Table 10·15

		$c_j \rightarrow$	2	20	−10	0	0
BASIC VARIABLES	C_B	X_B	X_1	X_2	X_3	X_4	G_1
x_2	20	0	0	1	0	0	1
x_1	2	10/3	1	0	2/3	0	−10/3
x_4	0	25/3	0	0	8/3	1	−40/3
	$z = 20/3$		0	0	34/3	0	40/3 ← Δ_j

Again, since the solution is non-integer one, insert one more fractional cut. From the third row of Table 10·15,

$$25/3 = 8/3\, x_3 + x_4 - 40/3\, g_1$$

or
$$(8 + 1/3) = (2 + 2/3)\, x_3 + (1 + 0)\, x_4 + (-14 + 2/3)\, g_1$$

The corresponding fractional cut will be $-1/3 = 0x_1 + 0x_2 - 2/3\, x_3 + 0x_4 - 2/3\, g_1 + g_2$

Inserting this constraint in Table 10·15, the following modified table is obtained.

Table 10·16

		$c_j \rightarrow$	2	20	−10	0	0	0
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2
x_2	20	0	0	1	0	0	1	0
x_1	2	10/3	1	0	2/3	0	−10/3	0
x_4	0	25/3	0	0	8/3	1	−40/3	0
g_2	0	→ −1/3	0	0	[−2/3]	0	−2/3	1
	$z = 20/3$		0	0	34/3 ↑	0	40/3	0 ← Δ_j ↓

Second Iteration. Using dual simplex method remove G_2 and introduce X_3.

Table 10·17

		$c_j \rightarrow$	2	20	−10	0	0	0
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2
x_2	20	0	0	1	0	0	1	0
x_1	2	3	1	0	0	0	−4	1
x_4	0	7	0	0	0	1	−16	4
x_3	−10	1/2	0	0	1	0	1	−3/2
	$z = 1$		0	0	0	0	2	17 ← Δ_j

Since the solution is still non-integer, a *third* fractional cut is required. From the last row of above table, we can construct the Gomorian constraint $-1/2 = -1/2\, g_2 + g_3$

Inserting this additional constraint in the above table, the modified simplex table becomes :

Table 10·18

		$c_j \rightarrow$	2	20	−10	0	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2	G_3	
x_2	20	0	0	1	0	0	1	0	0	
x_1	2	3	1	0	0	0	−4	1	0	
x_4	0	7	0	0	0	1	−16	4	0	
x_3	−10	1/2	0	0	1	0	1	−3/2	0	
g_3	0 →	−1/2	0	0	0	0	0	[−1/2]	1	
	$z = 1$		0	0	0	0	2	17 ↑	0 ↓	← Δ_j

Third Iteration. Using dual simplex method, remove G_3 and introduce G_2.

Table 10·19

		$c_j \rightarrow$	2	20	−10	0	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2	G_3	
x_2	20	0	0	1	0	0	1	0	0	
x_1	2	2	1	0	0	0	−4	0	2	
x_4	0	3	0	0	0	1	−16	0	8	
x_3	−10	2	0	0	1	0	1	0	−3	
g_2	0	1	0	0	0	0	0	1	−2	
	$z = -16$		0	0	0	0	2	0	34	$\leftarrow \Delta_j$

Thus an optimum integer solution is obained as : $x_1 = 2$, $x_2 = 0$, $x_3 = 2$, max. $z = -16$.

Example 4. *The owner of a ready-made garments store two types of shirts known as Zee-shirts and Button–down shirts. He makes a profit of Re. 1 and Rs. 4 per shirt on Zee-shirts and Button-down shirts respectively. He has two Tailors (A and B) at his disposal to stitch the shirts. Tailor A and Tailor B can devote at the most 7 hours and 15 hours per-day respectively. Both these shirts are to be stitched by both the tailors. Tailor A and Tailor B spend two hours and five hours respectively in stitching Zee-shirt, and four hours and three hours respectively in stitching a Button-down shirt. How many shirts of both the types should be stitched in order to maximize daily profit ?*

(a) Set-up and solve the linear programming problem.

(b) If the optimal solution is not integer-valued, use Gomory's technique to derive the optimal integer solution.

Formulation. Suppose the owner of ready-made garments decide to make x_1 Zee-shirts and x_2 Button-down shirts. Then the availability of time to tailors has the following restrictions :

$$2x_1 + 4x_2 \leq 7, \quad 5x_1 + 3x_2 \leq 15, \quad \text{and} \quad x_1, x_2 \geq 0.$$

The problem of the owner is to find the values of x_1 and x_2 to maximize the profit $z = x_1 + 4x_2$.

Solution. Introducing the slack variables $x_3 \geq 0$, $x_4 \geq 0$ in the constraints of the given problem, we have an initial basic feasible solution : $x_3 = 7$, $x_4 = 15$.

Computing the net-evaluations Δ_j and using simplex method an optimun solution is obtained as given in the following table.

Table 10·20

		$c_j \rightarrow$	1	4	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	
x_2	4	→ 7/4	1/2	1	1/4	0	
x_4	0	39/4	7/2	0	−3/4	1	
	$z = 7$		1	0	1	0	$\leftarrow \Delta_j$

Thus a non-integer solution is obtained as : $x_1 = 0$, $x_2 = \frac{7}{4}$, $x_4 = 39/4$, $z = 7$.

To find the integer valued solution, add a fractional cut constraint in the optimum simplex table. Since the fractional parts of $\mathbf{X_B}$ are $\left[\frac{3}{4}, \frac{3}{4}\right]$, select the row arbitrarily. So $f_{B1} = \frac{3}{4}$. Then from the first row of the Table 10·20 , we have

$$(1 + \tfrac{3}{4}) = (0 + \tfrac{1}{2})\, x_1 + (1 + 0)\, x_2 + (0 + \tfrac{1}{4})\, x_3 + (0 + 0)\, x_4$$

The corresponding fractional cut is therefore given by

$$-\tfrac{3}{4} = -\tfrac{1}{2} x_1 + 0x_2 - \tfrac{1}{4} x_3 + 0x_4 + g_1$$

Now inserting this additional constraint in the optimum simplex table, the modified table becomes.

Table 10·21

		$c_j \rightarrow$	1	4	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	
x_2	4	$1\frac{3}{4}$	1/2	1	1/4	0	0	
x_4	0	$9\frac{3}{4}$	7/2	0	− 3/4	1	0	
g_1	0→	− 3/4	[− 1/2]	0	− 1/4	0	1	
	$z = 7$		1 ↑	0	1	0	0 ↓	← Δ_j

First Iteration. Using dual simplex method, remove G_1 and insert X_1.

Table 10·22

		$c_j \rightarrow$	1	4	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	
x_2	4	1	0	1	0	0	1	
x_4	0	9/2	0	0	− 5/2	1	7	
x_1	1	→ 3/2	1	0	1/2	0	−2	
	$z = 11/2$		0	0	1/2	0	2	← Δ_j

Again, since the solution is non-integer one, insert another fractional cut in *Table* 10·22. From the third row of above table, we have $(1 + \frac{1}{2}) = (1 + 0)\, x_1 + (0 + 0)\, x_2 + (0 + \frac{1}{2})\, x_3 + (0 + 0)\, x_4 + (-2 + 0)\, g_1$

The corresponding fractional cut will be $-\frac{1}{2} = 0x_1 + 0x_2 - \frac{1}{2} x_3 + 0x_4 + 0 \cdot g_1 + g_2$.

Now inserting this additional constraint, the modified table becomes Table 10.23.

Table 10·23

		$c_j \rightarrow$	1	4	0	0	0		
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2	
x_2	4	1	0	1	0	0	1	0	
x_4	0	9/2	0	0	− 5/2	1	7	0	
x_1	1	3/2	1	0	1/2	0	−2	0	
← g_2	0	→− 1/2	0	0	[− 1/2]	0	0	1	
	$z = 11/2$		0	0	1/2 ↑	0	2	0 ↓	← Δ_j

Second Iteration. Using dual simplex method, remove G_2 and insert X_3.

Table 10.24

		$c_j \rightarrow$	1	4	0	0	0	0
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2
x_2	4	1	0	1	0	0	1	0
x_4	0	7	0	0	0	1	7	−5
x_1	1	1	1	0	0	0	−2	1
x_3	0	1	0	0	1	0	0	−2
	$z = 5$		0	0	0	0	2	1 ← Δ_j

This gives us an optimum integer solution : $x_1 = 1$, $x_2 = 1$, and max $z = 5$.

Thus the owner of ready-made garments should produce one Zee-shirt and also one Button-down shirt in order to get the maximum profit of Rs. 5.

Example 5. *A manufacturer of baby-dolls makes two types of dolls : Doll X and Doll Y. Processing of these two dolls is done on two machines A and B. Doll X requires two hours on machine A and six hours on machine B. Doll Y requires five hours on machine A and also five hours on machine B. There are sixteen hours of time per day available on machine A and thirty hours on machine B. The profit gained on both the dolls is same, i.e., one rupee per doll. What should be daily production of each of the two dolls ?*

(a) Set up and solve the linear programming problem.

(b) If the optimal solution is not integer valued, use Gomory's technique to derive the optimal solution.

Formulation of the problem. Suppose the manufacturer decides to produce x_1 dolls of type X and x_2 dolls of type Y. Then availability of time on two machines has the following restrictions :

$$2x_1 + 5x_2 \le 16, \quad 6x_1 + 5x_2 \le 30, \quad \text{and} \quad x_1, x_2 \ge 0.$$

The manufacturer wishes to determine the value of x_1 and x_2 so as to maximize the profit $z = \text{Rs.}\ (x_1 + x_2)$.

Solution. Introduce the slack variables $x_3 \ge 0$ and $x_4 \ge 0$ in the constraints of the given L.P. problem. An initial basic feasible solution is $x_3 = 16$ and $x_4 = 30$. Now using the simplex method, the optimum solution is obtained as given in the following table :

Table 10·25

		$c_j \rightarrow$	1	1	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	
x_2	1	$\rightarrow 9/5$	0	1	3/10	− 1/10	
x_1	1	7/2	1	0	− 1/4	1/4	
		$z = 53/10$	0	0	1/20	3/20	$\leftarrow \Delta_j$

This yields an optimum non-integer solution : $x_1 = \frac{7}{2}$, $x_2 = \frac{9}{5}$ and max $z = \frac{53}{10}$.

Since the fractional parts of X_B are $\left[\frac{4}{5}, \frac{1}{2}\right]$ and max $\left[\frac{4}{5}, \frac{1}{4}\right] = \frac{4}{5}$, therefore from the first row of above table,

$$(1 + \tfrac{4}{5}) = (0 + 0)\, x_1 + (1 + 0)\, x_2 + (0 + \tfrac{3}{10})\, x_3 + (-1 + \tfrac{9}{10})\, x_4$$

The corresponding fractional cut is given by

$$-\tfrac{4}{5} = 0x_1 + 0x_2 - \tfrac{3}{10} x_3 - \tfrac{9}{10} x_4 + g_1.$$

Now inserting this additional constraint into the optimum simplex table, the modified table becomes,

Table 10·26

		$c_j \rightarrow$	1	1	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	
x_2	1	9/5	0	1	3/10	− 1/10	0	
x_1	1	7/2	1	0	− 1/4	1/4	0	
g_1	0	− 4/5	0	0	− 3/10	[− 9/10]	1	
		$z = 53/10$	0	0	1/20	3/20 ↑	0 ↓	$\leftarrow \Delta_j$

First Iteration. Using *dual simplex method*, remove G_1 and introduce X_4.

Table 10·27

		$c_j \rightarrow$	1	1	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	
x_2	1	17/9	0	1	1/3	0	− 1/9	
x_1	1	59/18	1	0	− 1/3	0	5/18	
x_4	0	$\rightarrow$ 8/9	0	0	1/3	1	− 10/9	
		$z = 31/6$	0	0	0	0	1/6	$\leftarrow \Delta_j$

Since solution is still non-integer, insert one more fractional cut in the above table. From the third row of above table, we have

$$\tfrac{8}{9} = (0 + 0)\, x_1 + (0 + 0)\, x_2 + \left(0 + \tfrac{1}{3}\right) x_3 + (1 + 0)\, x_4 + (-2 + \tfrac{8}{9})\, g_1$$

The corresponding fractional cut becomes : $-\frac{8}{9} = 0x_1 + 0x_2 - \frac{1}{3}x_3 + 0x_4 - \frac{8}{9}g_1 + g_2$

Inserting this additional constraint, the modified table becomes :

Table 10·28

		$c_j \rightarrow$	1	1	0	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2	
x_2	1	17/9	0	1	1/3	0	− 1/9	0	
x_1	1	59/18	1	0	− 1/3	0	5/18	0	
x_4	0	8/9	0	0	1/3	1	− 10/9	0	
g_2	0	− 8/9	0	0	[− 1/3]	0	− 8/9	1	
		$z = 31/6$	0	0	0 ↑	0	1/6	0 ↓	$\leftarrow \Delta_j$

Second Iteration. Using *dual simplex method*, remove G_2 and insert X_3.

