the z-value for any all-integer solution must also be an integer, this meant that branching on subproblem 9 could never yield a z-value larger than 42. Thus, further branching on subproblem 9 could not beat the current LB of 42, and subproblem 9 was eliminated from consideration.

In Chapter 13, we show how dynamic programming can be used to solve knapsack problems.

PROBLEMS

Group A

1 Show how the following problem can be expressed as a knapsack problem in which all variables must equal 0 or 1. NASA is determining how many of three types of objects should be brought on board the space shuttle. The weight and benefit of each of the items are given in Table 60. If the space shuttle can carry a maximum of 26 lb of items 1–3, which items should be taken on the space shuttle?

TABLE 60

Item	Benefit	Weight (Pounds)
1	10	3
2	15	4
3	17	5

2 I am moving from New Jersey to Indiana and have rented a truck that can haul up to 1,100 cu ft of furniture. The volume and value of each item I am considering moving on the truck are given in Table 61. Which items should I bring to Indiana? To solve this problem as a knapsack problem, what unrealistic assumptions must we make?

3 Four projects are available for investment. The projects require the cash flows and yield the net present values (NPV) (in millions) shown in Table 62. If $6 million is available for investment at time 0, find the investment plan that maximizes NPV.

TABLE 61

Item	Value ($)	Volume (Cubic Feet)
Bedroom set	60	800
Dining room set	48	600
Stereo	14	300
Sofa	31	400
TV set	10	200

TABLE 62

Project	Cash Outflow at Time 0 ($)	NPV ($)
1	3	5
2	5	8
3	2	3
4	4	7

9.6 Solving Combinatorial Optimization Problems by the Branch-and-Bound Method

Loosely speaking, a **combinatorial optimization problem** is any optimization problem that has a finite number of feasible solutions. A branch-and-bound approach is often the most efficient way to solve them. Three examples of combinatorial optimization problems follow:

1 Ten jobs must be processed on a single machine. You know the time it takes to complete each job and the time at which each job must be completed (the job's due date). What ordering of the jobs minimizes the total delay of the 10 jobs?

2 A salesperson must visit each of 10 cities once before returning to his home. What ordering of the cities minimizes the total distance the salesperson must travel before returning home? Not surprisingly, this problem is called the *traveling salesperson problem* (TSP).

3 Determine how to place eight queens on a chessboard so that no queen can capture any other queen (see Problem 7 at the end of this section).

In each of these problems, many possible solutions must be considered. For instance, in Problem 1, the first job to be processed can be one of 10 jobs, the next job can be one of 9 jobs, and so on. Thus, even for this relatively small problem there are $10(9)(8) \cdots (1) = 10! = 3{,}628{,}000$ possible ways to schedule the jobs. A combinatorial optimization problem may have many feasible solutions, so it can require a great deal of computer time to enumerate all possible solutions explicitly. For this reason, branch-and-bound methods are often used for *implicit* enumeration of all possible solutions to a combinatorial optimization problem. As we will see, the branch-and-bound method should take advantage of the structure of the particular problem that is being solved.

To illustrate how branch-and-bound methods are used to solve combinatorial optimization problems, we show how the approach can be used to solve Problems 1 and 2 of the preceding list.

Branch-and-Bound Approach for Machine-Scheduling Problem

Example 10 illustrates how a branch-and-bound approach may be used to schedule jobs on a single machine. See Baker (1974) and Hax and Candea (1984) for a discussion of other branch-and-bound approaches to machine-scheduling problems.

EXAMPLE 10 Branch-and-Bound Machine Scheduling

Four jobs must be processed on a single machine. The time required to process each job and the date the job is due are shown in Table 63. The delay of a job is the number of days after the due date that a job is completed (if a job is completed on time or early, the job's delay is zero). In what order should the jobs be processed to minimize the total delay of the four jobs?

Solution Suppose the jobs are processed in the following order: job 1, job 2, job 3, and job 4. Then the delays shown in Table 64 would occur. For this sequence, total delay $= 0 + 6 + 3 + 7 = 16$ days. We now describe a branch-and-bound approach for solving this type of machine-scheduling problem.

Because a possible solution to the problem must specify the order in which the jobs are processed, we define

$$x_{ij} = \begin{cases} 1 & \text{if job } i \text{ is the } j\text{th job to be processed} \\ 0 & \text{otherwise} \end{cases}$$

The branch-and-bound approach begins by partitioning all solutions according to the job that is *last* processed. Any sequence of jobs must process some job last, so each sequence of jobs must have $x_{14} = 1$, $x_{24} = 1$, $x_{34} = 1$, or $x_{44} = 1$. This yields four branches with nodes 1–4 in Figure 23. After we create a node by branching, we obtain a lower bound on the total delay (D) associated with the node. For example, if $x_{44} = 1$, we know that job 4 is the last job to be processed. In this case, job 4 will be completed at the end of day $6 + 4 + 5 + 8 = 23$ and will be $23 - 16 = 7$ days late. Thus, any schedule having

TABLE 63
Durations and Due Date of Jobs

Job	Days Required to Complete Job	Due Date
1	6	End of day 8
2	4	End of day 4
3	5	End of day 12
4	8	End of day 16

TABLE 64
Delays Incurred If Jobs Are Processed in the Order 1–2–3–4

Job	Completion Time of Job	Delay of Job
1	6	0
2	$6 + 4 = 10$	$10 - 4 = 6$
3	$6 + 4 + 5 = 15$	$15 - 12 = 3$
4	$6 + 4 + 5 + 8 = 23$	$23 - 16 = 7$

