# SUBJECT NAME: OPERATION RESEARCH
# SUBJECT CODE: BOE03

## Unit 2
## Transportation Problem and Assignment Problem
## (special cases of Linear Programming)

**Department of Mathematics and Scientific Computing**
**Madan Mohan Malaviya University of Technology, Gorkhpur**

# *TRANSPORTATION PROBLEM*

# *SPECIAL CASES OF LINEAR PROGRAMMING*

# Transportation Problem (TP)

Distributing any commodity from any group of supply centers, called *sources*, to any group of receiving centers, called *destinations*, in such a way as to minimize the total distribution cost (shipping cost).

The transportation problem deals with the transportation of any product from $m$ origins (sources), $O_1, O_2 \ldots, O_m$, to $n$ destinations, $D_1, D_2 \ldots, D_n$, with the aim of minimizing the total distribution cost, where:

- The origin $O_i$ has a supply of $a_i$ units, $i = 1, \ldots, m$ and the destination $D_j$ has a demand for $b_j$ units to be delivered from the origins, $j = 1, \ldots, n$ such that

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \qquad \text{(Balanced TP)}$$

- $c_{ij}$ is the cost per unit distributed from the origin $O_i$ to the destination $D_j$ $i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n$

# MATRIX FORMAT OF TRANSPORTATION PROBLEM

The relevant data for any transportation problem can be summarized in a matrix format using a tableau called the transportation costs tableau. The tableau displays the origins with their supply, the destinations with their demand and the transportation per-unit costs.

# MATRIX FORMAT OF TRANSPORTATION PROBLEM

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | **Destinations** | | | | | | |
| | | $D_1$ | $D_2$ | $D_3$ | $\cdots$ $D_j$ $\cdots$ | $D_n$ | **Supply** |
| **Origins or Sources** | $O_1$ | $C_{11}$ | $C_{12}$ | $C_{13}$ | $\cdots$ $C_{1j}$ $\cdots$ | $C_{1n}$ | $a_1$ |
| | $O_2$ | $C_{21}$ | $C_{22}$ | $C_{23}$ | $\cdots$ $C_{2j}$ $\cdots$ | $C_{2n}$ | $a_2$ |
| | $O_3$ | $C_{31}$ | $C_{32}$ | $C_{33}$ | $\cdots$ $C_{3j}$ $\cdots$ | $C_{3n}$ | $a_3$ |
| | $\vdots$ $O_i$ $\vdots$ | $\vdots$ $C_{i1}$ $\vdots$ | $\vdots$ $C_{i2}$ $\vdots$ | $\vdots$ $C_{i3}$ $\vdots$ | $\vdots$ $\cdots$ $C_{ij}$ $\cdots$ $\vdots$ | $\vdots$ $C_{in}$ $\vdots$ | $\vdots$ $a_i$ $\vdots$ |
| | $D_m$ | $C_{m1}$ | $C_{m2}$ | $C_{m3}$ | $\cdots C_{mj} \cdots$ | $C_{mn}$ | $a_m$ |
| | **Demand** | $b_1$ | $b_2$ | $b_3$ | $\cdots$ $b_j$ $\cdots$ | $b_n$ | $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$ |

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \qquad \text{(Balanced TP)}$$

# Cont..

In mathematical terms, the above problem can be expressed as finding a set of $x_{ij}$'s, $i = 1, 2, \ldots, m, j = 1, 2, \ldots, n$, to meet supply and demand requirements at a minimum distribution cost. The corresponding linear model is:

$$\min z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

subject to

$$\sum_{j=1}^{n} x_{ij} \leq a_i, \qquad i = 1,2, \ldots \ldots m$$

$$\sum_{i=1}^{m} x_{ij} \geq b_j, \qquad j = 1,2, \ldots, n.$$

$$x_{ij} \geq 0, \qquad i = 1, 2, \ldots, m, \qquad j = 1, 2, \ldots, n$$

Thus, the problem is to determine $x_{ij}$ , the number of units to be transported from $i^{th}$ origin $O_i$ to $j^{th}$ destination $D_j$ , so that supplies will be consumed and demands satisfied at an overall minimum cost.