Table 10·29

		$c_j \rightarrow$	1	1	0	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2	
x_2	1	1	0	1	0	0	− 1	1	
x_1	1	25/6	1	0	0	0	7/6	− 1	
x_4	0	0	0	0	0	1	− 2	1	
x_3	0	→ 8/3	0	0	1	0	8/3	− 3	
		$z = 31/6$	0	0	0	0	1/6	0	$\leftarrow \Delta_j$

This solution is also non-integer one, so insert one more fractional cut. The fractional parts of X_B are $\left[\frac{1}{6}, \frac{2}{3}\right]$ and max $\left[\frac{1}{6}, \frac{2}{3}\right] = \frac{2}{3}$. Therefore, from the last row of the above table, we have

$$(2 + \tfrac{2}{3}) = (0 + 0)\,x_1 + (0 + 0)\,x_2 + (1 + 0)\,x_3 + (0 + 0)\,x_4 + (2 + \tfrac{2}{3})\,g_1 + (-3 + 0)\,g_2$$

The corresponding fractional cut will be

$$-\frac{2}{3} = 0x_1 + 0x_2 + 0x_3 + 0x_4 - \frac{2}{3}g_1 + 0g_2 + g_3 .$$

Now inserting this constraint, the modified table becomes :

Table 10·30

		$c_j \rightarrow$	1	1	0	0	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2	G_3	
x_2	1	1	0	1	0	0	− 1	1	0	
x_1	1	25/6	1	0	0	0	7/6	− 1	0	
x_4	0	0	0	0	0	1	− 2	1	0	
x_3	0	8/3	0	0	1	0	8/3	− 3	0	
g_3	0	− 2/3	0	0	0	0	[− 2/3]	0	1	
		$z = 31/6$	0	0	0	0	1/6 ↑	0	0 ↓	$\leftarrow \Delta_j$

Third Iteration. Using *dual simplex method*, remove G_3 and introduce G_1.

Table 10·31

		$c_j \rightarrow$	1	1	0	0	0	0	0	
BASIC VAR.	C_B	X_B	X_1	X_2	X_3	X_4	G_1	G_2	G_3	
x_2	1	2	0	1	0	0	0	1	− 3/2	
x_1	1	3	1	0	0	0	0	− 1	7/4	
x_4	0	2	0	0	0	1	0	1	− 3	
x_3	0	0	0	0	1	0	0	− 3	4	
g_1	0	1	0	0	0	0	1	0	− 3/2	
		$z = 5$	0	0	0	0	0	0	1/4	$\leftarrow \Delta_j$

This gives the optimum integer solution : $x_1 = 3$, $x_2 = 2$ and max $z = 5$.

Thus, the manufacturer should produce 3 dolls of type *X*, 2 dolls of type *Y* in order to get the maximum profit of Rs. 5.

Note. Alternative solutions are : $x_1 = 5$, $x_2 = 0$; and $x_1 = 4$, $x_2 = 1$.

10.6. GEOMETRICAL INTERPRETATION OF GOMORY'S CUTTING PLANE METHOD

The geometrical interpretation of cutting plane method can be easily understood through a practical example.

Let us consider the problem of ***Example 5*** :

Max.
$z = x_1 + x_2$, s.t. $2x_1 + 5x_2 \le 16$, $6x_1 + 5x_2 \le 30$, $x_1, x_2 \ge 0$.

The graphical solution of this problem is obtained in *Fig.* 10·3 with solution space represented by the convex region *OABC*. The optimum solution occurs at the extreme point *B*, *i.e.*

$$x_1 = 3.5,\ x_2 = 1.8,\ \max z = 5.3.$$

But, this solution is not integer-valued. While solving this problem by Gomory's method, we introduced the first Gomory's constraint : $-\frac{3}{10}x_3 - \frac{9}{10}x_4 \le -\frac{4}{5}$ ...(i)

In order to express this constraint in terms of x_1 and x_2, we make use of the constraint equations : $2x_1 + 5x_2 + x_3 = 16$ and $6x_1 + 5x_2 + x_4 = 30$,
where x_3 and x_4 are slack variables. From these, we get

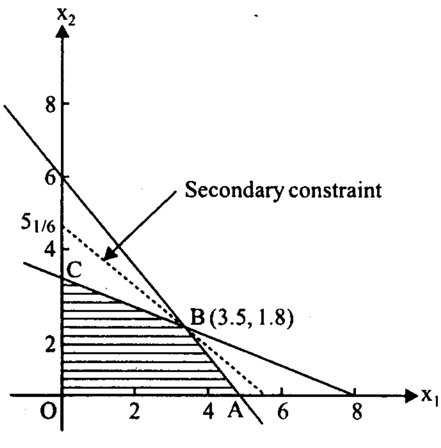


Fig. 10.3

$$x_3 = 16 - 2x_1 - 5x_2 \text{ and } x_4 = 30 - 6x_1 - 5x_2,$$

The *Gomory's* constraint (i) then becomes

$$-\tfrac{3}{10}(16 - 2x_1 - 5x_2) - \tfrac{9}{10}(30 - 6x_1 - 5x_2) \le -\tfrac{4}{5}, \quad i.e. \quad x_1 + x_2 \le 5\tfrac{1}{6}.$$

This constraint cuts off the feasible region and now the feasible region is reduced to somewhat less than the previous one as shown in *Fig.* 10·3 .

Similarly, the second *Gomory's* constraint is $g_1 \ge 1$. But,

$$-\tfrac{3}{10}x_3 - \tfrac{9}{10}x_4 + g_1 = -\tfrac{4}{5}, \quad i.e. \quad g_1 = (\tfrac{3}{10}x_3 + \tfrac{9}{10}x_4) - \tfrac{4}{5}$$

Substituting the values of x_3 and x_4 from the constraint equations of the given problem, we immediately get $g_1 = 31.8 - 6x_1 - 6x_2$. Therefore, $31{\cdot}8 - 6x_1 - 6x_2 \ge 1$ ($\because g_1 \ge 1$) or $x_1 + x_2 \le 5{\cdot}103$.

This constraint also cuts off some space of the feasible region. Since this constraint very minutely cuts off the solution space, so it has not been plotted on the graph. Because of such cuttings, this method was named as *cutting plane method.*

EXAMINATION PROBLEMS

Find the optimum integer solution of the following all integer programming problems :

1. Max. $z = x_1 + x_2$, subject to
   $3x_1 - 2x_2 \le 5$
   $x_1 \le 2$
   $x_1 \ge 0$, $x_2 \ge 0$, and are integers.
   [Meerut M.Sc. (Math.) 94]
   [Ans. $x_1 = 3$, $x_2 = 2$, max $z = 5$]

2. Max $z = x_1 - 2x_2$, subject to
   $4x_1 + 2x_2 \le 15$
   $x_1 >< = 0$, $x_2 \ge 0$, and integers

3. Max $z = 3x_2$, subject to
   $3x_1 + 2x_2 \le 7$
   $x_1 - x_2 \ge -2$

4. Max $z = x_1 + 5x_2$, subject to
   $x_1 + 10x_2 \le 20$
   $x_1 \le 2$

x_1 , $x_2 \geq 0$ and integers.
[Hint. Simplex method gives the integer solution.]
[Ans. $x_1 = 0$, $x_2 = 2$, max $z = 6$]

x_1 , $x_2 \geq 0$ and integers.
[Ans. $x_1 = 2$, $x_2 = 1$, max $z = 7$]

5. Max $z = 2x_1 + 2x_2$, subject to the constraints :
$5x_1 + 3x_2 \leq 8$
$x_1 + 2x_2 \leq 4$
x_1 , $x_2 \geq 0$ and are integers.
[Agra 99]
[Ans. $x_1 = 1$, $x_2 = 1$, and max $z = 4$]

6. Max. $z = 4x_1 + 3x_2$, subject to the constraints
$x_1 + 2x_2 \leq 4$
$2x_1 + x_2 \leq 6$
x_1 , $x_2 \geq 0$ and are integers.
[Agra 99]
[Ans. $x_1 = 3$, $x_2 = 0$, max $z = 12$]

7. Max $z = 3x_1 + 4x_2$, subject to the constraints :
$3x_1 + 2x_2 \leq 8$
$x_1 + 4x_2 \geq 10$
x_1 , $x_2 \geq 0$ and are integers.
[Ans. $x_1 = 0$, $x_2 = 4$, max. $z = 16$]

8. Max. $z = 11x_1 + 4x_2$, subject to the constraints :
$-x_1 + 2x_2 \leq 4$
$5x_1 + 2x_2 \leq 16$
$2x_1 - x_2 \leq 4$
$x_1 \geq 0$, $x_2 \geq 0$ and are integers.
[Meerut M.Sc.93]
[Ans. $x_1 = 2$, $x_2 = 3$, max $z = 34$]

9. Max $z = x_1 - x_2$, subject to the constraints :
$x_1 + 2x_2 \leq 4$,
$6x_1 + 2x_2 \leq 9$;
x_1 , $x_2 \geq 0$, and are integers.
[Meerut M.Sc. (Math) 92]
[Ans. $x_1 = 1$, $x_2 = 0$; max. $z = 2$]

10. Max. $z = 3x_1 - 2x_2 + 5x_3$, subject to the constraints.
$5x_1 + 2x_2 + 7x_3 \leq 28$
$4x_1 + 5x_2 + 5x_3 \leq 30$
x_1 , $x_2 < x_3 \geq 0$ and are integers.
[Hint. Simplex method gives the integer solution]
[Ans. $x_1 = 0$, $x_2 = 0$, $x_3 = 4$, max. $z = 20$

II–Branch and Bound Method

10.7. THE BRANCH-AND-BOUND METHOD

This section deals with the algorithm given by *Land* and *Doig* for solving the ***all-integer*** and ***mixed-integer*** problems. Why this method is given the name '*branch-and-bound*' will be made clear in the following sections. This is the most general technique for the solution of an I.P.P. in which *a few* or *all* the variables are constrained by their upper or lower bounds or by both. This technique is now discussed below.

The general idea of the method is to solve the problem first as a continuous linear programming problem and then the original problem is partitioned (branched) into two sub-problems by imposing the integer conditions on one of its integer variables that currently has a fractional optimal value. Let x_j be an integer-constrained variable whose optimum continuous value x_j* is obtained in terms of a fraction. Then clearly we shall have,

$$[x_j*] \leq x_j \leq [x_j*] + 1 \, .$$

Any feasible integer value, therefore, must satisfy one of the two conditions :

$$x_j \leq [x_j*] \quad \text{or} \quad x_j \geq [x_j*] + 1 \, .$$

These two constraints are mutually exclusive and thus cannot be true simultaneously and hence both cannot be introduced in the integer programming problem simultaneously. By introducing these constraints one by one in the continuous linear programming problem, we shall have two sub-problems, both being integer-valued.

After branching in this manner, two sub-problems are constructed by inserting $x_j \leq [x_j*]$ and $x_j \geq [x_j*] + 1$ one by one to the original set of constraints.

To be definite, let the mixed I.P.P. be :

$$\text{Max. } z = \sum_{j=1}^{n} c_j x_j \text{, subject to the constraints :} \qquad ...(10{\cdot}6)$$

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \text{, for } i = 1, 2, \dots, m \qquad ...(10{\cdot}7)$$

where x_j is integer valued for $j = 1, 2, \dots, k \; (\leq n)$, ...(10·8)

and $x_j \geq 0$ for $j = 1, 2, ..., k, k+1, \dots, n$ . ...(10·9)

In addition to above, let us assume that for each integer-valued variable x_j lower and upper bounds can be assigned so that these bounds surely contain the optimal values

$$L_j \leq x_j \leq U_j \text{ for } j = 1, 2, \ldots, k. \qquad \ldots(10\cdot10)$$

The following principal idea is behind the *'branch-and-bound technique* we are looking for :

Let us consider any variable x_j and let I be some integer value such that $L_j \leq I \leq U_j - 1$. Then an optimum solution to the problem (10·6) through (10·9) also satisfies

either *the linear constraint* $x_j \geq I + 1$ ...(10·11)

or *the linear constraint* $x_j \leq I$. ...(10·12)

To explain how this partitioning helps us, suppose we have overlooked the integer condition (10·8) and obtained an optimal solution to the L.P.P. consisting of (10·6), (10·7), (10·8) and (10·9) indicating $x_1 = 1\frac{3}{5}$ (for example). Obviously, $x_1 = 1\frac{3}{5}$ gives the range $1 < x_1 < 2$. Therefore, in an integer-valued solution, we must have either $x_1 \leq 1$ or $x_1 \geq 2$.

Thus there will be no integer valued feasible solution in the region $x_1 = 1$ to $x_1 = 2$ as shown in the following figure.

