Integer Programming - I

1

Outline

- Prototype Example
- Some BIP Applications
- Innovative Uses of Binary Variables in Model Formulation
- Some Formulation Examples

2

Integer Programming

- We have seen examples of applications of linear programming where noninteger values were permissible for decision variables.
- In many practical problems, the decision variables actually make sense only if they have integer values.
  - For example, it is often necessary to assign people, machines, and vehicles to activities in integer quantities.
- If requiring integer values is the only way in which a problem deviates from a linear programming formulation, then it is an integer programming (IP) problem.

3

Integer Programming

- If only *some* of the variables are required to have integer values (so the divisibility assumption holds for the rest), this model is referred to as mixed integer programming (MIP).
- Another area of application of integer programming are problems involving a number of interrelated "yes-or-no" decisions.
  - These problems are binary integer programming (BIP) problems.

4

Typical BIP Problem

- Example: CMC is considering building a factory in either LA or SF, or in both cities.
- It also is considering building at most one new warehouse, but the choice of location is restricted to a city where a new factory is being built.
- The *net present value* (total profitability considering the time value of money) of each shown in fourth column

TABLE 11.1 Data for the California Manufacturing Co. example

Decision Number	Yes-or-No Question	Decision Variable	Net Present Value	Capital Required
1	Build factory in Los Angeles?	x_1	\$9 million	\$6 million
2	Build factory in San Francisco?	x_2	\$5 million	\$3 million
3	Build warehouse in Los Angeles?	x_3	\$6 million	\$5 million
4	Build warehouse in San Francisco?	x_4	\$4 million	\$2 million
			Capital available:	\$10 million

Typical BIP Problem

- Total capital available is \$10 million.
- The objective is to find the feasible combination of alternatives that maximizes the total net present value.

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

TABLE 11.1 Data for the California Manufacturing Co. example

Decision Number	Yes-or-No Question	Decision Variable	Net Present Value	Capital Required
1	Build factory in Los Angeles?	x_1	\$9 million	\$6 million
2	Build factory in San Francisco?	x_2	\$5 million	\$3 million
3	Build warehouse in Los Angeles?	x_3	\$6 million	\$5 million
4	Build warehouse in San Francisco?	x_4	\$4 million	\$2 million
			Capital available:	\$10 million

Typical BIP Problem

- $Z = 9x_1 + 5x_2 + 6x_3 + 4x_4$
- The rightmost column of Table 12.1 indicates that the amount of capital expended on the four facilities cannot exceed \$10 million.
- Consequently, continuing to use units of millions of dollars, one constraint in the model is

$$6x_1 + 3x_2 + 5x_3 + 2x_4 \le 10$$

- Because the last two decisions represent *mutually exclusive alternatives* (the company wants *at most* one new warehouse), we also need the constraint

$$x_3 + x_4 \le 1$$

Typical BIP Problem

- Decisions 3 and 4 are contingent on decisions 1 and 2, respectively (the company would consider building a warehouse in a city only if a new factory also were going there).

$x_3 \le x_1$ & $x_4 \le x_2$ $x_4 = 0$ if $x_2 = 0$

- Therefore, after we rewrite these two constraints to bring all variables to the left-hand side, the complete BIP model is

Maximize $Z = 9x_1 + 5x_2 + 6x_3 + 4x_4$

subject to

$$\begin{aligned}
6x_1 + 3x_2 + 5x_3 + 2x_4 &\le 10 \\
x_3 + x_4 &\le 1 \\
-x_1 + x_3 &\le 0 \\
-x_2 + x_4 &\le 0 \\
x_j &\le 1 \\
x_j &\ge 0
\end{aligned}$$

Some BIP Applications

- Investment analysis: Should we make a certain fixed investment?
- Site selection: Should a certain site be selected for the location of a certain new facility?
- Designing a production and distribution network
  - Should a certain plant remain open?
  - Should a certain site be selected for a new plant?
- Dispatching shipment: How to send the shipments?
  - Should a certain route be selected for one of the trucks?
  - 1) A certain route 2) A certain size of truck 3) A certain time period for the departure

Some BIP Applications

- Scheduling interrelated activies: When should we begin production for various new orders? Should we begin marketing various new products?
- Airline activities: fleet assignment problem.
  - Given several different types of airplanes, the problem is to assign a specific type to each flight leg in the schedule.
  - Should a certain type be assigned to a certain flight leg?