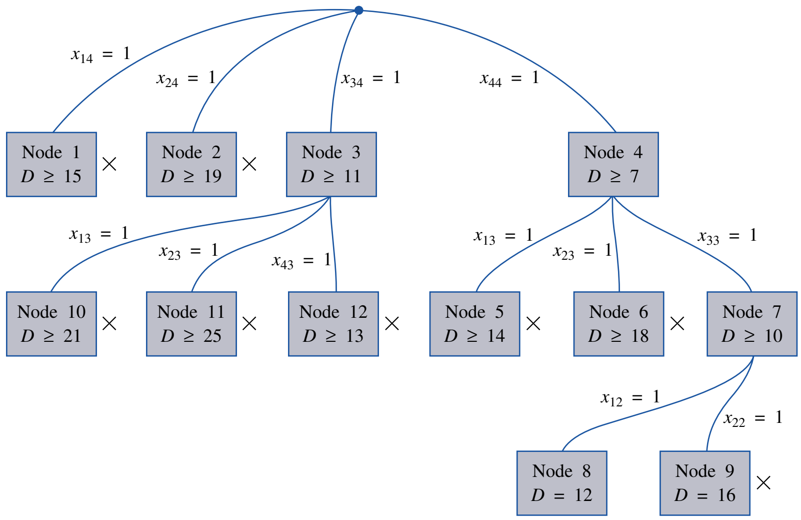


FIGURE 23
Branch-and-Bound Tree for Machine-Scheduling Problem

$x_{44} = 1$ must have $D \geq 7$. Thus, we write $D \geq 7$ inside node 4 of Figure 23. Similar reasoning shows that any sequence of jobs having $x_{34} = 1$ will have $D \geq 11$, $x_{24} = 1$ will have $D \geq 19$, and $x_{14} = 1$ will have $D \geq 15$. We have no reason to exclude any of nodes 1–4 from consideration as part of the optimal job sequence, so we choose to branch on a node. We use the jumptracking approach and branch on the node that has the smallest bound on D: node 4. Any job sequence associated with node 4 must have $x_{13} = 1$, $x_{23} = 1$, or $x_{33} = 1$. Branching on node 4 yields nodes 5–7 in Figure 23. For each new node, we need a lower bound for the total delay. For example, at node 7, we know from our analysis of node 1 that job 4 will be processed last and will be delayed by 7 days. For node 7, we know that job 3 will be the third job processed. Thus, job 3 will be completed after $6 + 4 + 5 = 15$ days and will be $15 - 12 = 3$ days late. Any sequence associated with node 7 must have $D \geq 7 + 3 = 10$ days. Similar reasoning shows that node 5 must have $D \geq 14$, and node 6 must have $D \geq 18$. We still do not have any reason to eliminate any of nodes 1–7 from consideration, so we again branch on a node. The jumptracking approach directs us to branch on node 7. Any job sequence associated with node 7 must have either job 1 or job 2 as the second job processed. Thus, any job sequence associated with node 7 must have $x_{12} = 1$ or $x_{22} = 1$. Branching on node 7 yields nodes 8 and 9 in Figure 23.

Node 9 corresponds to processing the jobs in the order 1–2–3–4. This sequence yields a total delay of 7 (for job 4) + 3 (for job 3) + (6 + 4 − 4) (for job 2) + 0 (for job 1) = 16 days. Node 9 is a feasible sequence and may be considered a candidate solution having $D = 16$. We now know that any node that cannot have a total delay of less than 16 days can be eliminated.

Node 8 corresponds to the sequence 2–1–3–4. This sequence has a total delay of 7 (for job 4) + 3 (for job 3) + (4 + 6 − 8) (for job 1) + 0 (for job 2) = 12 days. Node 8 is a feasible sequence and may be viewed as a candidate solution with $D = 12$. Because node 8 is better than node 9, node 9 may be eliminated from consideration.

Similarly, node 5 (having $D \geq 14$), node 6 (having $D \geq 18$), node 1 (having $D \geq 15$), and node 2 (having $D \geq 19$) can be eliminated. Node 3 cannot yet be eliminated, because it is still possible for node 3 to yield a sequence having $D = 11$. Thus, we now branch on node 3. Any job sequence associated with node 3 must have $x_{13} = 1$, $x_{23} = 1$, or $x_{43} = 1$, so we obtain nodes 10–12.

For node 10, $D \geq$ (delay from processing job 3 last) + (delay from processing job 1 third) $= 11 + (6 + 4 + 8 - 8) = 21$. Because any sequence associated with node 10 must have $D \geq 21$ and we have a candidate with $D = 12$, node 10 may be eliminated.

For node 11, $D \geq$ (delay from processing job 3 last) + (delay from processing job 2 third) $= 11 + (6 + 4 + 8 - 4) = 25$. Any sequence associated with node 11 must have $D \geq 25$, and node 11 may be eliminated.

Finally, for node 12, $D \geq$ (delay from processing job 3 last) + (delay from processing job 4 third) $= 11 + (6 + 4 + 8 - 16) = 13$. Any sequence associated with node 12 must have $D \geq 13$, and node 12 may be eliminated.

With the exception of node 8, every node in Figure 23 has been eliminated from consideration. Node 8 yields the delay-minimizing sequence $x_{44} = x_{33} = x_{12} = x_{21} = 1$. Thus, the jobs should be processed in the order 2–1–3–4, with a total delay of 12 days resulting.

