

## Integer Programming

When only integer solutions are practical or logical, it is sometimes assumed that noninteger solution values can be "rounded off" to the nearest feasible integer values. This method would cause little concern if, for example,  $x_1 = 8,000.4$  nails were rounded off to 8,000 nails because nails cost only a few cents apiece. However, if we are considering the production of jet aircraft and  $x_1 = 7.4$  jet airliners, rounding off could affect profit (or cost) by millions of dollars. In this case we need to solve the problem so that an optimal integer solution is guaranteed. In this chapter the different forms of integer linear programming models are presented.

There are three basic types of integer linear programming models: a total integer model, a 01 integer model, and a mixed integer model. In a total integer model, all the decision variables are required to have integer solution values. In a **01 integer model**, all the decision variables have integer values of zero or one. Finally, in a **mixed integer model**, some of the decision variables (but not all) are required to have integer solutions. The following three examples demonstrate each of these types of integer programming models.

The three types of integer programming models are total, 01, and mixed.

### A Total Integer Model Example

The owner of a machine shop is planning to expand by purchasing some new machines presses and lathes. The owner has estimated that each press purchased will increase profit by €100 per day and each lathe will increase profit by €150 daily. The number of machines the owner can purchase is limited by the cost of the machines and the available floor space in the shop. The machine purchase prices and space requirements are as follows :

| Machine | Required Floor Space (m <sup>2</sup> ) | Purchase Price |
|---------|--|----------------|
| Press   | 15                                     | €8,000         |
| Lathe   | 30                                     | €4,000         |

In a total integer model, all decision variables have integer solution values.

The owner has a budget of €40,000 for purchasing machines and 200 square meters of available floor space. The owner wants to know how many of each type of machine to purchase to maximize the daily increase in profit.

The linear programming model for an integer programming problem is formulated in exactly the same way as the non-integer linear programming examples. The only difference is that in this problem, the decision variables are restricted to integer values because the owner cannot purchase a fraction, or portion, of a machine. The linear programming model follows:

$$\text{Max } 100X_1 + 150X_2$$

s.t.

$$\text{Available budget) } 8000X_1 + 4000X_2 \leq 40000$$

$$\text{Floor space) } 15X_1 + 30X_2 \leq 200$$

$$X_1, X_2 \geq 0 \text{ and integer}$$

where here  $X_1$  = number of presses and  $X_2$  = number of Lathes.

The decision variables in this model are restricted to whole machines. The fact that both decision variables,  $X_1$  and  $X_2$ , can assume any integer value greater than or equal to zero is what gives this model its designation as a total integer model.

## A 01 Integer Model Example

A community council must decide which recreation facilities to construct in its community. Four new recreation facilities have been proposed: a swimming pool, a tennis center, an athletic field, and a gymnasium. The council wants to construct facilities that will maximize the expected daily usage by the residents of the community, subject to land and cost limitations. The expected daily usage and cost and land requirements for each facility follow:

| <b>Recreation Facility</b> | <b>Expected Usage (people/day)</b> | <b>Cost</b> | <b>Land Requirements (hectares)</b> |
|----------------------------|------------------------------------|-------------|-------------------------------------|
| Swimming pool              | 300                                | €35,000     | 4                                   |
| Tennis center              | 90                                 | 10,000      | 2                                   |
| Athletic field             | 400                                | 25,000      | 7                                   |
| Gymnasium                  | 150                                | 90,000      | 3                                   |

In a 01 integer model, the solution values of the decision variables are zero or one.

The community has a €120,000 construction budget and 12 hectares of land. Because the swimming pool and tennis center must be built on the same part of the land parcel, however, only one of these two facilities can be constructed. The council wants to know which of the recreation facilities to construct to maximize the expected daily usage. The model for this problem is formulated as follows:

Let

$x_1$  = construction of a swimming pool

$x_2$  = construction of a tennis center

$x_3$  = construction of an athletic field

$x_4$  = construction of a gymnasium

Max  $300x_1 + 90x_2 + 400x_3 + 150x_4$

s.t.

Budget)  $35000x_1 + 10000x_2 + 25000x_3 + 90000x_4 \leq 120000$

Land)  $4x_1 + 2x_2 + 7x_3 + 3x_4 \leq 12$

Facilities)  $x_1 + x_2 \leq 1$

$x_1, x_2, x_3, x_4 = 0, 1$

In this model, the decision variables can have a solution value of either zero or one. If a facility is not selected for construction, the decision variable representing it will have a value of zero. If a facility is selected, its decision variable will have a value of one.

The last constraint,  $x_1 + x_2 \leq 1$ , reflects the contingency that *either the swimming pool ( $x_1$ ) or the tennis center ( $x_2$ ) can be constructed, but not both*. In order for the sum of  $x_1$  and  $x_2$  to be less than or equal to one, either of the variables can have a value of one, or both variables can equal zero. This is also referred to as a mutually exclusive constraint.