Innovative Uses of Binary Variables in the Model Formulation

- *Auxiliary* binary variables are introduced into the model to simply help formulate the model as a pure or mixed BIP model.
  - x_j : *original* variables of the problem
  - y_i : *auxiliary* binary variables

Either-Or Constraints

- Either-or constraints: A choice can be made between two constraints, so that *only one* must hold.
- Suppose that

  Either $3x_1 + 2x_2 \leq 18$

  or $x_1 + 4x_2 \leq 16$
- At least one of these must hold but not necessarily both.
- This requirement must be reformulated to fit into LP format where *all* specified constraints must hold.
- Let *M* be a very large positive number. Then,

Either $3x_1 + 2x_2 \leq 18$, $x_1 + 4x_2 \leq 16 + M$

or $3x_1 + 2x_2 \leq 18 + M$, $x_1 + 4x_2 \leq 16$

Either-Or Constraints

- M has the effect of eliminating them because they would be satisfied automatically by any solutions that satisfy the other constraints of the problem. This formulation is equivalent to the set of constraints

  $3x_1 + 2x_2 \leq 18 + My$

  $x_1 + 4x_2 \leq 16 + M(1-y)$
- Because the *auxiliary variable* y must be either 0 or 1, this formulation guarantees that one of the original constraints must hold while the other is eliminated.
- This new set of constraints would then be appended to the other constraints in the overall model to give a pure or mixed IP problem (depending upon whether the x_j are integer or continuous variables)

13

Either-Or Constraints

- Should $x_1 + 4x_2 \leq 16$ or $3x_1 + 2x_2 \leq 18$ be selected?
- Because exactly one of these questions is to be answered affirmatively, we let the binary terms y and 1 –y, respectively, represent these yes-or-no decisions.
- Thus, y = 1 if the answer is yes to the first question (and no to the second), whereas 1-y = 1 (that is,y=0) if the answer is yes to the second question (and no to the first).
- Since y + 1 - y = 1 (one yes) automatically, there is no need to add another constraint to force these two decisions to be mutually exclusive. (If separate binary variables y_1 and y_2 had been used instead to represent these yes-or-no decisions, then an additional constraint $y_1 + y_2 =1$ would have been needed to make them mutually exclusive.)
- A formal presentation of this approach is given next for a more general case.

14

K out of N Constraints Must Hold

- Consider N possible constraints such that only some K of these *must* hold. (Assume that K < N.)
- The N – K constraints *not* chosen are eliminated from the problem.
- This case is a direct generalization of the preceding case, which had K = 1 and N = 2.

$$\begin{aligned} f_1(x_1, x_2, \dots, x_n) &\le d_1 \\ f_2(x_1, x_2, \dots, x_n) &\le d_2 \\ &\vdots \\ f_N(x_1, x_2, \dots, x_n) &\le d_N \end{aligned} \quad \Rightarrow \quad \begin{aligned} f_1(x_1, x_2, \dots, x_n) &\le d_1 + My_1 \\ f_2(x_1, x_2, \dots, x_n) &\le d_2 + My_2 \\ &\vdots \\ f_N(x_1, x_2, \dots, x_n) &\le d_N + My_N \end{aligned}$$

$$\sum_{i=1}^{N} y_i = N - k$$

- Note that $y_i = 0$ makes $My_i = 0$, which reduces the new constraint i to the original constraint i.
- On the other hand, $y_i = 1$ makes $(d_i + My_i)$ so large that the new constraint i is automatically satisfied by any solution

K out of N Constraints Must Hold

- Therefore, because the constraints on the y_i guarantee that K of these variables will equal 0 and those remaining will equal 1, K of the original constraints will be unchanged and the other (N – K) original constraints will, in effect, be eliminated.

Functions with N Possible Values

- Consider the situation where a given function is required to take on any one of N given values. Denote this requirement by

$f(x_1, x_2, \dots, x_n) = d_1$ or $d_2, \dots,$ or d_N.