# CONTD…

The first $m$ constraints correspond to the supply limits, and they express that the supply of commodity units available at each origin must not be exceeded. The next $n$ constraints ensure that the commodity unit requirements at destinations will be satisfied. The decision variables are defined positive, since these represent the number of commodity units transported.

Thus the transportation problem in standard form is shown below:

$$\min z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \qquad (1)$$

subject to

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

$$\sum_{i=1}^{m} x_{ij} = b_j, \qquad j = 1,2, \ldots, n \qquad (3)$$

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \qquad (4)$$

$$x_{ij} \geq 0, \qquad i = 1, 2, \ldots, m, \qquad j = 1, 2, \ldots, n \qquad (5)$$

# CONTD…

The above information can be put in the form of a general matrix shown below:

Destinations

| | | 1 | 2 | 3 | ... $j$ ... | $n$ | Supply |
|---|---|---|---|---|---|---|---|
| | 1 | $C_{11}$ $x_{11}$ | $C_{12}$ $x_{12}$ | $C_{13}$ $x_{13}$ | $C_{1j}$ $x_{1j}$ | $C_{1n}$ $x_{1n}$ | $a_1$ |
| | 2 | $C_{21}$ $x_{21}$ | $C_{22}$ $x_{22}$ | $C_{23}$ $x_{23}$ | $C_{2j}$ $x_{2j}$ | $C_{2n}$ $x_{2n}$ | $a_2$ |
| Sources or Origins | 3 | $C_{31}$ $x_{31}$ | $C_{32}$ $x_{32}$ | $C_{33}$ $x_{33}$ | $C_{3j}$ $x_{3j}$ | $C_{3n}$ $x_{3n}$ | $a_3$ |
| | : | $C_{i1}$ $x_{i1}$ | $C_{i2}$ $x_{i2}$ | $C_{i3}$ $x_{i3}$ | $C_{ij}$ $x_{ij}$ | $C_{in}$ $x_{in}$ | : $a_i$ : |
| | $m$ | $C_{m1}$ $x_{m1}$ | $C_{m2}$ $x_{m2}$ | $C_{m3}$ $x_{m3}$ | $C_{mj}$ $x_{mj}$ | $C_{mn}$ $x_{mn}$ | $a_m$ |
| Demand | | $b_1$ | $b_2$ | $b_3$ | ... $b_j$ ... | $b_n$ | |

# CONTD…

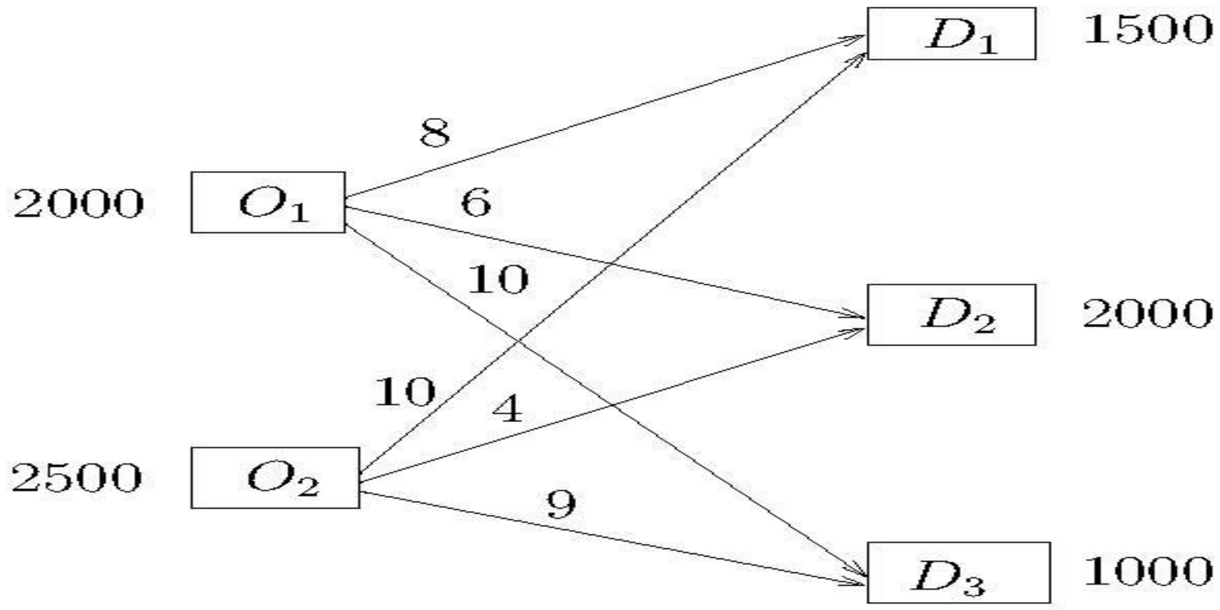
Thus the transportation problem is a L.P.P. of special type, where we are required to find the values of $m.n$ variables that minimize the objective function $z$ given by by (1), satisfying $(m+n)$ constraints and one restrictions (2)-(4) and non-negative restriction of variables (5).