Branch-and-Bound Approach for Traveling Salesperson Problem

EXAMPLE 11 Traveling Salesperson Problem

Joe State lives in Gary, Indiana. He owns insurance agencies in Gary, Fort Wayne, Evansville, Terre Haute, and South Bend. Each December, he visits each of his insurance agencies. The distance between each agency (in miles) is shown in Table 65. What order of visiting his agencies will minimize the total distance traveled?

Solution Joe must determine the order of visiting the five cities that minimizes the total distance traveled. For example, Joe could choose to visit the cities in the order 1–3–4–5–2–1. Then he would travel a total of $217 + 113 + 196 + 79 + 132 = 737$ miles.

To tackle the traveling salesperson problem, define

$$x_{ij} = \begin{cases} 1 & \text{if Joe leaves city } i \text{ and travels next to city } j \\ 0 & \text{otherwise} \end{cases}$$

Also, for $i \neq j$,

$$c_{ij} = \text{distance between cities } i \text{ and } j$$
$$c_{ii} = M, \text{ where } M \text{ is a large positive number}$$

It seems reasonable that we might be able to find the answer to Joe's problem by solving an assignment problem having a cost matrix whose ijth element is c_{ij}. For instance, suppose we solved this assignment problem and obtained the solution $x_{12} = x_{24} = x_{45} = x_{53} = x_{31} = 1$. Then Joe should go from Gary to Fort Wayne, from Fort Wayne to Terre Haute, from Terre Haute to South Bend, from South Bend to Evansville, and from Evansville to Gary. This solution can be written as 1–2–4–5–3–1. An itinerary that begins and ends at the same city and visits each city once is called a **tour.**

TABLE 65

Distance between Cities in Traveling Salesperson Problem

Day	Gary	Fort Wayne	Evansville	Terre Haute	South Bend
City 1 Gary	0	132	217	164	58
City 2 Fort Wayne	132	0	290	201	79
City 3 Evansville	217	290	0	113	303
City 4 Terre Haute	164	201	113	0	196
City 5 South Bend	58	79	303	196	0

If the solution to the preceding assignment problem yields a tour, then it is the optimal solution to the traveling salesperson problem. (Why?) Unfortunately, the optimal solution to the assignment problem need not be a tour. For example, the optimal solution to the assignment problem might be $x_{15} = x_{21} = x_{34} = x_{43} = x_{52} = 1$. This solution suggests going from Gary to South Bend, then to Fort Wayne, and then back to Gary. This solution also suggests that if Joe is in Evansville he should go to Terre Haute and then to Evansville (see Figure 24). Of course, if Joe begins in Gary, this solution will never get him to Evansville or Terre Haute. This is because the optimal solution to the assignment problem contains two **subtours.** A subtour is a round trip that does not pass through all cities. The current assignment contains the two subtours 1–5–2–1 and 3–4–3. If we could exclude all feasible solutions that contain subtours and then solve the assignment problem, we would obtain the optimal solution to the traveling salesperson problem. This is not easy to do, however. In most cases, a branch-and-bound approach is the most efficient approach for solving a TSP.

Several branch-and-bound approaches have been developed for solving TSPs [see Wagner (1975)]. We describe an approach here in which the subproblems reduce to assignment problems. To begin, we solve the preceding assignment problem, in which, for $i \neq j$, the cost c_{ij} is the distance between cities i and j and $c_{ii} = M$ (this prevents a person in a city from being assigned to visit that city itself). Because this assignment problem contains no provisions to prevent subtours, it is a relaxation (or less constrained problem) of the original traveling salesperson problem. Thus, if the optimal solution to the assignment problem is feasible for the traveling salesperson problem (that is, if the assignment solution contains no subtours), then it is also optimal for the traveling salesperson problem. The results of the branch-and-bound procedure are given in Figure 25.

We first solve the assignment problem in Table 66 (referred to as subproblem 1). The optimal solution is $x_{15} = x_{21} = x_{34} = x_{43} = x_{52} = 1$, $z = 495$. This solution contains two subtours (1–5–2–1 and 3–4–3) and cannot be the optimal solution to Joe's problem.

We now branch on subproblem 1 in a way that will prevent one of subproblem 1's subtours from recurring in solutions to subsequent subproblems. We choose to exclude the subtour 3–4–3. Observe that the optimal solution to Joe's problem must have either $x_{34} = 0$ or $x_{43} = 0$ (if $x_{34} = x_{43} = 1$, the optimal solution would have the subtour 3–4–3). Thus, we can branch on subproblem 1 by adding the following two subproblems:

Subproblem 2 Subproblem 1 + ($x_{34} = 0$, or $c_{34} = M$).

Subproblem 3 Subproblem 1 + ($x_{43} = 0$, or $c_{43} = M$).

We now arbitrarily choose subproblem 2 to solve, applying the Hungarian method to the cost matrix as shown in Table 67. The optimal solution to subproblem 2 is $z = 652$, $x_{14} = x_{25} = x_{31} = x_{43} = x_{52} = 1$. This solution includes the subtours 1–4–3–1 and 2–5–2, so this cannot be the optimal solution to Joe's problem.

We now branch on subproblem 2 in an effort to exclude the subtour 2–5–2. We must ensure that either x_{25} or x_{52} equals zero. Thus, we add the following two subproblems:

Subproblem 4 Subproblem 2 + ($x_{25} = 0$, or $c_{25} = M$).

Subproblem 5 Subproblem 2 + ($x_{52} = 0$, or $c_{52} = M$).