One special case is where this function is

$f(x_1, x_2, \dots, x_n) = \sum_{j=1}^{n} a_j x_j$

- as on the left-hand side of a linear programming constraint.
- Another special case is where $f(x_1, x_2, \dots, x_n) = x_j$ for a given value of j, so the requirement becomes that x_j must take on any one of N given values.
- The equivalent IP formulation of this requirement is the following

$f(x_1, x_2, \dots, x_n) = \sum_{i=1}^{n} d_i y_i$, $\sum_{i=1}^{n} y_i = 1$, and y_i is binary for i = 0 1,2, …, N

- In this case, there are N yes-or-no questions being asked.

17

Illustration through Wyndor Glass Co. Problem

- Management now wants to impose the restriction that the production time used by the two current new products be 6 *or* 12 *or* 18 hours per week.
- Thus, the third constraint of the original model ($3x_1 + 2x_2 \leq 18$) now becomes

$3x_1 + 2x_2 \leq 6$ or 12 or 18.

- In the preceding notation, N=3 with d_1=6, d_2=12, and d_3=18. Consequently, management's new requirement should be formulated as follows:

$3x_1 + 2x_2 \leq 6y_1 + 12y_2 + 18y_3$

$y_1 + y_2 + y_3 = 1$ and y_1, y_2, y_3 are binary.

18

The Fixed-Charge Problem

- It is quite common to incur a fixed charge or setup cost when undertaking an activity.
- For example, such a charge occurs when a production run to produce a batch of a particular product is undertaken and the required production facilities must be set up to initiate the run.
- In such cases, the total cost of the activity is the sum of a variable cost related to the level of the activity and the setup cost required to initiate the activity.
- Frequently the variable cost will be at least roughly proportional to the level of the activity.
- If this is the case, the total cost of the activity (say, activity j) can be represented by a function of the form

$$f_j(x_j) = \begin{cases} k_j + c_j x_j & \text{if } x_j > 0 \\ 0 & \text{if } x_j = 0 \end{cases}$$

- where x_j denotes the level of activity j ($x_j \geq 0$), k_j denotes the setup cost, and c_j denotes the cost for each incremental unit.

19

The Fixed-Charge Problem

Minimize $Z = f_1(x_1) + f_2(x_2) + \ldots + f_n(x_n)$
subject to
given linear programming constraints

- To convert this problem to an MIP format, n questions must be answered yes or no. Each of these yes-or-no decisions is then represented by an auxiliary binary variable y_j, so that

$$Z = \sum_{j=1}^{n} (c_j x_j + k_j y_j) \qquad \text{where, } y_j = \begin{cases} 1 & \text{if } x_j > 0 \\ 0 & \text{if } x_j = 0 \end{cases}$$

- Let M be an extremely large positive number that exceeds the maximum feasible value of any x_j (j = 1, 2, . . . , n)
- Then, the constraints

$x_j \leq M y_j$ for j = 1, 2, . . . , n

will ensure that $y_j = 1$ rather than 0 whenever $x_j > 0$.

20

The Fixed-Charge Problem

- The one difficulty remaining is that these constraints leave y_j free to be either 0 or 1 when x_j=0. Fortunately, this difficulty is automatically resolved because of the nature of the objective function.
- The case where $k_j = 0$ can be ignored because y_j can then be deleted from the formulation.
- So we consider the only other case, namely, where $k_j >$ 0.
- When $x_j = 0$, so that the constraints permit a choice between $y_j = 0$ and $y_j = 1$, $y_j = 0$ must yield a smaller value of Z than $y_j = 1$.
- Therefore, because the objective is to minimize Z, an algorithm yielding an optimal solution would always choose $y_j = 0$ when $x_j = 0$.

The Fixed-Charge Problem

- To summarize, the MIP formulation of the fixed-charge problem is

  Minimize $Z = \sum_{j=1}^{n} (c_j x_j + k_j y_j)$

  subject to

  the original constraints, plus

  $x_j - M y_j \leq 0$

  and, y_j is binary, for j = 1, 2, . . . , n
- If the x_j also had been restricted to be integer, then this would be a *pure* IP problem.