The restriction (4) indicate that one of $m+n$ constraints given in (2)-(3) is redundant. Thus in $m \times n$ transportation problem only $m+n-1$ equations form a linearly independent set of equations. Thus $\mathrm{B.F.S.}$ of $m \times n$ transportation problem will contain at most $m+n-1$ positive variables while the remaining $mn-(m+n-1)$ variables will be zero.

Hence number of basic variable in basic solution in $m \times n$ transportation problem $= m+n-1$

# CONTD…

**Example:** Two bread factories, $O_1$ and $O_2$, make the daily bread in a city. The bread is delivered to the three bakeries of the city: $D_1, D_2$ and $D_3$. The supplies of bread factories, the demands of bakeries and the per unit transportation costs are displayed in the following graph:




Write this problem in matrix form. Formulate this transportation problem as L.P.P.

# CONTD…

**Solution:** Given Transportation problem in Matrix form

| Origins | Destinations | | | | |
|---|---|---|---|---|---|
| | | $D_1$ | $D_2$ | $D_3$ | Supply |
| | $O_1$ | 8 $x_{11}$ | 6 $x_{12}$ | 10 $x_{13}$ | 2000 |
| | $O_2$ | 10 $x_{21}$ | 4 $x_{22}$ | 9 $x_{23}$ | 2500 |
| | Demand | 1500 | 2000 | 1000 | 4500 |

The two origins of the problem are shown in the left column of the tableau, the bread factories $O_1$ and $O_2$, and their supply are displayed in the right column. The destinations appear at the top row of the array, bakeries $D_1$, $D_2$ and $D_3$, and their demand are at the bottom row of the array. Cells lying in the $i$th row and $j$th column of the tableau give the per-unit distribution cost coefficient $c_{ij}$, $i = 1, 2$, $j = 1,2,3$. $x_{ij}$ is the number of units of bread to be distributed from the bread factory $O_i$ to the bakery $D_j$ , $i = 1, 2,\ j = 1, 2, 3$.

# CONTD…

We define the following decision variables:

The corresponding linear model is the following:

min $z = 8x_{11} + 6x_{12} + 10x_{13} + 10x_{21} + 4x_{22} + 9x_{23}$

subject to

$$x_{11} + x_{12} + x_{13} = 2000$$
$$x_{21} + x_{22} + x_{23} = 2500$$
$$x_{11} + x_{21} = 1500$$
$$x_{12} + x_{22} = 2000$$
$$x_{13} + x_{23} = 1000$$
$$x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23} \geq 0$$

We can write the constraints in equation form, because the total supply is equal to the total demand (Balance transportation Problem)

- No. Basic variable in Basic feasible solution= $m + n - 1 = 2 + 3 - 1 = 4$
- No. of non basic variable (variable with zero value) in basic feasible solution= $mn - (m + n - 1) = 6 - 4 = 2$

# Theorems and definitions

As we previously said, the transportation problem is just a special type of linear programming problem. We can take advantage of its special structure to adapt the simplex algorithm but we have some more efficient solution procedure. In this section we state some theorems and give some definitions that permit us derive the solution method for transportation problems.