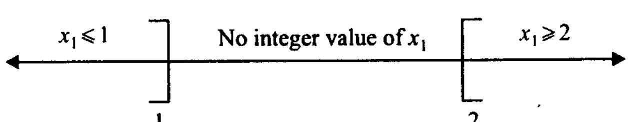


Fig. 10.4

Now our problem is to search for the optimum value of z either in the first region ($x_1 \leq 1$) or in the second region ($x_1 \geq 2$).

Thus we formulate and solve the following two sub-problems separately :

Sub-problem (1) : consisting of **(10·6), (10·7), (10·8) and $2 \leq x_1 \leq U_1$**

Sub-problem (2) : consisting of **(10·6), (10·7), (10·8), and $L_1 \leq x_1 \leq 1$.**

If for any one of the sub-problems, optimum integer solution is obtained then that problem is not partitioned further. Sometimes, it may also be possible that the sub-problem has no solution at all. Such sub-problem is also discarded for ever. But, if any *sub-problem* involves some non-integer variable, then it is again partitioned and this process of partitioning continues so long as it is applicable until each sub-problem either possesses an integer-valued optimum solution or there is an indication that it cannot provide a better solution. The optimum integer-valued solution among all the sub-problems is finally selected which gives overall optimum value of the objective function.

We now discuss below the step-by-step procedure that specifies how the partitioning (10·11) and (10·12) can be applied systematically to eventually get an optimum integer-valued solution.

Q. Explain the Branch and Bound principle used in I.P.

10·7-1 . Branch-and-Bound Algorithm

At the rth iteration we have available a *lower bound* (say, z_r) for the optimal value of the objective function. For convenience, we suppose that at the first iteration, z_1 is either strictly less than the optimal value, or equals the value of the objective function for a feasible solution that we have noted. In case, if we have no information about the problem we let $z_1 = -\infty$. In addition to a lower bound z_1 we also have a master list of linear programming problems to be solved differing only in the revision of the bounds (10·10). At the first iteration, the master list has only one problem consisting of (10·6), (10·7), (10·8) and (10·10).

The step-by-step procedure at this rth ($r = 0, 1, 2, \ldots$) iteration can be outlined as follows :

Step 1. Two possibilities may arise at the rth iteration :

(i) If the master list does not contain any linear programming problem (*i.e.*, empty), stop the computations.

(ii) Otherwise, go to ***step* 2** for removing a linear programming problem from the master list.

Step. 2. Solve the chosen problem to obtain the optimum solution by using bounded variable technique. Again, two possibilities may arise :

(i) If it has no feasible solution, or if the resulting optimal value of the objective function z is $\leq z_r$, then let $z_{r+1} = z_r$ and return back to ***step 1.***

(ii) Otherwise, go to ***step 3.***

Step 3. *(i)* If the optimal solution to the linear programming problem thus obtained satisfies the integer condition, then record it, let z_{r+1} be associated optimal value of the objective function, and return back to *step 1.*

(ii) Otherwise, go to ***step 4.***

Step 4. Select any variable x_j, for $j = 1, 2, \ldots, k$, that does not have an integer value in the obtained optimal solution to the chosen linear programming problem. Let x_j* denote this value, and $[x_j*]$ stand for largest integer less than or equal to x_j*. Now, include two linear programming problems in the master list. These two sub-problems are :

Sub-prob 1. Same as the problem chosen in ***step 1,*** except that the lower bound L_j on x_j is replaced by $[x_j*]+1$.

Sub-prob 2. Same as the problem chosen in ***step 1,*** except that the upper bound U_j on x_j is replaced by $[x_j*]$.

Let $z_{r+1} = z_r$, and return back to ***step 1.*** At the termination of the process if we find a integer-valued feasible solution giving z_r, it will be optimal, otherwise no integer-valued feasible solution exists.

Q. 1. Describe any one method of solving mixed integer programming problem.

2. Sketch the branch-and-bound method in integer programming. **[Agra 99]**

3. What is the main disadvantage of the branch and bound method ?

4. Explain with an example, how in some cases non-integer solution to a linear programming problem is meanigless.

10·7-2. Computational Demonstration of Branch-and-Bound Method

The computational procedure of *Branch-and-Bound* algorithm is now explained below by solving a numerical example.

Example 6. *Use Branch-and-Bound technique to solve the following integer programming problem :*

$$\text{Max. } z = 7x_1 + 9x_2 \quad \ldots(1)$$

$$\text{subject to} \quad -x_1 + 3x_2 \le 6$$

$$7x_1 + x_2 \le 35 \quad \ldots(2)$$

$$(0 \le x_1, x_2 \le 7) \quad \ldots(3)$$

$$\text{and } x_1, x_2 \text{ are integers.} \quad \ldots(4)$$

[Agra 98; Banasthali (MSc) 93; Bharthidasan B.Sc. (Math.) 90]

Solution. Step 1. At the initial iteration, we take $z^{(1)} = 0$ as the lower bound for z, since the solution $x_1 = x_2 = 0$ is feasible. The master list contains only the linear programming problem [(1), (2), (3)] which will be named as ***Sub-prob. 1.***

Step 2. Using graphical method, determine the optimal solution of ***Sub-prob.* 1** as $x_1 = 9/2$, $x_2 = 7/2$, $z* = 63$. Since the solution is not integer-valued, go to ***step 3,*** and choose x_1. Since $[x_1*] = [9/2] = 4$ add the following two sub-problems in the master list :

Sub-prob. 2 : (1), (2) and $5 \le x_1 \le 7$, $0 \le x_2 \le 7$

Sub-prob. 3 : (1), (2) and $0 \le x_1 \le 4$, $0 \le x_2 \le 7$.

Returning to first step with $z^{(2)} = z^{(1)} = 0$, we select the ***Sub-prob.*** 2. Now, *step* 2 determines that *Sub-prob* 2 has the feasible solution

$$x_1 = 5, x_2 = 0, z* = 35, \text{ (Solution of Sub-prob. 2.)} \quad \ldots(5)$$

Clearly, this solution satisfies the integer constraints. So we record it at this step, and take $z^{(3)} = 35$.

Again returning to *step 1* with $z^{(3)} = 35$, we have *Sub-Prob.* 3.

Step. 2. Immediately gives the optimum feasible solution to ***Sub-prob. 3*** as

$$x_1 = 4, x_2 = 10/3, z* = 58. \text{ [Solution to Sub-prob. 3]}$$

Since this solution is not integer-valued, go to *step 3.*

Step 3. Now consider x_2. Since $[x_2*] = [3\frac{1}{3}] = 3$, we add the following sub-problems to the master list :

Sub-prob. 4. (1), (2) and $0 \le x_1 \le 4$, $4 \le x_2 \le 7$

Sub-prob. 5. (1), (2) and $0 \le x_1 \le 4$, $0 \le x_2 \le 3$.

Returning to *step 1* with $z^{(4)} = z^{(3)} = 35$, we select the *Sub-Prob. 4*. In *Step 2* we find that *Sub-Prob. 4* has no feasible solution. So we, again, return to *Step 1* with $z^{(5)} = z^{(4)} = 35$. Only *Sub-Prob. 5* is now available on the master list. Using *step 2,* we obtain the optimum solution to *Sub-Prob. 5.*

$$z* = 55 , x_1 = 4 , x_2 = 3 . \text{[Solution to Sub-prob. 5]} \quad ...[6]$$

Clearly, this solution satisfies the integer conditions. So we record it at *step 3,* and let $z^{(6)} = 55$.
Again returning to *Step 1,* the master list becomes empty (*i.e.,* contains no sub-problem) and thus the process ends.
At the time of ending the process, we observe that only two feasible integer solutions (*5*) and (*6*) have been noted. The 'best one' of these two feasible integer solutions gives us the required optimum solution to the given integer programming problem.

Thus, finally, we get the optimum solution to the given I.P.P. as $z* = 55$, $x_1 = 4$, $x_2 = 3$.
The tree-diagram corresponding to this problem is shown in the following figure.
The entire calculations of this tree-diagram may be summarised as shown in the following table.

Tree-Diagram of *Example* 6

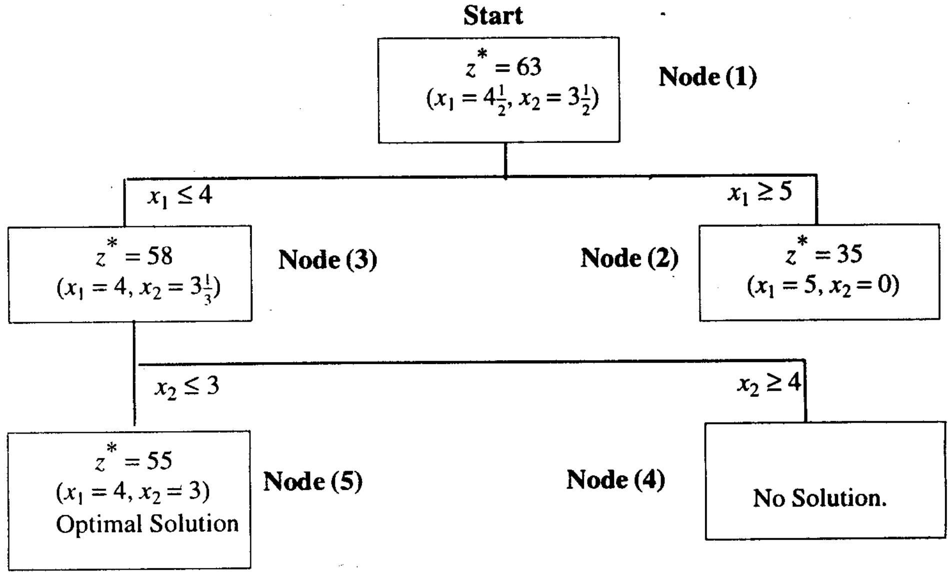


Fig. 10.5

Node	Solution			Additional Constraints	Type of solution
	x_1	x_2	z^*		
(1)	9/2	7/2	63	—	Non-integer (Original problem)
(2)	5	0	35	$x_1 \geq 5$	Integer ← $z*^{(1)}$
(3)	4	10/3	58	$x_1 \leq 4$	Non-integer
(4)	...	...	...	$x_1 \leq 4 , x_2 \geq 4$	No Solution
(5)	4	3	55	$x_1 \leq 4 , x_2 \leq 3$	Integer ← $z^{*(2)}$ (optimal)

Example 7. *Use Branch-and-Bound technique to solve the following problem.*

$$Max.\ z = 3x_1 + 3x_2 + 13x_3, \quad \textit{subject to} \quad ...(1)$$

$$\begin{Bmatrix} -3x_1 + 6x_2 + 7x_3 \le 8 \\ 6x_1 - 3x_2 + 7x_3 \le 8 \end{Bmatrix} \quad \text{...(2)}$$

$$0 \le x_j \le 5\,, \quad \text{...(3)}$$

and x_j are integers, for $j = 1, 2, 3$. ...(4)

Solution. First we find the optimal solution by inspection.

Iteration 1 :

Step 1. At the initial iteration, let the lower bound of z be $z^{(1)} = 0$, then $x_1 = x_2 = x_3 = 0$ is feasible. The master list consists of only the L.P.P. (1), (2) and (3), which is designated as *Sub-Problem 1*. Remove it in the *step 2*.

Step 2. Find the optimal solution of ***Sub-prob.*** *1* as $z^* = 16$, $x_1 = x_2 = 2\frac{2}{3}$, $x_3 = 0$.

Since the solution is not integer-valued, we proceed from *step 2* to *step 3,* and choose x_1 .

Step 3. Since $[x_1*] = [2\frac{2}{3}] = 2$, add the following two problems in the master list :

Sub-prob 2 : (1), (2), and $3 \le x_1 \le 5$, $0 \le x_2 \le 5$,$0 \le x_3 \le 5$.

Sub-prob 3 : (1), (2) and $0 \le x_1 \le 2$, $0 \le x_2 \le 5$, $0 \le x_3 \le 5$.

Iterations 2 and 3 :

Returning to *step 1* with $z^{(2)} = z^{(1)} = 0$, we remove *Sub-Prob 2.* It can be verified that *step 2* gives no feasible solution to *Sub-Prob 2.* Hence, put $z^{(3)} = z^{(2)} = 0$, and return to *step 1.*

In order to remove *Sub-Prob. 3* we obtain its optimal soluton in *step 2* as

$$x_1 = x_2 = 2\,,\ x_3 = \frac{2}{7}\,,\ z* = 15\frac{5}{7}\,. \qquad \textbf{\textit{(Sol. of Sub-prob. 3)}}$$

Clearly, this solution is not integer-valued. So, we proceed from *step 2* to *step 4.*

Step 4. Since $[x_3^*] = [\frac{2}{7}] = 0$, and therefore include two sub-problems in the master list :

Sub-prob. 4 : (1), (2), and $1 \le x_1 \le 2$, $0 \le x_2 \le 5$, $1 \le x_3 \le 5$.