Binary Representation of General Integer Variables

- Suppose that you have a pure IP problem where most of the variables are *binary* variables, but the presence of a few *general* integer variables prevents you from solving the problem by one of the very efficient BIP algorithms now available.
- A nice way to circumvent this difficulty is to use the *binary representation* for each of these general integer variables.
- Specifically, if the bounds on an integer variable x are

  $0 \le x \le u$ and if N is defined as the integer such that

  $2^N \le u \le 2^{N+1}$
- Then, the binary representation of x is

  $$x = \sum_{i=0}^{N} 2^i y_i$$, where y_i variables are auxiliary binary variables

Binary Representation of General Integer Variables

- $x_1 \le 5$ u = 5 for x_1
- $2x_1 + 3x_2 \le 30$ u = 10 for x_2
- N = 2 for x_1 since $(2^2 \le 5 \le 2^3)$, N = 3 for x_2 since $(2^3 \le 10 \le 2^4)$
- Therefore,

  $x_1 = y_0 + 2y_1 + 4y_2$ $x_2 = y_3 + 2y_4 + 4y_5 + 8y_6$
- Substituting these equations for x_1 and x_2 into the inequalities at the top, we obtain

  $y_0 + 2y_1 + 4y_2 \le 5$

  $2y_0 + 4y_1 + 8y_2 + 3y_3 + 6y_4 + 12y_5 + 24y_6 \le 30$
- For an IP problem where all the variables are (bounded) general integer variables, it is possible to use this same technique to reduce the problem to a BIP model.

Some Formulation Examples: Ex 1 (Making Choices When the Decision Variables Are Continuous)

- Restriction 1: From the three possible new products, at most two should be chosen to be produced.
- Restriction 2: Just one of the two plants should be chosen to be the sole producer of the new products.
- A standard product mix problem

TABLE 11.2 Data for Example 1 (the Good Products Co. problem)

	Production Time Used for Each Unit Produced			Production Time Available per Week
	Product 1	Product 2	Product 3	
Plant 1	3 hours	4 hours	2 hours	30 hours
Plant 2	4 hours	6 hours	2 hours	40 hours
Unit profit	5	7	3	(thousands of dollars)
Sales potential	7	5	9	(units per week)

Some Formulation Examples: Ex 1 (Making Choices When the Decision Variables Are Continuous)

Maximize $Z = 5x_1 + 7x_2 + 3x_3$

such that

$$3x_1 + 4x_2 + 2x_3 \le 30$$
$$4x_1 + 6x_2 + 2x_3 \le 40$$
$$x_1 \le 7$$
$$x_2 \le 5$$
$$x_3 \le 9$$

$$x_1, x_2, x_3 \ge 0$$

$$x_1 + x_2 + x_3 \text{ must be } \le 2$$

- This constraint does not fit into our format.
- If the decision variables were binary variables, we could write $x_1+x_2+x_3 \le 2$.
- However, with continuous decision variables, a more complicated approach is needed.

Some Formulation Examples: Ex 1 (Making Choices When the Decision Variables Are Continuous)

- Requirement 2

  Either $3x_1 + 4x_2 + 2x_3 \le 30$ or
  $4x_1 + 6x_2 + 2x_3 \le 40$ selected

- Formulating with auxiliary binary variables
- We introduce

$$y_j = \begin{cases} 1 & \text{if } x_j > 0 \ (can\ produce\ product\ j) \\ 0 & \text{if } x_j = 0 \ (cannot\ produce\ product\ j) \end{cases}$$

$x_1 \le My_1$ $\quad$ $y_1+y_2+y_3 \le 2$. $\quad$ y_j is binary. j = 1,2,3

$x_2 \le My_2$

$x_3 \le My_3$

- To deal with the second requirement

$$y_4 = \begin{cases} 1 & \text{if } 4x_1 + 6x_2 + 2x_3 \le 40 \ \ (choose\ P2) \\ 0 & \text{if } 3x_1 + 4x_2 + 2x_3 \le 30 \ \ (choose\ P1) \end{cases}$$

27

Some Formulation Examples: Ex 1 (Making Choices When the Decision Variables Are Continuous)

$$3x_1 + 4x_2 + 2x_3 \le 30 + My_4$$
$$4x_1 + 6x_2 + 2x_3 \le 40 + M(1 - y_4)$$ y_4 is binary.

- This is now an MIP model, with three (the x_j) variables not required to be integer and four binary variables, so an MIP algorithm can be used to solve the model.
- The optimal solution is $y_1 = 1$, $y_2 = 0$, $y_3 = 1$, $y_4 = 1$, $x_1 = 5\ 1/2$, $x_2 = 0$, and $x_3 = 9$.