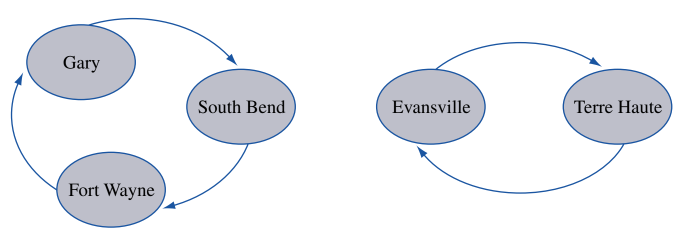


FIGURE 24
Example of Subtours in Traveling Salesperson Problem

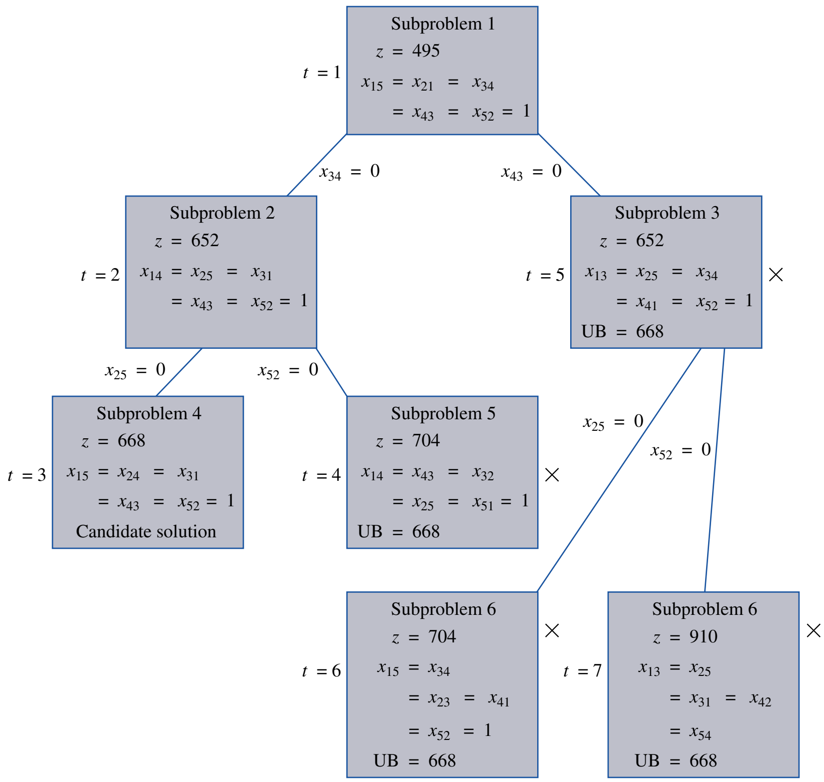


FIGURE 25
Branch-and-Bound Tree for Traveling Salesperson Problem

TABLE 66
Cost Matrix for Subproblem 1

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	79
City 3	217	290	M	113	303
City 4	164	201	113	M	196
City 5	58	79	303	196	M

TABLE 67
Cost Matrix for Subproblem 2

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	79
City 3	217	290	M	M	303
City 4	164	201	113	M	196
City 5	58	79	303	196	M

Following the LIFO approach, we should next solve subproblem 4 or subproblem 5. We arbitrarily choose to solve subproblem 4. Applying the Hungarian method to the cost matrix shown in Table 68, we obtain the optimal solution $z = 668$, $x_{15} = x_{24} = x_{31} = x_{43} = x_{52} = 1$. This solution contains no subtours and yields the tour 1–5–2–4–3–1. Thus, subproblem 4 yields a candidate solution with $z = 668$. Any node that cannot yield a z-value < 668 may be eliminated from consideration.

Following the LIFO rule, we next solve subproblem 5, applying the Hungarian method to the matrix in Table 69. The optimal solution to subproblem 5 is $z = 704$, $x_{14} = x_{43} = x_{32} = x_{25} = x_{51} = 1$. This solution is a tour, but $z = 704$ is not as good as the subproblem 4 candidate's $z = 668$. Thus, subproblem 5 may be eliminated from consideration.

Only subproblem 3 remains. We find the optimal solution to the assignment problem in Table 70, $x_{13} = x_{25} = x_{34} = x_{41} = x_{52} = 1$, $z = 652$. This solution contains the subtours 1–3–4–1 and 2–5–2. Because $652 < 668$, however, it is still possible for subproblem 3 to yield a solution with no subtours that beats $z = 668$. Thus, we now branch on subproblem 3 in an effort to exclude the subtours. Any feasible solution to the traveling salesperson problem that emanates from subproblem 3 must have either $x_{25} = 0$ or $x_{52} = 0$ (why?), so we create subproblems 6 and 7.

Subproblem 6 Subproblem 3 + ($x_{25} = 0$, or $c_{25} = M$).

Subproblem 7 Subproblem 3 + ($x_{52} = 0$, or $c_{52} = M$).

TABLE 68
Cost Matrix for Subproblem 4

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	M
City 3	217	290	M	M	303
City 4	164	201	113	M	196
City 5	58	79	303	196	M

TABLE 69
Cost Matrix for Subproblem 5

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	79
City 3	217	290	M	M	303
City 4	164	201	113	M	196
City 5	58	M	303	196	M

TABLE 70
Cost Matrix for Subproblem 3

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	79
City 3	217	290	M	113	303
City 4	164	201	M	M	196
City 5	58	79	303	196	M

We next choose to solve subproblem 6. The optimal solution to subproblem 6 is $x_{15} = x_{34} = x_{23} = x_{41} = x_{52} = 1$, $z = 704$. This solution contains no subtours, but its z-value of 704 is inferior to the candidate solution from subproblem 4, so subproblem 6 cannot yield the optimal solution to the problem.