Sub-prob. 5 : (1), (2), and $0 \le x_1 \le 2$, $0 \le x_2 \le 5$, $0 \le x_3 \le 0$.

Here we observe that *Sub-Prob. 4* and *Sub-Prob. 5* differ from *Sub- Prob. 3* only in the bounds on x_3 .

Iteration 4 :

Now returning to *step 1* with $z^{(4)} = 0$, we remove *Sub-Prob 4.* The optimal solution is thus obtained as

$$x_1 = x_2 = \frac{1}{3}\,,\ x_3 = 1\,,\ z* = 15\,. \qquad \textbf{\textit{(Sol. of Sub-prob. 4)}}$$

This leads to *step 4* again; let us select x_2 , yielding, as a consequence, the following two sub-problems for including in the mater list.

Sub-prob. 6 : (1), (2) and $0 \le x_1 \le 2$, $1 \le x_2 \le 5$, $1 \le x_3 \le 5$.

Sub-prob. *7.* : (1), (2), and $0 \le x_1 \le 2$, $0 \le x_2 \le 0$, $1 \le x_3 \le 5$.

It is obvious that *Sub-prob. 6* and *Sub-prob. 7* differ from *Sub-prob. 4* only in the bounds on x_2 .

Iteration 5 :

Now returning to *step 1* with $z^{(5)} = 0$, we remove *Sub-prob. 6* and check that *Sub-prob. 5* and *Sub-prob. 7* still remain on the master list. We can find in *step 2* that *Sub-prob. 6* has no feasible solution.

Iteration 6 :

So we return to *step 1* with $z^{(6)} = 0$. We now remove *Sub-prob.* 7 whose optimal solution is obtained as

$$x_1 = x_2 = 0\,,\ x_3 = 1\frac{1}{7}\,,\ z* = 14\frac{6}{7} \qquad \textbf{\textit{(Sol. of Sub-prob.7)}}$$

Since x_3 is fractional, we again repeat the *step 4.* Let us select x_3 . Here $[x_3*] = [1\frac{1}{7}] = 1$. So we add two more problems in the master list.

Sub-prob. 8 : (1), (2), and $0 \le x_1 \le 2$, $0 \le x_2 \le 0$, $2 \le x_3 \le 5$

Sub-prob. 9 : (1), (2), and $0 \le x_1 \le 2$, $0 \le x_2 \le 0$, $1 \le x_3 \le 1$.

Iterations 7, 8, 9 :

It can be easily verified that removal of *Sub-prob.* 8 at iteration 7 provides an indication of no feasible solution in *step 2,* and removal of *Sub-prob.* 9 at the 8th iteration yields in *step 2* :

$$x_1 = x_2 = 0\,,\ x_3 = 1\,,\ z^* = 13\,. \qquad \textbf{(Sol. of Sub-prob. 9)}$$

Therefore, at *step 3,* we record this optimal solution and let $z^{(9)} = 13$.

Returning to *step 1* again, we observe that only *Sub-prob.* 5 is now left on the master list whose optimal solution is :

$$x_1 = 2\,,\ x_2 = 2\tfrac{1}{3}\,,\ x_3 = 0\,,\ z^* = 13\,. \qquad \textbf{(Sol. of Sub-prob. 5)}$$

Since the value of objective function in the solution of *Sub-prob.* 9 and *Sub-prob.* 5 is the same and is equal to $z^* = 13$, we return to *step 1* and stop the computations because the master list is now empty.

Thus, finally, we get the optimal solution to the integer programming problem as recorded at the 8th iteration : $x_1 = x_2 = 0\,,\ x_3 = 1\,,\ z^* = 13$.

Tree-Diagram of Example 7

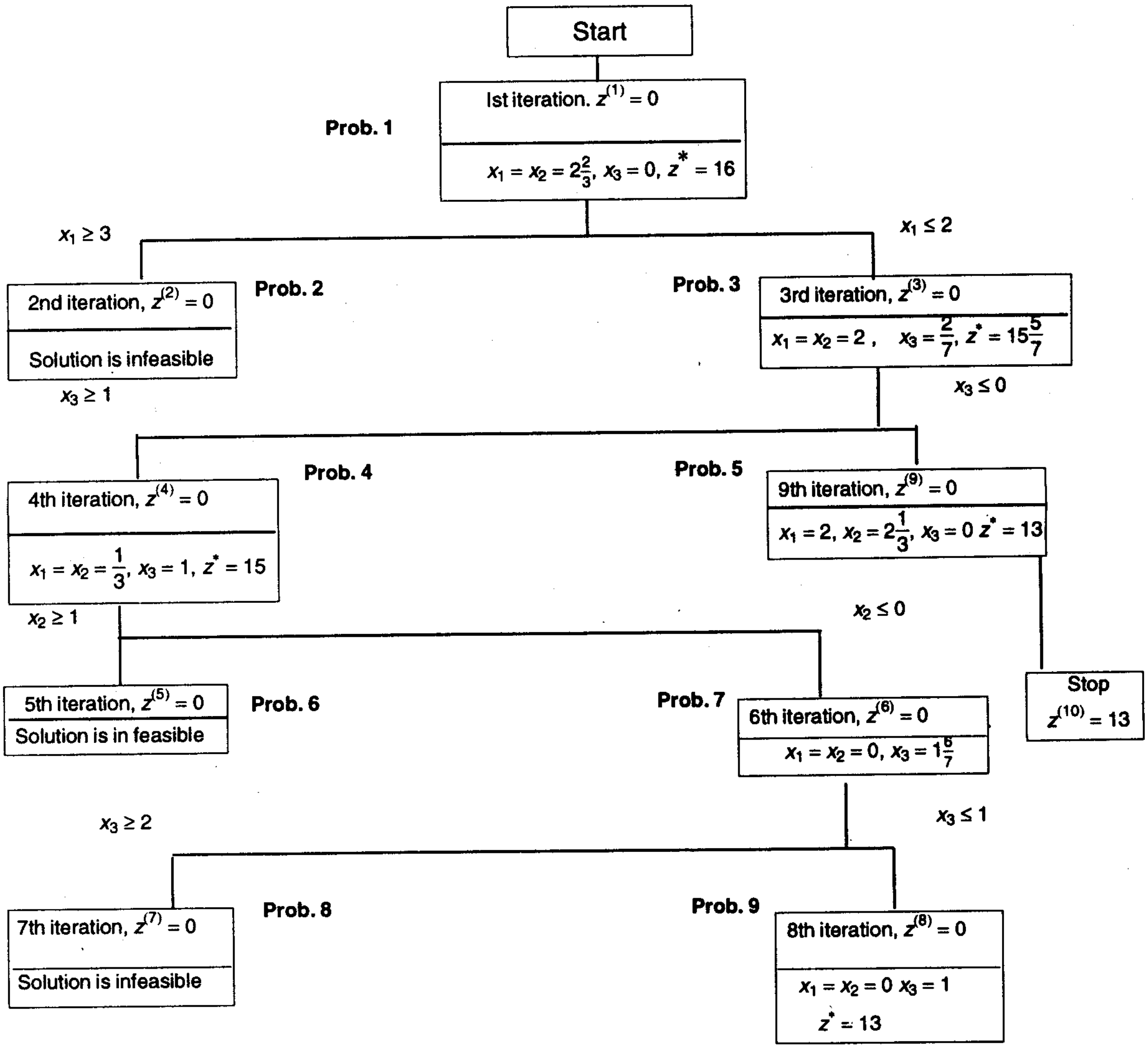


Fig. 10.6

Remarks :

1. In the solution of above problem we have made arbitrary choices in the algorithm at two places :
   (i) Selection of the problem to remove in *Step 1.*
   (ii) Selection of the variable x_j to give us additional problem in *Step 4*. The number of iterations required to solve a problem can vary considerably depending on how these selections are actually done. For example, choice of *Prob. 4* instead of *Prob. 5* at the 4th iteration turned-out to be auspecious. Although, auxiliary numerical tests have been developed to help us in making these choices, but these are not discussed in this text, because they are specially useful to technical specialists.
2. The above algorithm can be demonstrated by means of a tree-like diagram as shown in Fig 8·5 and 8·6 . We have noted that each node in the tree diagram represents a problem on the master list, each branch is leading to one of the problems added to the master list in *Step 4*. On account of this graphical analogy the word 'branch' is used in the name of the algorithm 'Branch-and-Bound'. The word 'bound' is suggested by the test in *Step 2.*

10.8. GEOMETRICAL INTERPRETATION OF BRANCH-AND-BOUND METHOD

The geometrical interpretation of *Branch-and-Bound* method can be easily understood by the following practical example.

Example 8. *Explain the geometrical interpretation of Branch-and-Bound method by solving the following I.P.P. :*

Max. $z = x_1 + x_2$, *subject to the constraints :* $3x_1 + 2x_2 \leq 12$, $x_2 \leq 2$, $x_1 \geq 0$, $x_2 \geq 0$ ***and are integers.***

Solution. Step 1. *To solve the problem by graphical methuod without intehger conditions.*

The graphical solution of ***Sub-problem 1*** : Max. $z = x_1 + x_2$, subject to

$3x_1 + 2x_2 = 12$, $x_2 \leq 2$, x_2, $x_2 \geq 0$ is shown by the convex region OABC in Fig. 10·7. The optimum solution occurs at the extreme point $B(x_1 = 8/3, x_2 = 2)$ with max. $z = 14/3$.

Step 2. Since the solution obtained above is not integer-valued, the given linear programming problem isbranched into two *sub-problems* as follows :

The non-integer value of $x_1 = 8/3$ gives the range $2 < 8/3 < 3$. Thus, two sub-problems are stated as follows :

Sub-prob 2 : Max. $z = x_1 + x_2$, s.t. $3x_1 + 2x_2 \leq 12$, $0 \leq x_2 \leq 2$, $0 \leq x_1 \leq 2$

Sub-prob 3 : Max $z = x_1 + x_2$, s.t. $3x_1 + 2x_2 \leq 12$, $0 \leq x_2 \leq 2$, $x_1 \geq 3$.

The optimum solution of **sub-problem 2** is : $x_1 = 2$, $x_2 = 2$ and max. $z = 4$ as shown in Fig. 10·8. while the optimum solution of **sub-problem 3** is : $x_1 = 3$, $x_2 = 3/2$ and max. $z = 9/2$ as shown in Fig. 10.9.

In **sub-problem 2**. all the variables have integer values. So there is no need of further sub-division. But, **sub-problem-3** having non-integer solution needs further sub-division.

Step 3. In sub-problem-3, the non-integer value of $x_2 = 3/2$ gives the range $1 < x_2 < 2$. So we contruct two more sub-problems by adding the constraints $x_2 \leq 1$ and $x_2 \geq 2$ one by one in ***sub-problem 3.*** Thus two additional sub-problems are :

Sub-problem 4 : Max. $z = x_1 + x_2$ s.t. $3x_1 + 2x_2 \leq 12$, $0 \leq x_2 \leq 2$, $x_1 \geq 3$, $0 \leq x_2 \leq 1$.

Sub-problem 5 : Max. $z = x_1 + x_2$, s.t. $3x_1 + 2x_2 \leq 12$, $0 \leq x_2 \leq 2$, $x_1 \leq 3$, $0 \leq x_2 \geq 2$.

In ***sub-problem - 4,*** the constraint $x_2 \leq 2$ is redundant. The optimal solution to this sub-problem is obtained as $x_1 = 10/3$, $x_2 = 1$, and max. $z = 13/3$ as shown in Fig. 10·10. This solution is not integer valued.

Here it is clear that any further branching of **sub-problem 4** will not improve the value of the objective function because the next sub-division will impose the restrictions $x_1 \leq 3$ and $x_1 \geq 4$. then the optimum solutions are obtained as $(x_1 \approx 3, x_2 = 1)$ and $(x_1 = 4, x_2 = 0)$ respectively. Both of these solutions give the maximum value of z equal to 4.

Further, it may be moted that there exists no feasible solution to *sub-problem* 5.

Step 4. Finally, maximum value of the objective function z is obtained as 4 and the integer valued solution is any of the following three :

$$(x_1 = 2, x_2 = 2) \quad \text{or} \quad (x_1 = 3, x_2 = 1) \quad \text{or} \quad (x_1 = 4, x_2 = 0)$$

EXAMINATION PROBLEMS

Use Branch-and-Bound technique to solve the following problems.

1. Max $z = 3x_1 + 3x_2 + 13x_3$
subject to
$-3x_1 + 6x_2 + 7x_3 \leq 8$
$5x_1 - 3x_2 + 7x_3 \leq 8$
$0 \leq x_j \leq 5$
and all x_j are integers.