28

Some Formulation Examples: Ex 2 (Violating Proportionality)

- The SUPERSUDS CORPORATION will purchase a total of five TV spots for commercials for 3 products. The problem we will focus on is how to allocate the five spots to these three products with a maximum of three spots (and a minimum of zero) for each product.
- This table is obtained considering the cost of producing commercials-additional sales.
- By inspection (The optimal solution is to allocate two spots to product 1, no spots to product 2, and three spots to product 3.)

TABLE 11.3 Data for Example 2 (the Supersuds Corp. problem)

Number of TV Spots	Profit: Product 1	Profit: Product 2	Profit: Product 3
0	0	0	0
1	1	0	−1
2	3	2	2
3	3	3	4

The linear objective function gives the total profit as marked on the table

Some Formulation Examples: Ex 2 (Violating Proportionality)

Formulation with Auxiliary Binary Variables.

- A natural formulation would be to let x_1, x_2, x_3 be the number of TV spots allocated to the respective products.
- In our previous examples, $z = 3x_1 + 5x_2$, there was proportionality.
- However, each of these columns violates the assumption of proportionality.
- Therefore, we cannot write a *linear* objective function in terms of these integer decision variables.
- We introduce an auxiliary binary variable y_{ij} for each positive integer value of $x_i = j$ ($j = 1,2,3$), where y_{ij} has the interpretation

$$y_{ij} = \begin{cases} 1 & \text{if } x_i = j \\ 0 & otherwise \end{cases}$$

(For example, $y_{21} = 0$, $y_{22} = 0$, and $y_{23} = 1$ mean that $x_2 = 3$)

Some Formulation Examples: Ex 2 (Violating Proportionality)

Formulation with Auxiliary Binary Variables

The resulting linear BIP model is

The linear objective function gives the total profit according to Table 11.3

Maximize $Z = y_{11} + 3y_{12} + 3y_{13} + 2y_{22} + 3y_{23} - y_{31} + 2y_{32} + 4y_{33}$

subject to

Ensure that each x_i will be assigned just one of its possible values.

$$y_{11} + y_{12} + y_{13} \leq 1$$
$$y_{21} + y_{22} + y_{23} \leq 1$$
$$y_{31} + y_{32} + y_{33} \leq 1$$

$$y_{11} + 2y_{12} + 3y_{13} + y_{21} + 2y_{22} + 3y_{23} + y_{31} + 2y_{32} + 3y_{33} = 5$$

The last functional constraint ensures that $x_1 + x_2 + x_3 = 5$.

31

Some Formulation Examples: Ex 2 (Violating Proportionality)

Formulation with Auxiliary Binary Variables

Solving this BIP

y_{11}=0, y_{12}=1, y_{13}=0, so x_1=2

y_{21}=0, y_{22}=0, y_{23}=0, so x_2=0

y_{31}=0, y_{32}=0, y_{33}=1, so x_3=3

32

Some Formulation Examples: Ex 2 (Violating Proportionality)

Another Formulation

$$y_{ij} = \begin{cases} 1 & \text{if } x_i > j \\ 0 & otherwise \end{cases}$$

Thus

x_i=0 ⇨ y_{i1}=0, y_{i2}=0, y_{i3}=0 so $x_i = y_{i1} + y_{i2} + y_{i3}$ for i = 1,2,3

x_i=1 ⇨ y_{i1}=1, y_{i2}=0, y_{i3}=0

x_i=2 ⇨ y_{i1}=1, y_{i2}=1, y_{i3}=0

x_i=3 ⇨ y_{i1}=1, y_{i2}=1, y_{i3}=1

Because allowing y_{i2}=1 is contingent upon y_{i1}=1 and allowing y_{i3}=1 is contingent upon y_{i2}=1, these definitions are enforced by adding the constraints