**Theorem:** The necessary and sufficient condition for a transportation problem to have a solution is that the total demand equals the total supply ie

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

The theorem above states that, under the assumption that the total supply equals the total demand, the transportation problem always has a feasible solution. However, such assumption is not always held. When the total supply does not equal the total demand, the problem has to be adapted before being solved, and the solution will be interpreted afterwards.

# Balanced transportation problem:

**Balanced transportation problem:**

A transportation problem is said to be balanced if

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

If the transportation problem is unbalanced, we have to convert it into a balanced one before solving it. There are two possible cases:

# CONTD…

**Case 1:** The demand exceeds the supply,

$$\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$$

It is not possible to satisfy the total demand with the existing supply. In this case, a dummy source or origin $O_{m+1}$ is added to balance the model. Its corresponding supply and unit transportation cost are the following

$$a_{m+1} = \sum_{j=1}^{n} b_j - \sum_{i=1}^{m} a_i$$

$$c_{m+1,j} = 0, j = 1,2, \dots n.$$

# CONTD…

**Example:** Consider the following transportation problem in matrix format:

| | 1 | 2 | 3 | Supply |
|---|---|---|---|---|
| 1 | 2 | 4 | 3 | 10 |
| 2 | 6 | 1 | 4 | 20 |
| Demand | 20 | 20 | 20 | |

- Total supply = $a_1 + a_2 = 10 + 20 = 30.$
- Total demand = $b_1 + b_2 + b_3 = 20 + 20 + 20 = 60.$

The demand exceeds the supply. We add the dummy origin 3 to balance the problem, being $a_3 = 60 - 30 = 30$ its supply. We consider unit transportation costs $c_{31}, c_{32}$ and $c_{33}$ to be zero. This leads to the following balanced transportation problem in matrix format:

| | 1 | 2 | 3 | Supply |
|---|---|---|---|---|
| 1 | 2 | 4 | 3 | 10 |
| 2 | 6 | 1 | 4 | 20 |
| 3 | 0 | 0 | 0 | 30 |
| Demand | 20 | 20 | 20 | |

# CONTD…

**Case 2:** The supply exceeds the demand.

$$\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$$

Being the total supply higher than the total demand, we add a dummy destination $D_{n+1}$ to the problem, such that its demand and unit transportation costs are following

$$b_{n+1} = \sum_{i=1}^{m} a_i - \sum_{j=1}^{n} b_j$$

$$c_{i,n+1} = 0, i = 1,2, \dots n.$$

Example. Consider the following transportation problem in matrix format.

| | 1 | 2 | 3 | Supply |
|---|---|---|---|---|
| 1 | 3 | 2 | 1 | 50 |
| 2 | 6 | 4 | 4 | 50 |
| Demand | 20 | 20 | 20 | |

- Total supply = $a_1 + a_2 = 50 + 50 = 100.$
- Total demand = $b_1 + b_2 + b_3 = 20 + 20 + 20 = 60.$

# CONTD…

The total supply is higher than the total demand. We add a dummy destination 4, with a demand $b_4 = 100 - 60 = 40$. The unit transportation costs $c_{14}$ and $c_{24}$ are considered as zero. The balanced transportation problem in matrix format is:

| | 1 | 2 | 3 | 4 | Supply |
|---|---|---|---|---|---|
| 1 | 3 | 2 | 1 | 0 | 50 |
| 2 | 6 | 4 | 4 | 0 | 50 |
| Demand | 20 | 20 | 20 | 40 | |

**Theorem:** A balanced transportation problem always has a feasible solution.

**Theorem:** A balanced transportation problem always has a basic feasible solution. Such a solution consists of $m + n - 1$ positive variables at most.

# CONTD…

**Feasible solution:** A feasible solution to a transportation problem is a set of non-negative allocations ($x_{ij} > 0$)that satisfies the row and column sum restrictions.

**Basic feasible solution:** A feasible solution to a transportation problem is said to be a basic feasible solution if it contains no more than $m + n - 1$ non – negative allocations, where $m$ is the number of rows and $n$ is the number of columns of the transportation problem.

**Optimal solution:** A feasible solution (not necessarily basic) that minimizes (maximizes) the transportation cost (profit) is called an optimal solution.