If the community had specified that *either the swimming pool ( $x_1$ ) or the tennis center ( $x_2$ ) must be built*, but not both, then the last constraint would become an equation,  $x_1 + x_2 = 1$ . This would result in a solution that would include  $x_1 = 1$  or  $x_2 = 1$ , but both would not equal one (nor would both equal zero). In this manner, the model forces a choice between the two facilities. For this reason, it is often called a multiple-choice constraint .

A variation of the multiple-choice constraint can be used to formulate a situation where some specific number of facilities out of the total must be constructed. For example, if the community council had specified that exactly two of the four facilities must be built, this constraint would be formulated as

$$x_1 + x_2 + x_3 + x_4 = 2$$

If, alternatively, the council had specified that no more than two facilities must be constructed, the constraint would be

$$x_1 + x_2 + x_3 + x_4 \leq 2$$

Another type of 01 model constraint is a conditional constraint. In a conditional constraint, the construction of one facility would be conditional upon the construction of another.

Suppose, for example, that the pet project of the head of the community council is the swimming pool, and she also believes the tennis center is frivolous. The council head is very influential, so the rest of the council knows that the tennis center has no chance of being selected if the pool is not selected first. However, even if the pool is selected, there is no guarantee that the tennis center will also be selected. Thus, the tennis center ( $x_2$ ) is conditional upon construction of the swimming pool ( $x_1$ ). This condition is formulated as

$$x_2 \leq x_1$$

Notice that the tennis center ( $x_2$ ) cannot equal one (i.e., be selected) unless the pool ( $x_1$ ) equals one. If the pool ( $x_1$ ) equals zero (i.e., it is not selected), then the tennis center ( $x_2$ ) must also equal zero. However, this condition does allow the pool ( $x_1$ ) to equal one and be selected and the tennis center to equal zero and not be selected.

A variation of this type of conditional constraint is the corequisite constraint, wherein if one facility is constructed, the other one would also be constructed and vice versa. For example, suppose the council worked out a political deal among themselves, wherein if the pool is accepted, the tennis center must also be selected and vice versa. This constraint is written as

$$x_2 = x_1$$

This constraint makes  $x_1$  and  $x_2$  equal the same value, either zero or one.

### **A Mixed Integer Model Example**

Nancy Smith has €250,000 to invest in three alternative investments condominiums, land, and municipal bonds. She wants to invest in the alternatives that will result in the greatest return on investment after 1 year.

In a mixed integer model, some solution values for decision variables are integers and others can be nonintegers.

Each condominium costs €50,000 and will return a profit of €9,000 if sold at the end of 1 year; each acre of land costs €12,000 and will return a profit of €1,500 at the end of 1 year; and each municipal bond costs €8,000 and will result in a return of €1,000 if sold at the end of 1 year. In addition, there are only 4 condominiums, 15 acres of land, and 20 municipal bonds available for purchase.

The linear programming model for this problem is formulated as follows:

$x_1$  = condominiums purchased

$x_2$  = acres of land purchased

$x_3$  = bonds purchased

$$\max 9000x_1 + 1500x_2 + 1000x_3$$

s.t.

$$\text{Budget) } 50000x_1 + 12000x_2 + 8000x_3 \leq 250000$$

$$\text{condominium) } x_1 \leq 4$$

$$\text{Acres) } x_2 \leq 15$$

$$\text{Bonds) } x_3 \leq 20$$

$$x_2 \geq 0 \text{ and } x_1, x_3 \geq 0 \text{ and integer}$$

Notice that in this model the solution values for condominiums ( $x_1$ ) and municipal bonds ( $x_3$ ) must be integers. It is not possible to invest in a fraction of a condominium or to purchase part of a bond. However, it is possible to purchase less than an acre of land (i.e., a portion of an acre). Thus, two of the decision variables ( $x_1$  and  $x_3$ ) are restricted to integer values, whereas the other variable ( $x_2$ ) can take on any real value greater than or equal to zero.

## Integer Programming Graphical Solution

It may seem logical that an easy way to solve integer programming problems is to round off fractional solution values to integer values. However, that can result in less than optimal (i.e., suboptimal) results. This outcome can be seen by using graphical analysis. As an example, consider the total integer model for the machine shop formulated in the previous section:

$$\text{Max } Z = 100X_1 + 150X_2$$

s.t.

$$\text{Available budget) } 8000X_1 + 4000X_2 \leq 40000$$

$$\text{Floor space) } 15X_1 + 30X_2 \leq 200$$

$$X_1, X_2 \geq 0 \text{ and integer}$$

where here  $X_1$  = number of presses and  $X_2$  = number of Lathes.