The only remaining subproblem is subproblem 7. The optimal solution to subproblem 7 is $x_{13} = x_{25} = x_{31} = x_{42} = x_{54} = 1$, $z = 910$. Again, $z = 910$ is inferior to $z = 668$, so subproblem 7 cannot yield the optimal solution.

Subproblem 4 thus yields the optimal solution: Joe should travel from Gary to South Bend, from South Bend to Fort Wayne, from Fort Wayne to Terre Haute, from Terre Haute to Evansville, and from Evansville to Gary. Joe will travel a total distance of 668 miles.

Heuristics for TSPs

When using branch-and-bound methods to solve TSPs with many cities, large amounts of computer time may be required. For this reason, **heuristic methods,** or **heuristics,** which quickly lead to a good (but not necessarily optimal) solution to a TSP, are often used. A heuristic is a method used to solve a problem by trial and error when an algorithmic approach is impractical. Heuristics often have an intuitive justification. We now discuss two heuristics for the TSP: the nearest-neighbor and the cheapest-insertion heuristics.

To apply the nearest-neighbor heuristic (NNH), we begin at any city and then "visit" the nearest city. Then we go to the unvisited city closest to the city we have most recently visited. Continue in this fashion until a tour is obtained. We now apply the NNH to Example 11. We arbitrarily choose to begin at city 1. City 5 is the closest city to city 1, so we have now generated the arc 1–5. Of cities 2, 3, and 4, city 2 is closest to city 5, so we have now generated the arcs 1–5–2. Of cities 3 and 4, city 4 is closest to city 2. We now have generated the arcs 1–5–2–4. Of course, we must next visit city 3 and then return to city 1; this yields the tour 1–5–2–4–3–1. In this case, the NNH yields an optimal tour. If we had begun at city 3, however, the reader should verify that the tour 3–4–1–5–2–3 would be obtained. This tour has length $113 + 164 + 58 + 79 + 290 = 704$ miles and is not optimal. Thus, the NNH need not yield an optimal tour. A popular heuristic is to apply the NNH beginning at each city and then take the best tour obtained.

In the cheapest-insertion heuristic (CIH), we begin at any city and find its closest neighbor. Then we create a subtour joining those two cities. Next, we replace an arc in the subtour [say, arc (i, j)] by the combination of two arcs—(i, k) and (k, j), where k is not in the current subtour—that will increase the length of the subtour by the smallest (or cheapest) amount. Let c_{ij} be the length of arc (i, j). Note that if arc (i, j) is replaced by arcs (i, k) and (k, j), then a length $c_{ik} + c_{kj} - c_{ij}$ is added to the subtour. Then we continue with this procedure until a tour is obtained. Suppose we begin the CIH at city 1. City 5 is closest to city 1, so we begin with the subtour (1, 5)–(5, 1). Then we could replace (1, 5) by (1, 2)–(2, 5), (1, 3)–(3, 5), or (1, 4)–(4, 5). We could also replace arc (5, 1) by (5, 2)–(2, 1), (5, 3)–(3, 1), or (5, 4)–(4, 1). The calculations used to determine which arc of (1,5)–(5,1) should be replaced are given in Table 71 (* indicates the correct replacement). As seen in the table, we may replace either (1, 5) or (5, 1). We arbitrarily choose to replace arc (1, 5) by arcs (1, 2) and (2, 5). We currently have the subtour (1, 2)–(2, 5)–(5, 1). We must now replace an arc (i, j) of this subtour by the arcs (i, k) and (k, j), where $k = 3$ or 4. The relevant computations are shown in Table 72.

We now replace (1, 2) by arcs (1, 4) and (4, 2). This yields the subtour (1, 4)–(4, 2)–(2, 5)–(5, 1). An arc (i, j) in this subtour must now be replaced by arcs $(i, 3)$ and $(3, j)$. The relevant computations are shown in Table 73. We now replace arc (1, 4) by arcs (1, 3) and (3, 4). This yields the tour (1, 3)–(3, 4)–(4, 2)–(2, 5)–(5, 1). In this example, the CIH yields an optimal tour—but, in general, the CIH does not necessarily do so.

TABLE 71
Determining Which Arc of (1, 5)–(5, 1) Is Replaced

Arc Replaced	Arcs Added to Subtour	Added Length
(1, 5)*	(1, 2)–(2, 5)	$c_{12} + c_{25} - c_{15} = 153$
(1, 5)	(1, 3)–(3, 5)	$c_{13} + c_{35} - c_{15} = 462$
(1, 5)	(1, 4)–(4, 5)	$c_{14} + c_{45} - c_{15} = 302$
(5, 1)*	(5, 2)–(2, 1)	$c_{52} + c_{21} - c_{51} = 153$
(5, 1)	(5, 3)–(3, 1)	$c_{53} + c_{31} - c_{51} = 462$
(5, 1)	(5, 4)–(4, 1)	$c_{54} + c_{41} - c_{51} = 302$