2. Max. $z = 7x_1 + 9x_2$
subject to
$-x_1 + 3x_2 \leq 6$
$7x_1 + x_2 \leq 35$
$0 \leq x_1, x_2 \geq 7$
[IGNOU (MCA II) 2000]

3. Max. $z = 3x_1 + x_2$
subject to
$3x_1 - x_2 + x_3 = 12$
$3x_1 + 11x_2 + x_4 = 66$
$x_j \geq 0, j = 1, 2, 3, 4.$

4. Max. $z = x_1 + x_2$
subject to
$4x_1 - x_2 \leq 10$
$2x_1 + 5x_2 \leq 10$
$x_1, x_2 = 0, 1, 2, 3.$

5. Min. $z = -5x_1 + 7x_2 + 10x_3 - 3x_4 + x_5$
subject to the constraints
$x_1 + 3x_2 - 5x_3 + x_4 + 4x_5 \leq 0$
$2x_1 + 6x_2 - 3x_3 + 2x_4 + 2x_5 \geq 4$
$x_2 - 2x_3 - x_4 + x_5 \leq -2$
$x_i = 0, 1, (i = 1, 2, \ldots, 5).$

6. $z = 21x_1 + 11x_2$
subject to
$7x_1 + 4x_2 + x_3 = 13$
$x_2 \leq 5, x_1, x_2, x_3 \geq 0$
and integers
[Vidyasagar 97]

7. Min. $z = -4x_1 + 5x_2 + x_3 - 3x_4 + x_5$ subject to the constraints
$-x_1 + 2x_2 - x_4 - x_5 \leq -2, -4x_1 + 5x_2 + x_3 - 3x_4 + x_5 \leq -2, -x_1 - 3x_2 + 2x_3 + 6x_4 - 25x_5 \leq 1$ every $x_j = 0, 1.$

10.9. APPLICATIONS OF INTEGER PROGRAMMING

We present in this section a number of applications of integer programming (all-integer and mixed). Some of these applications are connected with the direct formulation of the problem.

1. **Travelling Salesman Problem.** Let us asume that there are n towns with known distances between any pair of cities. A salesman wants to start from his home town; visit each town once, and then return to his starting point. The objective is to minimize the total travelling time (or cost or distance).
This problem can be formulated as zero-one integer programming problem. In a linear programming problem, if all the variables are restricted to take the values of 0 or 1 only, then such linear programming problem is called zero-one programming. The formulation of Travelling Salesman Problem is as follows:
$$\text{Min } z = \sum_i \sum_j \sum_k d_{ij}\, x_{ijk}, \; i \neq j,$$
where d_{ij} denotes the distance from town i to town j, and i, j, k are integers varying from 1 to n
$$x_{ijk} = \begin{cases} 1, \text{ if the } k\text{th directed arc is from town } i \text{ to town } j. \\ 0, \text{ if otherwise,} \end{cases}$$
The constraints are of the following type:
(i) $\sum_i \sum_{j, i \neq j} x_{ijk} = 1, \quad k = 1, 2, \ldots, n.$
This implies that only one directed arc may be assigned to a specific value of k.
(ii) $\sum_j \sum_k x_{ijk} = 1, i = 1, 2, \ldots, n.$
This implies that only other town may be reached from a specifiec town i.
(iii) $\sum_i \sum_k xsubijk = 1, j = 1, 2, \ldots, n.$
This implies that only one other town can initiate directed arc to a specified town j.
(iv) $\sum_{i, i \neq j} x_{ijk} = \sum_{r, r \neq j} x_{jr(k+1)},$ for all j and k.
This constraint will ensure that the round trip will consist of connected arcs. It is given that the kth directed arc ends at some specific town j, the $(k + 1)$th directed are must start at the same town j.
This problem has several practical applications.

2. **Fixed Charge Problelm.** It is the problem where it is required to produce at least N units of a certain product on n different machines.
Let x_j be the number of units produced on machine $j, j = 1, 2, \ldots n.$.

The production cost function for the jth machine is given by

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

where k_j is the setup cost for machine j. Thus, the formulation of the problem is given by :

$$\text{Min. } \mathbf{z} = \sum_{j=1}^{n} c_j(x_j) \text{ , subject to } \sum_{j=1}^{n} x_j \geq N \text{ , } x_j \geq 0 \text{ and integer.}$$

It is important ot note in the above formulation that the objective function is non-linear because of the presence of the fixed-charge k_j. This difficulty may be removed by using the mixed integer programming as follows :

Let M be a very large number exceeding the capacity of any of the machines and let $y_j = 0$ or 1 for all j. The above formulation thus reduces to :

$$\text{Min. } \mathbf{z} = \sum_{j=1}^{n} k_j y_j + \sum_{j=1}^{n} c_j x_j ,$$

subject to $\sum_{j=1}^{n} x_j > N$, $x_j \leq My_j$ for all j, $x_j \geq 0$ and integer, $y_j = 0$ *or* 1 for all j

Thus, we can solve this problem by usual techniques discussed in this chapter.

Q. 1. State the computational process for solving a linear programming problem with upper bound conditions.

2. Discuss the importance of integer programming problem in optimization theory. Formate the travelling salesman problem as an integer programming problem.
3. State the fixed charges problem. Show how to formulate this problem as a mixed integer programming problem.
4. Explain gomory's method for solving an all integer linear programming problem. Formulate the travelling salesman problem as an integer programming problem.

10.10. ZERO-ONE (0-1) PROGRAMMING

If all the variables in a linear programming problem are restricted to take the value *zero* or *one* only, then such L.P.P. is known as *zero-one programming* problem. Various methods are available for solving the *zero-one programming problems.*

The study of zero-one programming problems is specially important because of two reasons :

(i) A certain class of *integer non-linear* programming problems can be converted into equivalent *zero-one* linear programming problems.

(ii) A large variety of management and industrial problems can also be formulated as zero-one programming problems.

The general integer programming methods such as Branch-and-Bound method can be used to solve a zero-one L.P.P. simply by introducing the additional constraints that all the variables must be less than or equal to one. The general integer programming methods were primarily developed for solving such type of problems, they do not take advantage of the special features of zero-one L.P.P. Thus a number of methods have been developed to solve zero-one linear programming problems more easily.

The theoretical development of these methods is beyond the scope of this book.

Q. 1. What is meant by zero-one programming problem ?

2. Write a short note on integer programming.

SELF-EXAMINATION PROBLEMS

1. Derive the expression for Gomory-cut in the case of mixed integer linear programming problem. Apply it to obtain initial iterate to the following problem :

   Min. $z = -110x_1 - 80x_2 - 60x_3 - 180x_4$, subject to the constraints

   $x_1 + x_2 + x_3 + x_4 + x_5 = 20$

   $2x_1 + 3x_2 + 4x_3 + 5x_4 + x_6 = 50$

   $x_1, x_2, x_3 = 0, 1, 2 \ldots\ldots, x_4, x_5, x_6 \geq 0$.

Following is the optimal iterate tableau of the corresponding linear programming problem of miximization.

		$c_j \rightarrow$	110	80	60	180	0	0	
Basic Var.	C_B	X_B	X_1	X_2	X_3	X_4	S_1	S_2	
x_1	110	50/3	1	2/3	1/3	0	5/3	− 1/3	
x_4	180	10/3	0	1/3	2/3	1	− 2/3	1/3	
	$z = 2433\frac{1}{3}$		0	$-53\frac{1}{3}$	$-96\frac{2}{3}$	0	$-63\frac{1}{3}$	$-23\frac{1}{4}$	$\leftarrow \Delta_j$

2. A company stocks an item that deteriorates with time as measured in weekly periods. The company has on hand four such items. The present ages of these items are A_1, A_2, A_3 and A_4. it has contracted to sell the stock as follows : it must deliver one item at each of weeks t_1, t_2, t_3 and t_4 from now : the revenue for an item is a function of its age at the time of delivery.

Formulate this optimization problem as a programming problem.

[Your answer should specifically indicate the feasible range of each variable involved.]

3. Suppose that three items are to be sequenced through n machines. Each item must be processed first on machine 1, then on machine 2, ..., and finally on machine n. The sequence of jobs may be different for each machine. Let t_{ij} be the time required to perform the work on item i by machine j; assume each t_{ij} is an integer. The objective is to minimize the total make-span to complete all items. Formulate the problem as an integer programming model.

4. Formulate the following Capital Budgeting problem as a zero-one integer programming problem given in the following data. There are four projects under consideration. Assume that the project run into three years. Total available funds are Rs. 75,000 (to be used at the rate of Rs. 25,000/– each year). The expected profit and cost break-up is as follows :

Projects	Expected Profit	Cost		
		Year 1	Year 2	Year 3
1	90,000	8,000	10,000	12,000
2	60,000	2,000	5,000	8,000
3	1,80,000	15,000	10,000	5,000
4	1,00,000	10,000	5,000	5,000

5. Suppose five items are to be loaded on the vessel. The weight W, volume V and price p are tabulated below. The maximum cargo weight and cargo volume are $W = 112$, $V = 109$ respectively. Determine the most valuable cargo load in discrete unit of each item :

Item	1	2	3	4	5
W	5	8	3	2	7
V	1	8	6	5	4
Price (Rs.)	4	7	6	5	4

Formulate the problem as integer programming model and then solve.

6. (a) Suppose that salesman has to travel n cities where he starts from his home city and visits each of other $n - 1$ cities once and only once and returns home city. Let d_{ij} be the distance between city i and city j. Formulate the problem as integer programming problem if he wishes to minimize the total distance travelled.

(b) Describe the cutting plane method to solve integer programming problem.

7. Following is the optimal table of an L.P.P.

Basic V.	X_B	X_1	X_2	S_1	A_1	A_2	S_2	
x_1	3/5	1	0	1/5	3/5	− 1/5	0	
x_2	6/5	0	1	− 3/5	− 4/5	3/5	0	
s_2	0	0	0	1	1	− 1	1	
	$z = 12/5$	0	0	1/5	$M - \frac{2}{5}$	$M - 1/5$	0	$\leftarrow \Delta_j$

Find the optimal solution to the problem when x_2 is required to take an integer value. **[Roorkee M.Sc. I (OR) 96]**

8. Consider the following integer programming problem :

Maximize $9x_1 + 7x_2$,

Subject to $3x_1 - x_2 \leq 6$,

$x_1 + 7x_2 \leq 35$

where x_1, $x_2 \geq 0$ and are integers. **[IGNOU (MCA II) 2000]**

OBJECTIVE QUESTIONS

1. In a mixed-integer programming problem
   (a) all of the decision variables require integer solutions. (b) few of the decision variables require integer solutions.
   (c) different objective functions are mixed together. (d) none of the above.
2. The use of cutting plane method
   (a) reduces the number of constraints in the given problem.
   (b) yields better value of objective function.
   (c) require use of standard LP approach between each cutting plane application.
   (d) all of the above.
3. The 0-1 integer programming problem
   (a) requires the decision variables to have values between zero and one.
   (b) requires that the constraints all have coefficients between zero and one.
   (c) requires that the decision variables have coefficients between zero and one.
   (d) all of the above.
4. The part of the feasible solution space eliminated by plotting a cut contains
   (a) only non-integer solutions. (b) only integer solutions.
   (c) both (a) and (b). (d) none of the above.
5. While solving IP problem any non-integer variable in the solution is picked-up to
   (a) obtain the cut constraint. (b) enter the solution.
   (c) leave the solution. (d) none of the above.
6. Branch and Bound method divides the feasible solution space into smaller parts by
   (a) branching. (b) bounding. (c) enumerating. (d) all of the above.
7. Rounding off solution values of decision variables in an LP problem may not be acceptable because
   (a) it does not satisfy constraints.
   (b) it violates non-negativity conditions.
   (c) objective function value is less than the objective function value of LP.
   (d) none of the above.
8. In the Branch and Bound approach to a max. problem, a node is terminated if
   (a) a node has an infeasible solution.
   (b) a node yields a solution that is feasible but not an integer.
   (c) upper bound is less than the current sub-problem's lower bound.
   (d) all of the above.
9. Which of the following is the consequence of adding a new cut constraint to an optimal simplex table
   (a) addition of a new variable to the table.
   (b) makes the previous optimal solution infeasible.
   (c) eliminates non-integer solution from the solution space.
   (d) all of the above.
10. In a Branch and Bound minimization tree, the lower bounds on objective function value
    (a) do not decrease in value. (b) do not increase in value. (c) remain constant. (d) none of the above.

Answers

1. (b) **2.** (a) **3.** (a) **4.** (a) **5.** (a) **6.** (a) **7.** (d) **8.** (d) **9.** (d) **10.** (b).

✱✱✱

11

TRANSPORTATION PROBLEMS

11.1. INTRODUCTION

As already defined and discussed earlier, the simplex procedure can be regarded as the most generalized method for linear programming problems. However, there is very interesting class of *'Allocation Methods'* which is applied to a lot of very practical problems generally called *'Transportation Problems'*. Whenever it is possible to place the given linear programming problem in the transportation frame-work, it is far more simple to solve it by *'Transportation Technique'* than by *'Simplex'*.