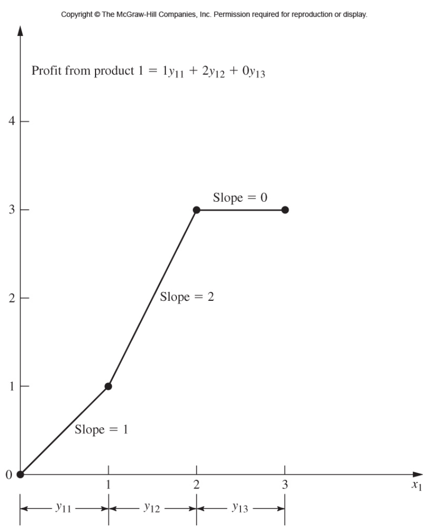
$y_{i2} \le y_{i1}$ and $y_{i3} \le y_{i2}$, for i = 1,2,3

Since y_{11}, y_{12}, y_{13} provide successive increments (if any) in the value of x_1 (starting from a value of 0), the coefficients of y_{11}, y_{12}, y_{13} are given by the respective *increments* in the product 1 column of Table 11.3 (1-0 = 1, 3-1 = 2, 3-3 = 0) (These correspond to $y_{11} - y_{10}$, $y_{12} - y_{11}$, $y_{13} - y_{12}$, respectively.)

33

Some Formulation Examples: Ex 2 (Violating Proportionality)

- These *increments* are the *slopes* in the figure, yielding $1y_{11}+2y_{12}+0y_{13}$ for the product 1 portion of the objective function.
- Note that applying this approach to all three products still must lead to a *linear* objective function.



34

Some Formulation Examples: Ex 2 (Violating Proportionality)

After we bring all variables to the left-hand side of the constraints, the resulting complete BIP model is:

Maximize $Z = y_{11} + 2y_{12} + 2y_{22} + y_{23} - y_{31} + 3y_{32} + 2y_{33}$

TABLE 11.3 Data for Example 2 (the Supersuds Corp. problem)

	Profit					
	Product					
Number of TV Spots	1		2		3	
0	0		0		0	
1	1	1	0		−1	-1
2	3	2	2	2	2	3
3	3	0	3	1	4	2

$y_{12} - y_{11} \leq 0$

$y_{13} - y_{12} \leq 0$

$y_{22} - y_{21} \leq 0$

$y_{23} - y_{22} \leq 0$

$y_{32} - y_{31} \leq 0$

$y_{33} - y_{32} \leq 0$

$y_{11} + y_{12} + y_{13} + y_{21} + y_{22} + y_{23} + y_{31} + y_{32} + y_{33} = 5$ and each y_{ij} is binary.

Solving this BIP model gives an optimal solution of

y_{11}=1, y_{21}=1, y_{31}=0, so x_1=2

y_{12}=0, y_{22}=0, y_{32}=0, so x_2=0

y_{13}=1, y_{23}=1, y_{33}=1, so x_3=3

Some Formulation Examples: Ex 3 (Covering All Characteristics)

- SOUTHWESTERN AIRWAYS needs to assign its crews to cover all its upcoming flights.
- We will focus on the problem of assigning three crews based in San Francisco to the flights listed in the first column of Table 11.4.

TABLE 11.4 Data for Example 3 (the Southwestern Airways problem)

	Feasible Sequence of Flights											
Flight	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
1. San Francisco to Los Angeles	1			1			1			1		
2. San Francisco to Denver		1			1			1			1	
3. San Francisco to Seattle			1			1			1			1
4. Los Angeles to Chicago				2			2		3	2		3
5. Los Angeles to San Francisco	2					3				5	5	
6. Chicago to Denver				3	3				4			
7. Chicago to Seattle							3	3		3	3	4
8. Denver to San Francisco		2		4	4				5			
9. Denver to Chicago					2			2			2	
10. Seattle to San Francisco			2				4	4				5
11. Seattle to Los Angeles						2			2	4	4	2
Cost, $1,000’s	2	3	4	6	7	5	7	8	9	9	8	9

Coefficients for Z

Some Formulation Examples: Ex 3 (Covering All Characteristics)

- The other 12 columns show the 12 feasible sequences of flights for a crew. (The numbers in each column indicate the order of the flights.)
- Exactly three of the sequences need to be chosen (one per crew) in such a way that every flight is covered. (It is permissible to have more than one crew on a flight, where the extra crews would fly as passengers, but union contracts require that the extra crews would still need to be paid for their time as if they were working.)
- The cost of assigning a crew to a particular sequence of flights is given (in thousands of dollars) in the bottom row of the table.
- The objective is to minimize the total cost of the three crew assignments that cover all the flights.