TABLE 72
Determining Which Arc of (1, 2)–(2, 5)–(5, 1) Is Replaced

Arc Replaced	Arcs Added	Added Length
(1, 2)	(1, 3)–(3, 2)	$c_{13} + c_{32} - c_{12} = 375$
(1, 2)*	(1, 4)–(4, 2)	$c_{14} + c_{42} - c_{12} = 233$
(2, 5)	(2, 3)–(3, 5)	$c_{23} + c_{35} - c_{25} = 514$
(2, 5)	(2, 4)–(4, 5)	$c_{24} + c_{45} - c_{25} = 318$
(5, 1)	(5, 3)–(3, 1)	$c_{53} + c_{31} - c_{51} = 462$
(5, 1)	(5, 4)–(4, 1)	$c_{54} + c_{41} - c_{51} = 302$

TABLE 73
Determining Which Arc of (1, 4)–(4, 2)–(2, 5)–(5, 1) Is Replaced

Arc Replaced	Arcs Added	Added Length
(1, 4)*	(1, 3)–(3, 4)	$c_{13} + c_{34} - c_{14} = 166$
(4, 2)	(4, 3)–(3, 2)	$c_{43} + c_{32} - c_{42} = 202$
(2, 5)	(2, 3)–(3, 5)	$c_{23} + c_{35} - c_{25} = 514$
(5, 1)	(5, 3)–(3, 1)	$c_{53} + c_{31} - c_{51} = 462$

Evaluation of Heuristics

The following three methods have been suggested for evaluating heuristics:

1 Performance guarantees

2 Probabilistic analysis

3 Empirical analysis

A performance guarantee for a heuristic gives a worst-case bound on how far away from optimality a tour constructed by the heuristic can be. For the NNH, it can be shown that for any number r, a TSP can be constructed such that the NNH yields a tour that is r times as long as the optimal tour. Thus, in a worst-case scenario, the NNH fares poorly. For a symmetric TSP satisfying the triangle inequality (that is, for which $c_{ij} = c_{ji}$ and $c_{ik} \leq c_{ij} + c_{jk}$ for all i, j, and k), it has been shown that the length of the tour obtained by the CIH cannot exceed twice the length of the optimal tour.

In probabilistic analysis, a heuristic is evaluated by assuming that the location of cities follows some known probability distribution. For example, we might assume that the cities are independent random variables that are uniformly distributed on a cube of unit length, width, and height. Then, for each heuristic, we would compute the following ratio:

$$\frac{\text{Expected length of the path found by the heuristic}}{}$$

The closer the ratio is to 1, the better the heuristic.

For empirical analysis, heuristics are compared to the optimal solution for a number of problems for which the optimal tour is known. As an illustration, for five 100-city TSPs, Golden, Bodin, Doyle, and Stewart (1980) found that the NNH—taking the best of all solutions found when the NNH was applied beginning at each city—produced tours that averaged 15% longer than the optimal tour. For the same set of problems, it was found that the CIH (again applying the best solution obtained by applying CIH to all cities) produced tours that also averaged 15% longer than the optimal tour.

REMARKS

1 Golden, Bodin, Doyle, and Stewart (1980) describe a heuristic that regularly comes within 2–3% of the optimal tour.

2 It is also important to compare heuristics with regard to computer running time and ease of implementation.

3 For an excellent discussion of heuristics, see Chapters 5–7 of Lawler (1985).

An Integer Programming Formulation of the TSP

We now discuss how to formulate an IP whose solution will solve a TSP. We note, however, that the formulation of this section becomes unwieldy and inefficient for large TSPs. Suppose the TSP consists of cities 1, 2, 3, . . . , N. For $i \neq j$ let c_{ij} = distance from city i to city j and let $c_{ii} = M$, where M is a very large number (relative to the actual distances in the problem). Setting $c_{ii} = M$ ensures that we will not go to city i immediately after leaving city i. Also define

$$x_{ij} = \begin{cases} 1 & \text{if the solution to TSP goes from city } i \text{ to city } j \\ 0 & \text{otherwise} \end{cases}$$

Then the solution to a TSP can be found by solving

$$\min z = \sum_i \sum_j c_{ij} x_{ij} \tag{40}$$

$$\text{s.t.} \quad \sum_{i=1}^{i=N} x_{ij} = 1 \quad (\text{for } j = 1, 2, \ldots, N) \tag{41}$$

$$\sum_{j=1}^{j=N} x_{ij} = 1 \quad (\text{for } i = 1, 2, \ldots, N) \tag{42}$$

$$u_i - u_j + N x_{ij} \leq N - 1 \quad (\text{for } i \neq j;\ i = 2, 3, \ldots, N;\ j = 2, 3, \ldots, N) \tag{43}$$

$$\text{All } x_{ij} = 0 \text{ or } 1, \text{ All } u_j \geq 0$$

The objective function (40) gives the total length of the arcs included in a tour. The constraints in (41) ensure that we arrive once at each city. The constraints in (42) ensure that we leave each city once. The constraints in (43) are the key to the formulation. They ensure the following:

1 Any set of x_{ij}'s containing a subtour will be infeasible [that is, they violate (43)].

2 Any set of x_{ij}'s that forms a tour will be feasible [there will exist a set of u_j's that satisfy (43)].

To illustrate that any set of x_{ij}'s containing a subtour will violate (43), consider the subtour illustration given in Figure 24. Here $x_{15} = x_{21} = x_{43} = x_{43} = x_{52} = 1$. This assignment contains the two subtours 1–5–2–1 and 3–4–3. Choose the subtour that does *not* contain city 1 (3–4–3) and write down the constraints in (43) corresponding to the arcs in this subtour. We obtain $u_3 - u_4 + 5x_{34} \leq 4$ and $u_4 - u_3 + 5x_{43} \leq 4$. Adding these constraints yields $5(x_{34} + x_{43}) \leq 8$. Clearly, this rules out the possibility that $x_{43} = x_{34} = 1$, so the subtour 3–4–3 (and any other subtour!) is ruled out by the constraints in (43).