Let the nature of transportation problem be examined first. If there are more than one centres, called *'origins'*, from where the goods need to be shipped to more than one places called *'destinations'* and the costs of shipping from each of the *origins* to each of the *destinations* being different and known, the problem is to ship the goods from various *origins* to different *destinations* in such a manner that the cost of shipping or transportation is minimum.

Thus, we can formally define the transportation problem as follows :

Definition. ***The Transportation Problem** is to transport various amounts of a single homogeneous commodity, that are initially stored at various origins, to different destinations in such a way that the total transportation cost is a minimum.*

For example, a tyre manufacturing concern has m factories located in m different cities. The total supply potential of manufactured product is absorbed by n retail dealers in n different cities of the country. Then, transportation problem is to determine the transportation schedule that minimizes the total cost of transporting tyres from various factory locations to various retail dealers.

The various features of linear programming can be observed in these problems. Here the availability as well as the requirements of the various centres are finite and constitute the limited resources. It is also assumed that the cost of shipping is linear (for example, the costs of shipping of *two* objects will be *twice* that of shipping a *single* object). However, this condition is not often true in practical problems, but will have to be assumed so that the linear programming technique may be applicable to such problems. These problems thus could also be solved by 'Simplex'. Mathematically, the problem may be stated as given in the following section.

Q. Define transportation problem. **[Bhubneshwar (IT) 2004]**

11.2. MATHEMATICAL FORMULATION

Let there be m origins, ith origin possessing a_i units of a certain product, whereas there are n destinations (n may or may not be equal to m) with destination j requiring b_j units. Costs of shipping of an item from each of m origins (sources) to each of the n destinations are known either directly or indirectly in terms of mileage, shipping hours, etc. Let c_{ij} be the cost of shipping one unit product from ith origin (source) to jth destination., and 'x_{ij}' be the amount to be shipped from ith origin to jth destination.

It is also assumed that total availabilities Σa_i satisfy the total requirements Σb_j, *i.e.*,

$$\Sigma a_i = \Sigma b_j \quad (i = 1, 2, \ldots, m \,; j = 1, 2, \ldots, n) \qquad \ldots(11\cdot1)$$

(***In case***, $\Sigma a_i \neq \Sigma b_j$ some manipulation is required to make $\Sigma a_i = \Sigma b_j$, which will be shown later).

The problem now is to determine non-negative (≥ 0) values of 'x_{ij}' satisfying both, the availability constraints :

$$\sum_{j=1}^{n} x_{ij} = a_i \qquad \text{for } i = 1, 2, \ldots, m \qquad \ldots(11\cdot2)$$

as well as the requirement constraints :

$$\sum_{i=1}^{m} x_{ij} = b_j \qquad \text{for } j = 1, 2, \ldots, n \qquad \ldots(11\cdot3)$$

and *minimizing* the total cost of transportation (shipping)

$$z = \sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij}\, c_{ij} \quad (\textit{objective function}). \qquad \ldots(11\cdot4)$$

It may be observed that the constraint equations (11·2), (11·3) and the objective function (11·4) are all linear in x_{ij}, so it may be looked like a linear programming problem.

This special type of LPP will be called a ***Transportation Problem*** (T.P.).

Remark : By requiring strict inequalities $a_i > 0$ and $b_j > 0$ we are not restricting anything. Since all $x_{ij} \geq 0$, it follows that each $a_i \geq 0$ and each $b_j \geq 0$. Moreover any $a_k = 0 \Rightarrow x_{kj} = 0$ and thus can be eliminated from the problem.

Q. 1. Explain Transportation problem and show that it can be considered as L.P.P.
2. Formulate transportation problem as a L.P.P.
3. Specify a transportation problem (TP). Is this an LPP ? **[AIMS (MBA) 2002]**
4. Explain the difference between a transportation problem and an assignment problem. Explain situations where an assignment problem can arise. **[Meerut (Maths) 99]**
5. Show that assignment problem is the special case of the transportation problem. **[IAS (Main) 88]**
6. Give the mathematical formulation and difference between 'Transportation' and 'Assignment' problems. **[Agra 99; Kanpur 96; Meerut (IPM) 91; 90]**

11.3. MATRIX FORM OF TRANSPORTATION PROBLEM

Consider the transportation problem as mathematically formulated above. The set of constraints $\sum_{j=1}^{n} x_{ij} = a_i \ (i = 1, 2, \ldots, m)$ and $\sum_{i=1}^{m} x_{ij} = b_j \ (j = 1, 2, \ldots n)$ represent $m + n$ equations in mn non-negative variables x_{ij}. Each variable x_{ij} appears in exactly two constraints, one associated with the ith origin O_i and the other with the jth destination D_j. In the above ordering of constraints, first we write the origin-equations and then destination-equations. Then the transportation problem can be restated in the matrix form as :

Minimize $z = \mathbf{CX}$, $\mathbf{X} \in \mathbf{R}^{mn}$, subject to the constraints $\mathbf{AX} = \mathbf{b}$, $\mathbf{X} \geq \mathbf{0}$, $\mathbf{b} \in \mathbf{R}^{m+n}$

where $\mathbf{X} = [x_{11}, \ldots, x_{1n}, x_{21}, \ldots, x_{2n}, \ldots, x_{m1}, \ldots, x_{mn}]$, $\mathbf{C}$ *is the cost vector,* $\mathbf{b} = [a_1, a_2, \ldots, a_m, b_1, b_2, \ldots, \ldots, b_n]$ and $\mathbf{A}$ is an $(m + n) \times mn$ real matrix containing the coefficients of constraints.

It is worthnoting that the elements of A are either 0 or 1. Thus the *general* LPP can be reduced to *transportation problem* if

(i) a_{ij}'s are restricted to the values 0 and 1; and (ii) the units among the constraints are homogeneous.

For example, if $m = 2$, $n = 3$, the matrix A is given by

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e}_{23}^{(1)} & \mathbf{e}_{23}^{(2)} \\ \mathbf{I}_3 & \mathbf{I}_3 \end{bmatrix}$$

And, therefore, for a general transportation problem, we may write

$$\mathbf{A} = \begin{bmatrix} \mathbf{e}_{mn}^{(1)} & \mathbf{e}_{mn}^{(2)} & \cdots & \mathbf{e}_{mn}^{(m)} \\ \mathbf{I}_n & \mathbf{I}_n & \cdots & \mathbf{I}_n \end{bmatrix}$$

where $\mathbf{e}_{mn}^{(i)}$ is an $m \times n$ matrix having a row of unit elements as its ith row and 0's everywhere else, and $\mathbf{I}_n$ is the $n \times n$ identity matrix.

If $\mathbf{a}_{ij}$ denotes the column vector of A associated with any variable x_{ij}, then it can be easily verified that

$$\mathbf{a}_{ij} = \mathbf{e}_i + \mathbf{e}_{m+j}, \text{ where } \mathbf{e}_i, \mathbf{e}_{m+j} \in \mathbf{R}^{m+n} \text{ are unit vectors.}$$

Q. 1. Show how a 2 × 3 transportation problem may be transformed into a special network termed bipartite network.
2. If a transportation porblem has p factories and 2 retail shops, what is the number of variables and what is the number of constraints? [IGNOU 99, 96]

11.4. FEASIBLE SOLUTION, BASIC FEASIBLE SOLUTION, AND OPTIMUM SOLUTION

The terms *feasible solution, basic feasible solution* and *optimum solution* may be formally defined with reference to the transportation problem (T.P.) as follows:

(i) Feasible Solution (FS). A set of non-negative individual allocations ($x_{ij} \geq 0$) which simultaneously removes deficiencies is called a ***feasible solution.***

(ii) Basic Feasible Solution (BFS). A feasible solution to a m-origin, n- destination problem is said to be *basic* if the number of positive allocations are $m + n - 1$, *i.e.*, one less than the sum of rows and columns (*see Theorem* 11·2).
If the number of allocations in a basic feasible solution are less than $m + n - 1$, it is called *degenerate* BFS (otherwise, ***non-degenerate*** BFS).

(iii) Optimum Solution. A feasible solution (not necessarily basic) is said to be optimal if it minimizes the total transportation cost.

Q. Define the terms (a) Feasible solution (b) Basic feasible solution (c) Optimum in solution.

11·4-1. Existence of Feasible Solution

Theorem 11.1. (***Existence of Feasible Solution***). *A necessary and sufficient condition for the existence of feasible solution of a transportation problem is* $\Sigma\, a_i = \Sigma\, b_j$ $(i = 1, \ldots, m ; j = 1, \ldots, n)$. [Rewa (M.P.) 93]

Proof. ***The condition is necessary.*** Let there exist a feasible solution to the transportation problem. Then,

$$\sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij} = \sum_{i=1}^{m} a_i, \quad \sum_{j=1}^{n} \sum_{i=1}^{m} x_{ij} = \sum_{j=1}^{n} b_j \Leftrightarrow \sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

The condition is sufficient. Let $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j = k$ (say).

If $\lambda_i \neq 0$ be any real number such that $x_{ij} = \lambda_i\, b_j$ for all i and j, then λ_i is given by

$$\sum_{j=1}^{n} x_{ij} = \sum_{j=1}^{n} \lambda_i\, b_j = \lambda_i \sum_{j=1}^{n} b_j = k\,\lambda_i \Rightarrow \lambda_i = \frac{1}{k} \sum_{j=1}^{n} x_{ij} = \frac{a_i}{k}.$$

Thus, $\quad x_{ij} = \lambda_i\, b_j = \dfrac{a_i\, b_j}{k} \geq 0$, since $a_i > 0$, $b_j > 0$ for all i and j

Hence a feasible solution exists.

11·4-2. Basic Feasible Solution of Transportation Problem

It has been observed that a *transportation problem* is a special case of a *linear programming problem.* So a basic feasible solution of a transportation problem has the same definition as earlier given for L.P.P. (in *Sec* 3·8, page 93). However, we observe that in the case of a T.P., there are only $m + n - 1$ basic variables out of mn unknowns. This happens due to redundancy in the constraints of the transportation problem. This can be easily justified by proving the following theorem.

Theorem 11.2. *The number of basic variables in a transportation problem are at the most* $m + n - 1$.

Proof. To prove this, consider the first $m + n - 1$ constraints of the transportation problem as

$$\sum_{i=1}^{m} x_{ij} = b_j, \quad j = 1, 2, \ldots, n - 1 \qquad \textbf{(n − 1 equations)} \qquad \ldots(1)$$

and $\sum_{j=1}^{n} x_{ij} = a_i, \quad i = 1, 2, .. m.$ ***(m equations)*** ...(2)

Now adding $(n-1)$ destination-constraints (1), we get

$$\sum_{j=1}^{n-1} \sum_{i=1}^{m} x_{ij} = \sum_{j=1}^{n-1} b_j \quad ...(3)$$

Also, adding m origin-constraints (2), we get

$$\sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij} = \sum_{i=1}^{m} a_i \quad ...(4)$$

Then, subtracting (3) from (4), we get

$$\sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij} - \sum_{j=1}^{n-1} \sum_{i=1}^{m} x_{ij} = \sum_{i=1}^{m} a_i - \sum_{j=1}^{n-1} b_j \quad ...(5)$$

or $$\sum_{i=1}^{m} \left(\sum_{j=1}^{n} x_{ij} - \sum_{j=1}^{n-1} x_{ij} \right) = \sum_{j=1}^{n} b_j - \sum_{j=1}^{n-1} b_j \left(\because \sum_i a_i = \sum_j b_j \right)$$

or $$\sum_{i=1}^{m} \left(x_{in} + \sum_{j=1}^{n-1} x_{ij} - \sum_{j=1}^{n-1} x_{ij} \right) = b_n + \sum_{j=1}^{n-1} b_j - \sum_{j=1}^{n-1} b_j$$

or $\sum_{i=1}^{m} x_{in} = b_n$ (which is exactly the last (nth) destination-constraint)

This obviously indicates that if the first $m + n - 1$ constraints are satisfied then $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$ ensures that the $(m + n)$th constraint will be automatically satisfied.

Thus, out of $m + n$ equations, one (any) is redundant and remaining $m + n - 1$ equations form a linearly independent set. Hence the theorem is proved.

It is concluded that ***a basic feasible solution will consist of at most m + n − 1 positive variables, others being zero. In the degenerate case, some of the basic variables will also be zero, i.e., the number of positive variables will now become less than m + n − 1. By fundamental theorem of linear programming, one of the basic feasible solutions will be the optimal solution.***

11·4–3. Existence of Optimal Solution

Theorem 11.3. (Existence of an optimal solution). *There always exists an optimal solution to a balanced transportation problem.*

Proof. Let $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$, so that a feasible solution x_{ij} exists. It, therefore, follows from the constraints of the problem that each x_{ij} is bounded, *viz.*, $0 \le x_{ij} \le \min (a_i, b_j)$.

Thus the feasible region of the problem is *closed*, *bounded* and *non-empty*, and hence there exists an optimal solution.

Note. In future discussion we shall assume that the above condition holds for the transportation problem without mentioning it.