First, we will solve this model as a regular linear programming model without the integer requirements. This model results in a noninteger solution of 2.22 presses and 5.56 lathes, or  $X_1 = 2.22$  and  $X_2 = 5.56$ . Because the solution values must be integers, let us first round off these two values to the closest integer values, which would be  $X_1 = 2$  and  $X_2 = 6$ . However, if we substitute these values into the second model constraint, we find that this integer solution violates the constraint and thus is infeasible:

$$15X_1 + 30X_2 \leq 200$$

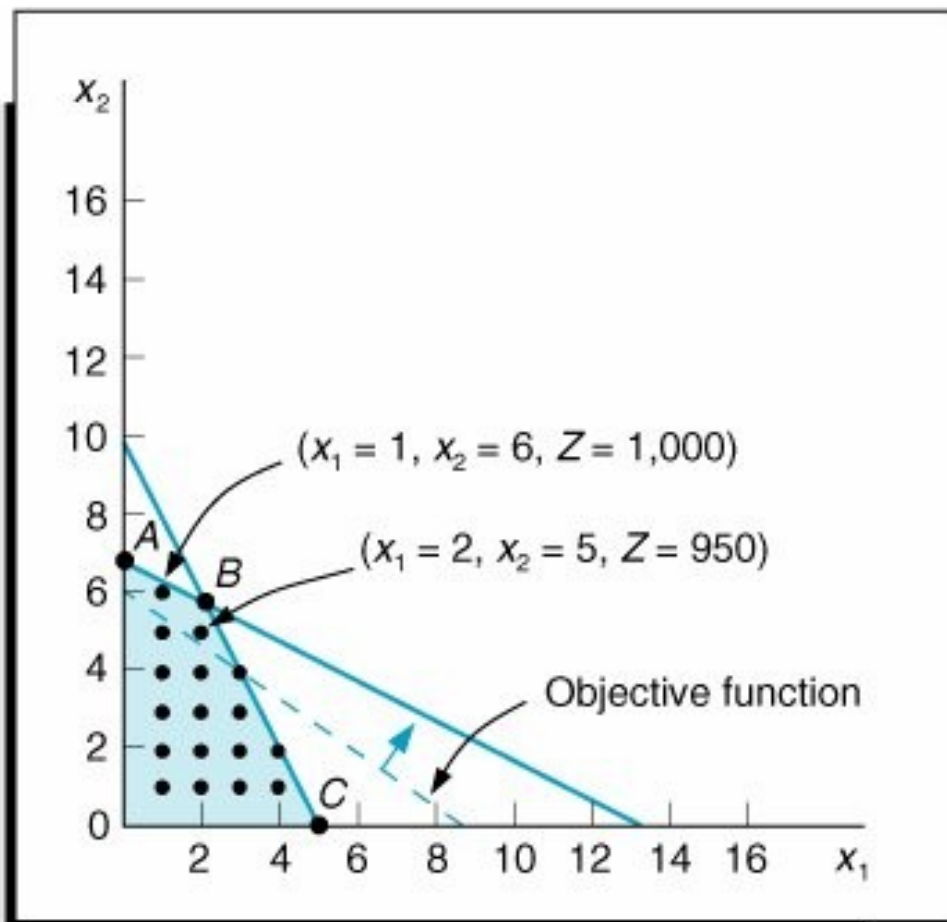
$$15(2) + 30(6) \leq 200 \text{ i.e. } 210 \leq 200$$

Rounding noninteger solution values up to the nearest integer value can result in an infeasible solution.

**In a model in which the constraints are all  $\leq$  (and the constraint coefficients are positive), a feasible solution is always ensured by rounding down.** Thus, a feasible integer solution for this problem is  $X_1 = 2$ ,  $X_2 = 5$  and with profit  $Z = \text{€}950$ .

A feasible solution is ensured by rounding down noninteger solution values.

However, one of the difficulties of simply rounding down non-integer values is that another integer solution may result in a higher profit (i.e., in this problem, there may be an integer solution that will result in a profit higher than €950). In order to determine whether a better integer solution exists, let us analyze the graph of this model, which is shown in the Figure below.



In Figure above the dots indicate integer solution points, and the point  $X_1=2, X_2=5$  is the rounded-down solution. Notice that as the objective function edge moves outward through the feasible solution space, there is an integer solution point that yields a greater profit than the rounded-down solution. This solution point is  $X_1=1, X_2=6$ . At this point,  $Z=€1,000$ , which is €50 more profit per day for the machine shop than is realized by the rounded-down integer solution.

A rounded-down integer solution can result in a less than optimal (suboptimal) solution.

This graphical analysis explicitly demonstrates the error that can result from solving an integer programming problem by simply rounding down. In the machine shop example, the optimal integer solution is  $X_1=1, X_2=6$ , instead of the rounded-down solution,  $X_1=2, X_2=5$  (which is often called the suboptimal solution or result). Because erroneous results are caused by rounding down regular solutions, a more direct approach for solving integer problems is required.

The traditional approach for solving integer programming problems is the **branch and bound method**. It is a mathematical solution approach that can be applied to a number of different types of problems. The branch and bound method is based on the principle that the total set of feasible solutions (such as the feasible area in the Figure above) can be partitioned into smaller subsets of solutions. These smaller subsets can then be evaluated systematically until the best solution is found. The branch and bound method is a tedious and often complex mathematical process.