We now show that for any set of x_{ij}'s that does not contain a subtour, there exist values of the u_j's that will satisfy all constraints in (43). Assume that city 1 is the first city visited (we visit all cities eventually, so this is okay). Let t_i = the position in the tour where city i is visited. Then setting $u_i = t_i$ will satisfy all constraints in (43). To illustrate, consider the tour 1–3–4–5–2–1. Then we choose $u_1 = 1$, $u_2 = 5$, $u_3 = 2$, $u_4 = 3$, $u_5 = 4$. We now show that with this choice of the u_i's all constraints in (43) are satisfied. First, consider any constraint corresponding to an arc having $x_{ij} = 1$. For example, the constraint corresponding to x_{52} is $u_5 - u_2 + 5x_{52} \leq 4$. Because city 2 immediately follows city 5, $u_5 - u_2 = -1$. Then the constraint for x_{52} in (43) reduces to $-1 + 5 \leq 4$, which is true. Now consider a constraint corresponding to an x_{ij} (say, x_{32}) satisfying $x_{ij} = 0$. For x_{32}, we obtain the constraint $u_3 - u_2 + 5x_{32} \leq 4$. This reduces to $u_3 - u_2 \leq 4$. Because $u_3 \leq 5$ and $u_2 > 1$, $u_3 - u_2$ cannot exceed $5 - 2$.

This shows that the formulation defined by (40)–(43) eliminates from consideration all sequences of N cities that begin in city 1 and include a subtour. We have also shown that this formulation does not eliminate from consideration any sequence of N cities beginning in city 1 that does not include a subtour. Thus, (40)–(43) will (if solved) yield the optimal solution to the TSP.

Using LINGO to Solve TSPs

TSP.lng

The IP described in (40)–(43) can easily be implemented with the following LINGO program (file TSP.lng).

```
MODEL:
  1]SETS:
  2]CITY/1..5/:U;
  3]LINK(CITY,CITY):DIST,X;
  4]ENDSETS
  5]DATA:
  6]DIST= 50000 132 217 164 58
  7]132 50000 290 201 79
  8]217 290 50000 113 303
  9]164 201 113 50000 196
 10]58 79 303 196 5000;
 11]ENDDATA
 12]N=@SIZE(CITY);
 13]MIN=@SUM(LINK:DIST*X);
 14]@FOR(CITY(K):@SUM(CITY(I):X(I,K))=1;);
 15]@FOR(CITY(K):@SUM(CITY(J):X(K,J))=1;);
 16]@FOR(CITY(K):@FOR(CITY(J)|J#GT#1#AND#K#GT#1:
 17]U(J)-U(K)+N*X(J,K)<N-1;));
 18]@FOR(LINK:@BIN(X););
END
```

In line 2, we define our five cities and associate a U(J) with city J. In line 3, we create the arcs joining each combination of cities. With the arc from city I to city J, we associate the distance between city I and J and a 0–1 variable X(I,J), which equals 1 if city J immediately follows city I in a tour.

In lines 6–10, we input the distance between the cities given in Example 11. Note that the distance between city I and itself is assigned a large number, to ensure that city I does not follow itself.

In line 12, we use **@SIZE** to compute the number of cities (we use this in line 17). In line 13, we create the objective function by summing over each link (I,J) the product of the distance between cities I and J and X(I,J). Line 14 ensures that for each city we enter the city exactly once. Line 15 ensures that for each city we leave the city exactly once. Lines 16–17 create the constraints in (43). Note that we only create these constraints for combinations J,K where J > 1 and K > 1. This agrees with (43). Note that when J = K line 17 generates constraints of the form N*X(J,J) ≤ N − 1, which imply that all X(J,J) = 0. In line 18, we ensure that each X(I,J) = 0 or 1. We need not constrain the U(J)'s, because LINGO assumes they are nonnegative. *Note:* Even for small TSPs, this formulation will exceed the capacity of student LINGO.

PROBLEMS

Group A

1 Four jobs must be processed on a single machine. The time required to perform each job and the due date for each job are shown in Table 74. Use the branch-and-bound method to determine the order of performing the jobs that minimizes the total time the jobs are delayed.

2 Each day, Sunco manufactures four types of gasoline: lead-free premium (LFP), lead-free regular (LFR), leaded premium (LP), and leaded regular (LR). Because of cleaning and resetting of machinery, the time required to produce a batch of gasoline depends on the type of gasoline last produced. For example, it takes longer to switch between a lead-free gasoline and a leaded gasoline than it does to switch between two lead-free gasolines. The time (in minutes) required to manufacture each day's gasoline requirements are shown in Table 75. Use a branch-and-bound approach to determine the order in which the gasolines should be produced each day.

3 A Hamiltonian path in a network is a closed path that passes exactly once through each node in the network before returning to its starting point. Taking a four-city TSP as an example, explain why solving a TSP is equivalent to finding the shortest Hamiltonian path in a network.

4 There are four pins on a printed circuit. The distance between each pair of pins (in inches) is given in Table 76.

a Suppose we want to place three wires between the pins in a way that connects all the wires and uses the minimum amount of wire. Solve this problem by using one of the techniques discussed in Chapter 8.

b Now suppose that we again want to place three wires between the pins in a way that connects all the wires and uses the minimum amount of wire. Also suppose that if more than two wires touch a pin, a short circuit will occur. Now set up a traveling salesperson problem that can be used to solve this problem. (*Hint:* Add a pin 0 such that the distance between pin 0 and any other pin is 0.)