Q. 1. Prove that the transportation problem always possesses a feasible solution.

2. If all the sources are emptied and all the destinations are filled, show that $\Sigma a_i = \Sigma b_j$ is a necessary and sufficient condition for the existence of a feasible solution to the transportation problem. **[Delhi B.Sc. (Maths.) 90]**

3. Prove that the solution of the transportation problem is invariant under the addition (subtraction) of the same constant to (from) any row or column of the unit cost matrix of the problem.

4. Derive a mathematical model for a cost-minimizing 'Transportation Problem'. Show that every transportation problem has a feasible solution.

11.5. TABULAR REPRESENTATION

Suppose there are m factories and n warehouses. The transportation problem is usually represented in a tabular form (*Table* 11·1). Calculations are made directly on the 'transportation arrays' which give the current trial solution.

Table 11·1

Warehouse → Factory ↓	W_1	W_2	...	W_j	...	W_n	Factory Capacities
F_1	c_{11}	c_{12}	...	c_{1j}	...	c_{1n}	a_1
F_2	c_{21}	c_{22}	...	c_{2j}	...	c_{2n}	a_2
:	:	:		:		:	:
F_i	c_{i1}	...c_{i2}	...	c_{ij}	...	c_{in}	a_i
:	:	:	:	:	:	:	:
F_m	c_{m1}	c_{m2}	...	c_{mj}	...	c_{mn}	a_m
Warehouse requirements	b_1	b_2	...	b_j	...	b_n	$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$

Table 11·2

Warehouse → Factory ↓	W_1	W_2	...	W_j	...	W_n	Factory Capacities
F_1	x_{11}	x_{12}	...	x_{1j}	...	x_{1n}	a_1
F_2	x_{21}	x_{22}	...	x_{2j}	..	x_{2n}	a_2
:	:	:		:		:	:
F_i	x_{i1}	...x_{i2}	...	x_{ij}	...	x_{in}	a_i
:	:	:	:	:	:	:	:
F_m	x_{m1}	x_{m2}	...	x_{mj}	...	x_{mn}	a_m
Warehouse requirements	b_1	b_2	...	b_j	...	b_n	$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$

In general, *Tables 11·1 and 11·2* are combined by inserting each unit cost c_{ij} together with the corresponding amount x_{ij} into the cell (i, j). The product x_{ij} (c_{ij}) gives the net cost of shipping x_{ij} units from factory F_i to warehouse W_j.

Note. Whenever the amount x_{ij} and the corresponding unit cost c_{ij} are entered in the cell (i, j), there may be a confusion to distinguish between them. Therefore, in order to remove such confusion the quantities in parenthesis will denote the unit cost c_{ij}.

Q. 1. Describe the transportation table.

2. Describe the matrix form of the transportation problem. Illustrate with 2 origins and 3 destinations.

11.6. SPECIAL STRUCTURE OF TRANSPORTATION PROBLEM

The transportation problem has a *triangular basis, i.e.* the system of equations is represented in terms of basic variables only; non-basic variables are considered to be zero. The matrix of coefficients of the variables is triangular. In other words, there is an equation in which only one basic variable occurs ; in a second equation not more than two basic variables occur, in a third equation not more than three basic variables occur, and so on.

Equations (11·2) and (11·3) may be called the row and column equations, respectively.

Theorem 11.4. *The transportation problem has a triangular basis.*

Proof. To prove this theorem, consider equations (11·2) and (11·3) written row-wise and column-wise in the tabular form (*Table 11·2*).

There cannot be an equation in which no basic variable exists, otherwise the equation cannot be satisfied, for $a_i \neq 0$ or $b_j \neq 0$. The theorem will be proved by contradiction.

Suppose every equation has at least two basic variables. Then there will be at least two basic variables in each row, and the total number of basic variables will be at least $2m$. Also each column equation will have at least two basic variables, and hence in all there will be at least $2n$ variables. Let the total number of basic variables be N. Thus, $N \geq 2m$, $N \geq 2n$. Now, three cases may arise.

Case 1. If $m > n$, then $N \geq 2m$ becomes $N > m + n$.

Case 2. If $m < n$, then $N < 2n$ becomes $N < n + m$.

Case 3. If $m = n$, then $N \geq 2m$ becoems $N = m + n$.

Thus, it is observed that in every case $N \geq m + n$. But $N = m + n - 1$, which is a contradiction. Thus, the assumption of existing at least two basic variables in each row and each column is wrong. Therefore, at least one such row or column equation exists having one basic variable only.

Let x_{rc} be the only variable in the rth row and the cth column. Then, $x_{rc} = a_r$. Then equation can be eliminated from the system by deleting the rth row equation and substituting $x_{rc} = a_r$ in the cth column equation. Thus, rth row now stands cancelled, and b_c is replaced by $b'_c = b_c - a_r$.

The resulting system now consists of $m - 1$ row equations and n column equations of which $m + n - 2$ are linearly independent. Thus, there are $m + n - 2$ basic variables in this system. Repeat the process and it is concluded that there is an equation in the reduced system which has only one basic variable. If this equation happens to be the cth column equation in the original system, the cth column equation now contains two basic variables. Thus the original system has an equation which has at most two basic variables. Continue this process and ultimately it can be shown that there is an equation which has at most three basic variables, and so on.

So the theorem is now completely proved.

Q. 1. Prove that there are only $m + n - 1$ independent equations in a transportation problem, m and n being the number of origins and destinations, and any one equation can be dropped as the redundant equation.

2. What do you mean by the triangular form of a system of linear equations ? When we can say that a system of linear equations has a triangular basis.

3. Show that all bases for transportation problem are triangular.

4. What do you mean by non-degenerate basic feasible solution of a transportation problem.

5. State a transportation problem. When does it have a unique solution ? Explain.

11.7. LOOPS IN TRANSPORTATION TABLE AND THEIR PROPERTIES

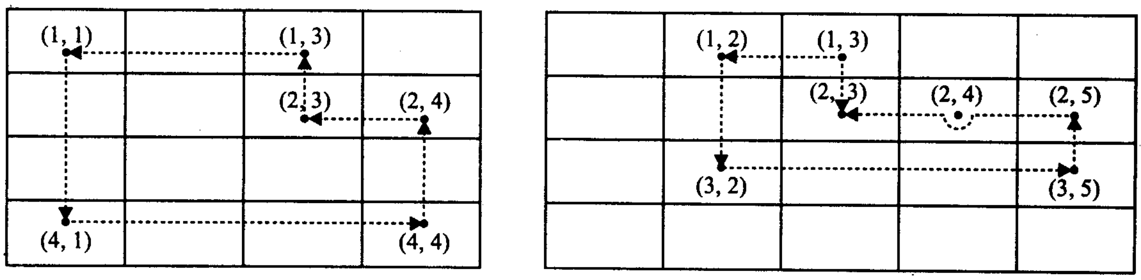
Loop. Def. *In a transportation table, an ordered set of four or more cells is said to form a* LOOP if,

(*i*) *any two adjacent cells in the ordered set lie either in the same row or in the same column ; and*

(*ii*) *any three or more adjacent cells in the ordered set do not lie in the same row or in the same column.*

The first cell of the set is considered to follow the last one in the set.

If we join the cells of a loop by horizontal and vertical lineup segments, we get a closed path satisfying the above conditions (*i*) and (*ii*). Let us denote the (i, j)th cell of the transportation table by (i, j) . Then it can be observed from the diagramatic illustration in Fig. 11.1, that the set $L = \{(1, 1), (4, 1), (4, 4), (2, 4), (2, 3), (1, 3)\}$ form a loop; and on the other hand, the set $L' = \{(3, 2), (3, 5), (2, 5), (2, 4), (2, 3), (1, 3), (1, 2)\}$ does not form a loop, because three cell entires (2, 3), (2, 4) and (2, 5) lie in the same row (second).



(i) Loop L

(ii) Non-loop L′

Fig. 11.1

Theorem 11.5. *Every loop has an even number of cells.*

Proof. For any loop, we can always choose arbitrarily a starting point and a direction by an arrow mark ($\rightarrow$). We consider a loop formed by n number of cells which are consecutively numbered from 1 to n. Now assume that cell 1 and 2 exist in the same column. Thus the step from cell 1 to cell 2 involves a row change. Obviously, step from cell 2 to cell 3 must involve a column change, from cell 3 to cell 4 a row change, and so on; in general, the step to cell k involves a row change, if and only if, k is even. Since the step to cell 2 involved a row change, the step from cell n to cell 1 must be a column change and the step from cell $n - 1$ to cell n a row change. Hence n will be even.

Set containing a loop. Def. *A set X of cells of a transportation table is said to* **contain a loop** *if the cells of X or of a subset of X can be sequenced (ordered) so as to form a loop.*

Theorem 11.6 (Linear Dependence and Loops). *Let X be a set of column vectors of the coefficient matrix of a T.P.. Then, a necessary and sufficient condition for vectors in X to be linearly dependent is that the set of their corresponding cells in the transportation table contains a loop.*

Proof. Let us consider an m-origin, n-destination T.P. expressed in its matrix form:

Minimize $z = \mathbf{CX}$; $\mathbf{C}, \mathbf{X} \in \mathbf{R}^{mn}$, subject to the constraints : $\mathbf{AX} = \mathbf{b}$, $\mathbf{X} \geq \mathbf{0}$, $\mathbf{b} \in \mathbf{R}^{m+n}$

where $\mathbf{b} = (a_1, a_2, \ldots, a_m, b_1, b_2, \ldots, b_n)$, $\mathbf{A}$ is an $(m + n) \times mn$ real matrix containing the coefficients of constraints and $\mathbf{C}$ is the cost vector.

To prove, the condition is sufficient :

Let us assume that the cells associated with the vectors of $\mathbf{X}$ contain a loop

$$L = \{ (i, j), (i, k), (l, k), \ldots, (p, o), (p, j) \}$$

If $\mathbf{a}_{ij}$ denotes the column vector of matrix $\mathbf{A}$ associated with the variable x_{ij} [the cell (i, j)], then it follows from the discussion in **sec. 11·3** that $\mathbf{a}_{ij} = \mathbf{e}_i + \mathbf{e}_{m+j}$, where $\mathbf{e}_i$, $\mathbf{e}_{m+j} \in \mathbf{R}^{m+n}$ are unit vectors. Thus $\mathbf{X}$ includes the column vectors :

$$\mathbf{a}_{ij} = \mathbf{e}_i + \mathbf{e}_{m+j}, \ \mathbf{a}_{ik} = \mathbf{e}_i + \mathbf{e}_{m+k}, \ \mathbf{a}_{lk} = \mathbf{e}_l + \mathbf{e}_{m+k}, \ \mathbf{a}_{lm} = \mathbf{e}_l + \mathbf{e}_{m+m}, \ldots, \ \mathbf{a}_{po} = \mathbf{e}_p + \mathbf{e}_{m+o},$$

and $\qquad a_{pj} = e_p + e_{m+j}$.

Hence by successive addition and subtraction, we get $\mathbf{a}_{ij} - \mathbf{a}_{ik} + \mathbf{a}_{lk} - \mathbf{a}_{lm} + \ldots + \mathbf{a}_{po} - \mathbf{a}_{pj} = \mathbf{0}$

(by *Theorem 11.5*, a loop contains an even number of cells)

Therefore, this particular subset of $\mathbf{X}$, and hence $\mathbf{X}$ itself, is a *linearly dependent* set.

To prove, the condition is necessary :

Let us assume that $\mathbf{X}$ is a linearly dependent set. Then, there must exist scalars λ_{ij} not all zero such that

$$\Sigma\lambda_{ij}\, \mathbf{a}_{ij} = 0 , \text{ where } \mathbf{a}_{ij} \in \mathbf{X} .$$

For simplification, remove all those vectors from $\mathbf{X}$ for which $\lambda_{ij} = 0$.

Now we choose arbitrarily a vector from the remaining vectors in $\mathbf{X}$. Let it be $\mathbf{a}_{ij} = \mathbf{e}_i + \mathbf{e}_{m+j}$. We claim that $\mathbf{X}$ must contain at least one more vector whose second subscript is j. Suppose to the contrary that it does not, then since $\lambda_{ij} \neq 0$, the $(m + j)$th component of the vector equation $\Sigma\lambda_{ij}\, \mathbf{a}_{ij} = \mathbf{0}$ gives $\lambda_{ij}.1 = 0$ implying $\lambda_{ij} = 0$, a contradiction. So $\mathbf{X}$ contains at least one more vector with second subscript j.

Suppose that this vector is $\mathbf{a}_{kj} = \mathbf{e}_k + \mathbf{e}_{m+j}$. By similar reasoning, we conclude that there must be at least one more vector in $\mathbf{X}$ with the first subscript k; say, $\mathbf{a}_{kl} = \mathbf{e}_k + \mathbf{e}_{m+l}$. By same argument once again, $\mathbf{X}$ must contain at least one vector with the second subscript l. Let it be, say, $\mathbf{a}_{il} = \mathbf{e}_i + \mathbf{e}_{m+l}$.