37

Some Formulation Examples: Ex 3 (Covering All Characteristics)

Formulation with Binary Variables.

- With 12 feasible sequences of flights, we have 12 yes-or-no decisions:
- Should sequence j be assigned to a crew? (j=1, 2, …,12)
- Therefore, we use 12 binary variables to represent these respective decisions:

$$x_j = \begin{cases} 1 & \text{if } sequence\ j\ is\ assigned\ to\ a\ crew \\ 0 & otherwise \end{cases}$$

38

Some Formulation Examples: Ex 3 (Covering All Characteristics)

- The most interesting part of this formulation is the nature of each constraint that ensures that a corresponding flight is covered.
- For example, consider the last flight in Table 11.4 [Seattle to Los Angeles (LA)].
- Five sequences (namely, sequences 6, 9, 10, 11, and 12) include this flight.
- Therefore, at least one of these five sequences must be chosen.
- The resulting constraint is

$$x_6 + x_9 + x_{10} + x_{11} + x_{12} \geq 1$$

Some Formulation Examples: Ex 3 (Covering All Characteristics)

- Using similar constraints for the other 10 flights, the complete BIP model is

Minimize $Z = 2x_1 + 3x_2 + 4x_3 + 6x_4 + 7x_5 + 5x_6 + 7x_7 + 8x_8 + 9x_9 + 9x_{10} + 8x_{11} + 9x_{12}$

subject to

$x_1 + x_4 + x_7 + x_{10} \geq 1$ (SF to LA)

$x_2 + x_5 + x_8 + x_{11} \geq 1$ (SF to Denver)

$x_3 + x_6 + x_9 + x_{12} \geq 1$ (SF to Seattle)

$x_4 + x_7 + x_9 + x_{10} + x_{12} \geq 1$ (LA to Chicago)

$x_1 + x_6 + x_{10} + x_{11} \geq 1$ (LA to SF)

$x_4 + x_5 + x_9 \geq 1$ (Chicago to Denver)

$x_7 + x_8 + x_{10} + x_{11} + x_{12} \geq 1$ (Chicago to Seatlle)

$x_2 + x_4 + x_5 + x_9 \geq 1$ (Denver to SF)

$x_5 + x_8 + x_{11} \geq 1$ (Denver to Chicago)

$x_3 + x_7 + x_8 + x_{12} \geq 1$ (Seattle to SF)

$x_6 + x_9 + x_{10} + x_{11} + x_{12} \geq 1$ (Seattle to LA)

Some Formulation Examples: Ex 3 (Covering All Characteristics)

- Using similar constraints for the other 10 flights, the complete BIP model is

subject to $\sum_{j=1}^{12} x_j = 3$ (assign three crews)

and

x_j is binary, for j= 1, 2, …, 12.

One optimal solution for this BIP model is

$x_3 = 1$ (assign sequence 3 to a crew)
$x_4 = 1$ (assign sequence 4 to a crew)
$x_{11} = 1$ (assign sequence 11 to a crew)

And all other $x_j = 0$, for a total cost of $18,000.
(Another optimal solution is $x_1 = 1$, $x_5 = 1$, $x_{12} = 1$, and all other $x_j = 0$.)

41

Set Covering vs. Set Partitioning Problems

- This example illustrates a broader class of problems called set covering problems.
- Any set covering problem can be described in general terms as involving a number of potential *activities* (such as flight sequences) and *characteristics* (such as flights).
- Each activity possesses some but not all of the characteristics.
- The objective is to determine the least costly combination of activities that collectively possess (cover) each characteristic at least once.
- Thus, let S_i be the set of all activities that possess characteristic i. At least one member of the set S_i must be included among the chosen activities, so a constraint,

$$\sum_{j \epsilon S_i} x_j \geq 1$$

is included for each characteristic i.

42

Set Covering vs. Set Partitioning Problems

- A related class of problems, called set partitioning problems, changes each such constraint to

$$\sum_{j \epsilon S_i} x_j = 1$$

- So, now *exactly* one member of each set S_i must be included among the chosen activities.
- For the crew scheduling example, this means that each flight must be included *exactly* once among the chosen flight sequences, which rules out having extra crews (as passengers) on any flight.

References

Hillier&Lieberman

- Chapter 12: Integer Programming 12.1-12.4.