5 **a** Use the NNH to find a solution to the TSP in Problem 2. Begin with LFR.

b Use the CIH to find a solution to the TSP in Problem 2. Begin with the subtour LFR–LFP–LFR.

6 LL Pea stores clothes at five different locations. Several times a day it sends an "order picker" out to each location to pick up orders. Then the order picker must return to the packaging area. Describe a TSP that could be used to minimize the time needed to pick up orders and return to the packaging area.

Group B

7 Use branch-and-bound to determine a way (if any exists to place four queens on a 4 × 4 chessboard so that no queen can capture another queen. (*Hint:* Let $x_{ij} = 1$ if a queen is placed in row i and column j of the chessboard and $x_{ij} = 0$ otherwise. Then branch as in the machine-delay problem.

TABLE 74

Job	Time to Perform Job (Minutes)	Due Date of Job
1	7	End of minute 14
2	5	End of minute 13
3	9	End of minute 18
4	11	End of minute 15

TABLE 75

Last-Produced Gasoline	Gas to Be Next Produced			
	LFR	LFP	LR	LP
LFR	—	50	120	140
LFP	60	—	140	110
LR	90	130	—	60
LP	130	120	80	—

Note: Assume that the last gas produced yesterday precedes the first gas produced today.

TABLE 76

	1	2	3	4
1	0	1	2	2
2	1	0	3	2.9
3	2	3	0	3
4	2	2.9	3	0

Many nodes may be eliminated from consideration because they are infeasible. For example, the node associated with the arcs $x_{11} = x_{22} = 1$ is infeasible, because the two queens can capture each other.)

8 Although the Hungarian method is an efficient method for solving an assignment problem, the branch-and-bound method can also be used to solve an assignment problem. Suppose a company has five factories and five warehouses. Each factory's requirements must be met by a single warehouse, and each warehouse can be assigned to only one factory. The costs of assigning a warehouse to meet a factory's demand (in thousands) are shown in Table 77.

Let $x_{ij} = 1$ if warehouse i is assigned to factory j and 0 otherwise. Begin by branching on the warehouse assigned to factory 1. This creates the following five branches: $x_{11} = 1$, $x_{21} = 1$, $x_{31} = 1$, $x_{41} = 1$, and $x_{51} = 1$. How can we obtain a lower bound on the total cost associated with a branch? Examine the branch $x_{21} = 1$. If $x_{21} = 1$, no further assignments can come from row 2 or column 1 of the cost matrix. In determining the factory to which each of the unassigned warehouses (1, 3, 4, and 5) is assigned, we cannot do better than assign each to the smallest cost in the warehouse's row (excluding the factory 1 column). Thus, the minimum-cost assignment having $x_{21} = 1$ must have a total cost of at least $10 + 10 + 9 + 5 + 5 = 39$.

Similarly, in determining the warehouse to which each of the unassigned factories (2, 3, 4, and 5) is assigned, we cannot do better than to assign each to the smallest cost in the factory's column (excluding the warehouse 2 row). Thus, the minimum-cost assignment having $x_{21} = 1$ must have a total cost of at least $10 + 9 + 5 + 5 + 7 = 36$. Thus, the total cost of any assignment having $x_{21} = 1$ must be at least $\max(36, 39) = 39$. So, if branching ever leads to a candidate solution having a total cost of 39 or less, the $x_{21} = 1$ branch may be eliminated from consideration. Use this idea to solve the problem by branch-and-bound.

9[†] Consider a long roll of wallpaper that repeats its pattern every yard. Four sheets of wallpaper must be cut from the roll. With reference to the beginning (point 0) of the wallpaper, the beginning and end of each sheet are located as shown in Table 78. Thus, sheet 1 begins 0.3 yd from the beginning of the roll (and 1.3 yd from the beginning of the roll) and sheet 1 ends 0.7 yd from the beginning of the roll (and 1.7 yd from the beginning of the roll). Assume we are at the beginning of the roll. In what order should the sheets be cut to minimize the total amount of wasted paper? Assume that a final cut is made to bring the roll back to the beginning of the pattern.

10[‡] A manufacturer of printed circuit boards uses programmable drill machines to drill six holes in each board. The x and y coordinates of each hole are given in Table 79. The time (in seconds) it takes the drill machine to move from one hole to the next is equal to the distance between the points. What drilling order minimizes the total time that the drill machine spends moving between holes?

11 Four jobs must be processed on a single machine. The time required to perform each job, the due date, and the penalty (in dollars) per day the job is late are given in Table 80.

Use branch-and-bound to determine the order of performing the jobs that will minimize the total penalty costs due to delayed jobs.

TABLE 77

	Factory ($)				
Warehouse	1	2	3	4	5
1	5	15	20	25	10
2	10	12	5	15	19
3	5	17	18	9	11
4	8	9	10	5	12
5	9	10	5	11	7

TABLE 78

Sheet	Beginning (Yards)	End (Yards)
1	0.3	0.7
2	0.4	0.8
3	0.2	0.5
4	0.7	0.9

TABLE 79

x	y	Hole
1	2	1
3	1	2
5	3	3
7	2	4
8	3	5

TABLE 80

Job	Time (Days)	Due Date	Penalty
1	4	Day 4	4
2	5	Day 2	5
3	2	Day 13	7
4	3	Day 8	2

[†]Based on Garfinkle (1977).

[‡]Based on Magirou (1986).