Thus we have determined four vectors in $\mathbf{X}$, namely $\mathbf{a}_{ij}$, $\mathbf{a}_{kj}$, $\mathbf{a}_{kl}$ and $\mathbf{a}_{il}$ whose corresponding cells, form a loop. Thus the proof is complete.

If the last vector is $\mathbf{a}_{nl} = \mathbf{e}_n + \mathbf{e}_{m+l}$ instead of $\mathbf{a}_{il}$, then as explained just before there must exist at least one more vector with first subscript n. If it is $\mathbf{a}_{nj}$, a loop is complete, if not, let it be $\mathbf{a}_{n0} = \mathbf{e}_n + \mathbf{e}_{m+0}$. $\mathbf{X}$ must contain at least one more vector with second subscript 0. Now two cases will arise :

(1) The first subscript of newly discovered vector is one that has already been identified. In this case a loop has been completed.

(2) The first subscript of the newly discovered vector is also new. In this case, since the number of vectors in $\mathbf{X}$ is finite (by extending the above reasoning), we conclude that eventually a loop must be formed.

Corollary. *A feasible solution to a transportation problem is basic if, and only if, the corresponding cells in the transportation table do not contain a loop.*

Proof : Left as an exercise.

This corollary provides us a method to verify whether the current feasible solution to the transportation problem is basic or not.

Q. 1. A feasible solution to a transportation problem is basic, if and only if, the corresponding cells in the transportation table do not contain.................

2. With reference to a transportation problem define the following terms :
(i) Feasible solution (ii) Basic feasible solution, (iii) Optimal solution, (iv) Non-degenerate basic feasible solution.

3. Define 'loop' in a transportation table. What role do they play ? **[Madurai B.Sc (Math.) 94]**

4. In the classical transportation problem explain as to how many independent equations are there when there are *m*-origins and *n*-destinations. What happens and how to handle the solution, when the initial assignment in the problem gives less than this number of occupied cells ?

Initial Basic Feasible Solution

11.8. THE INITIAL BASIC FEASIBLE SOLUTION TO TRANSPORTATION PROBLEM

Methods of finding an *optimal solution* of the transportation problem will consist of two main steps :

(i) ***To find an initial basic feasible solution :***

(ii) ***To obtain an optimal solution by making successive improvements to initial basic feasible solution until no further decrease in the transportation cost is possible.***

There will be fewer improvements to make if initially we start with a better initial basic feasible solution. So first we shall discuss below the methods for obtaining initial basic feasible solution of a T.P.

Remark : Although the transportation problem can be solved using the regular simplex method, its special properties provide a more convenient method for solving this type of problems. This method is based on the same theory of simplex method. It makes use, however, of some shortcuts which provide a less burdensome computational scheme.

11·8–1 Methods for Initial Basic Feasible Solution

Some simple methods are described here to obtain the initial basic feasible solution of the transportation problem. These methods can be easily explained by considering the following numerical example. However, the relative efficiency of these methods is still unanswerable.

Example 1. *Find the initial basic feasible solution of the following transportation problem.*

Table 11.3

Warehouse → Factory ↓	W_1	W_2	W_3	W_4	Factory Capacity
F_1	19	30	50	10	7
F_2	70	30	40	60	9
F_3	40	8	70	20	18
Warehouse Requirement	5	8	7	14	34

Solution.

First Method : ***North-West Corner Rule (Steping Stone Method).*** **[Kanpur (B.Sc.) 2003; IAS (Main) 96 Type]**

In this method, first construct an empty 3 × 4 matrix complete with row and column requirements (*Table 11·4*).

Table 11·4

	W_1	W_2	W_3	W_4	Available
F_1					7
F_2					9
F_3					18
Requirements→	5	8	7	14	

Insert a set of allocations in the cells in such a way that the total in each row and each column is the same as shown against the respective rows and columns. Start with cell (1, 1) at the north-west corner (upper left-hand corner) and allocate as much as possible there. In other words, $x_{11} = 5$, the maximum which can be alloocated to this cell as the total requirement of this column is 5. This allocation ($x_{11} = 5$) leaves the surplus amount of 2 units for row 1 *(Factory F_1)*, so allocate $x_{12} = 2$ to cell (1, 2). Now, allocations for first row and first column are complete, but there is a deficiency of 6 units in column 2. Therefore, allocate $x_{22} = 6$ in the cell (2, 2). Column 1 and column 2 requirements are satisfied, leaving a surplus amount of 3 units for row 2. So allocate $x_{23} = 3$ in the cell (2, 3), and column 3 still requires 4 units. Therefore, continuing in this way, from left to right and top to bottom, eventually complete all requirements by an allocation $x_{34} = 14$ in the *south-east* corner. Table 11·5 shows the resulting feasible solution.

Table 11·5

5 (19)	2 (30)			7
	6 (30)	3 (40)		9
		4 (70)	14 (20)	18
5	8	7	14	

On multiplying each individual allocation by its corresponding unit cost in '( )' and adding, the total cost becomes = 5 (19) + 2 (30) + 6 (30) + 3 (40) + 4 (70) + 14 (20) = Rs. 1015 .

Q. Explain the application of North-West Corner Rule with an example.

Second Method : *The Row Minima Method.*

Step 1. The transportation table of the given problem has 12 cells. Following the ***row minima method,*** since min (19, 30, 50, 10) = 10 , the first allocation is made in the cell (1, 4), the amount of the allocation is given by $x_{14} = \min (7 , 14) = 7$. This exhausts the availability from factory F_1 and thus we cross-out the first row from the transportation table (Table 11.6).

Table 11·6

	W_1	W_2	W_3	W_4	
F_1	(19)	(30)	(50)	7 (10)	x
F_2	(70)	(30)	(40)	(60)	9
F_3	(40)	(80)	(70)	(20)	18
	5	8	7	7	

Table 11·7

	W_1	W_2	W_3	W_4	
F_1				7	x
F_2	(70)	8 (30)	(40)	(60)	9
F_3	(40)		(70)	(20)	18
	5	8	7	7	

Step 2. In the resulting transportation table (*Table* 11·7) , since min (70, 30, 40, 60) = 30 , the second allocation is made in the cell (2, 2), the amount of allocation being $x_{22} = \min (9, 8) = 8$. This satisfies the requirement of warehouse W_2 and thus we cross-out the second column from the transportation table obtaining new *Table* 11·8 .

Step 3. In *Table* 11·8 , since min (70, 40, 60) = 40 ,the third allocation is made in the cell (2, 3), the amount being $x_{23} = \min [1, 7] = 1$. This exhausts the availability from factory F_2 ,

Table 11·8

	W_1	W_2	W_3	W_4	
F_1				7	x
F_2	(70)	8 (30)	1 (40)	(60)	x
F_3	(40)		(70)	(20)	18
	5	x	6	7	

Table 11·9

	W_1	W_2	W_3	W_4	
F_1				7	x
F_2	(70)	8 (30)	1 (40)	(60)	x
F_3	(40)		(70)	(20)	18
	5	x	6	7	

and thus we cross-out the second row from the table 11.8 getting the *Table 11·9*,

Step 4. The next allocation is made in the cell (3,4), since min (40, 70, 20) = 20, the amount of allocation being x_{34} = min (7, 18) = 7. This exhausts the requirement of warehouse W_4 and thus we cross-out the fourth column from the ***Table 11·9***.

Step 5. The next allocation is made in the cell (3, 1), since min (40, 70) = 40, the amount of allocation being x_{31} = min (5, 11) = 5. This satisfies the requirement of warehouse W_1 and so we cross-out the first column W_1 to get new ***Table 11.11.***

Table 11·10

	W_1	W_2	W_3	W_4	
F_1				7 •	x
F_2		8 •	1 •		x
F_3	(40)		(70)	7 •	11
	5	x	6	x	

Table 11·11

	W_1	W_2	W_3	W_4	
F_1				7 •	x
F_2		8 •	1 •		x
F_3	5 •		6 • 70	7 •	x
	x	x	x	x	

Step 6. The last allocation of amount x_{33} = 6 is obviously made in the cell (3, 3). This exhausts the availability from factory F_3 and requirement of warehouse W_3 simultaneously. So we cross-out third row and third column to get the final solution ***Table 11·12***.

Since the basic cells indicated by (•) do not form a loop, an initial basic feasible solution has been obtained. The solution is displayed in ***Table 11·12***.

Table 11·12

	W_1	W_2	W_3	W_4	
F_1				7 •	x
F_2		8 •	1 •		x
F_3	5 •		6 •	7 •	x
	x	x	x	x	

The transportation cost is given by

$z = 7 \times 10 + 8 \times 30 + 1 \times 40 + 5 \times 40 + 6 \times 70 + 7 \times 20$

= Rs. 1110.

Third Method : *The Column Minima Method.*

This method is similar to **row-minima method** except that we apply the concept of minimum cost on columns instead of rows. So, the reader can easily solve the above problem by column minima method also.

Fourth Method : *Lowest Cost Entry Method* (Matrix Minima Method).

The initial basic feasible solution obtained by this method usually gives a lower beginning cost. In this method, first write the cost and requirements matrix (***Table 11·13***).

Start with the lowest cost entry (8) in the cell (3, 2) and allocate as much as possible, *i.e.*, $x_{32} = 8$. The next lowest cost (10) lies in the cell (1, 4), so allocate $x_{14} = 7$. The next lowest cost (19) lies in the cell (1, 1), so make no allocation, because the amount available from factory F_1 was already used in the cell (1, 4). Next lowest cost entry is (20) in the cell (3, 4) where at the most it is possible to allocate $x_{34} = 7$ in order to complete the requirements of 7 units in column 4.

Further, next lowest cost is (30) in cells (2, 2) and (1, 2) so no allocation is possible, because the requirement of column 2 has already been exhausted. This way, required feasible solution is obtained (*Table 11·13*).

This feasible solution results in lower transportation cost, *i.e.*,
2(70) + 3(40) + 8(8) + 8(40) + 7(10) + 7(20) = **Rs. 814.**
This cost is less by Rs. 201, *i.e.*, **Rs. (1015–814)** as compared to the cost obtained by ***north-west*** *corner rule.*

Table 11·13

					Available
	•(19)	•(30)	•(50)	7(10)	7
	2(70)	•(30)	7(40)	•(60)	9
	3(40)	8(8)	•(70)	7(20)	18
Requirements	5	8	7	14	

Q. Explain the application of Matrix-Minimum method with an example.

Fifth Method. ***Vogel's Approximation Method (Unit Cost Penalty Method).*** **[Banasthali (M.Sc.) 93]**

Step 1. In lowest cost entry method, it is not possible to make an allocation to the cell (1, 1) which has the second lowest cost in the matrix. It is trivial that allocation should be made in at least one cell of each row and each column.

Table 11·14

	W_1	W_2	W_3	W_4	Available
F_1	(19)	(30)	(50)	(10)	7
F_2	(70)	(30)	(40)	(60)	9
F_3	(40)	(8)	(70)	(20)	18
Requirement	5	8	7	14	

Step 2. Next enter the *difference between the lowest and second lowest cost* entries in each column beneath the corresponding column, and put the defference between the lowest and second lowest cost entries of each row to the right of that row. Such individual differences can be thought of a **penalty** for making allocations in second lowest cost entries instead of lowest cost entries in each row or column. For example, allocate one unit in the second lowest cost cell (3, 1) instead of cell (1, 1) with lowest unit cost (19). There will be a loss (penalty) of Rs 21 per unit. In case, the lowest and second lowest costs in a row/ column are equal, the penalty will be taken zero.

Table 11·15

	W_1	W_2	W_3	W_4	Available	Penalties
F_1	*(19)	*(30)	*(50)	*(10)	7	(9)
F_2	*(70)	*(30)	*(40)	*(60)	9	(10)
F_3	*(40)	8 (8)	*(70)	*(20)	18/10	(12)
Requirements :	5	8/0	7	14		
Penalties :	(21)	(22) ↑	(10)	(10)		

Step 3. Select the row or column for which the ***penalty*** is the largest, *i.e.*, (22) (***Table* 11·15**), and allocate the maximum possible amount to the cell (3, 2) with the lowest cost (8) in the particular column (row) making $x_{32} = 8$. If there are more than one largest penalty rows (columns), select one of them arbitrarily.

Step 4. Cross-out that column (row) in which the requirement has been satisfied. In this example, second column has been crossed-out. Then find the corresponding penalties correcting the amount available from factory F_3. Construct the first reduced penalty matrix ***Table* 11·16.**

Table 11·16

	W_1	W_3	W_4	Available	Penalties
F_1	5(19)	*(50)	*(10)	7	(9)
F_2	*(70)	*(40)	*(60)	9	(20)
F_3	*(40)	*(70)	*(20)	10 (Note)	(20)
Requirements	5/0	7	14		
Penalties	(21) ↑	(10)	(10)		