**Non -degenerate basic feasible solution:** A basic feasible solution to a $(m \times n)$ transportation problem is said to be non – degenerate if,

1. The total number of non-negative allocations is exactly $m + n - 1$ at positive level and
2. These $m + n - 1$ allocations are in independent positions.

**Degenerate basic feasible solution:** A basic feasible solution in which the total number of non-negative allocations is less than $m + n - 1$ at positive level, is called degenerate basic feasible solution.

# Initial Basic Feasible Solution of Transportation Method

Initial basic feasible solution can be obtained by using any of the following three methods:

1. The Northwest Corner method
2. Minimum cost method (Method of Matrix minima)
3. Vogel's approximation method (Unit Cost Penalty Method)

# The Northwest Corner method

**Step 1:** Select the upper left corner cell of the transportation matrix and allocate $\min(a_1, b_1)$

**Step 2:**

(a) Subtract this value from supply and demand of respective row and column.

(b) If the supply is 0, then cross (strike) that row and move down to the next cell.

(c) If the demand is 0, then cross (strike) that column and move right to the next cell.

(d) If supply and demand both are 0, then cross (strike) both row & column and move diagonally to the next cell.

**Step 3:** Repeat this steps until all supply and demand values are 0.

# EXAMPLE

**Example:** Find the initial basic feasible solution using

1.The Northwest Corner method

2. Method of Matrix minima

3. Vogel’s approximation method

Destination

| Source | D1 | D2 | D3 | D4 | Supply |
|---|---|---|---|---|---|
| O1 | 3 | 1 | 7 | 4 | 300 |
| O2 | 2 | 6 | 5 | 9 | 400 |
| O3 | 8 | 3 | 3 | 2 | 500 |
| Demand: | 250 | 350 | 400 | 200 | 1200 |

# The Northwest Corner method

Given three sources $O_1, O_2$ and $O_3$ and four destinations $D_1$, $D_2$, $D_3$ and $D_4$. For the sources $\boldsymbol{O_1, O_2}$ and $\boldsymbol{O_3}$, the supply is 300, 400 and $\mathbf{500}$ respectively. The destinations $\boldsymbol{D_1, D_2, D_3}$ and $\boldsymbol{D_4}$ have demands $\mathbf{250, 350, 400}$ and $\mathbf{200}$ respectively.

According to North West Corner method,$(\boldsymbol{O_1, D_1})$ has to be the starting point i.e. the north-west corner of the table. Each and every value in the cell is considered as the cost per transportation. Compare the demand for column $\boldsymbol{D_1}$ and supply from the source $\boldsymbol{O_1}$ and allocate the minimum of two to the cell $(\boldsymbol{O_1, D_1})$ as shown in the figure.

The demand for Column $\boldsymbol{D_1}$ is completed so the entire column $\boldsymbol{D_1}$ will be cancelled. The supply from the source $\boldsymbol{O_1}$ remains **300 – 250 = 50**.

Destination

| Source | D1 | D2 | D3 | D4 | Supply |
|---|---|---|---|---|---|
| O1 | 250 / 3 | 1 | 7 | 4 | ~~300~~ 50 |
| O2 | 2 | 6 | 5 | 9 | 400 |
| O3 | 8 | 3 | 3 | 2 | 500 |
| Demand: | ~~250~~ 0 | 350 | 400 | 200 | 1200 |

# CONTD…

Now from the remaining table i.e. excluding column $D_1$, check the north-west corner i.e. $(O_1, D_2)$ and allocate the minimum among the supply for the respective column and the rows. The supply from $O_1$ is **50** which is less than the demand for $D_2$ (i.e. 350), so allocate **50** to the cell $(O_1, D_2)$. Since the supply from row $O_1$ is completed cancel the row $O_1$ . The demand for column $D_2$ remain $350 - 50 = 300$.

Destination

| Source | D1 | D2 | D3 | D4 | Supply |
| --- | --- | --- | --- | --- | --- |
| O1 | 250<br>3 | 50<br>1 | 7 | 4 | ~~300~~ ~~50~~ 0 |
| O2 | 2 | 6 | 5 | 9 | 400 |
| O3 | 8 | 3 | 3 | 2 | 500 |
| Demand: | ~~250~~ 0 | ~~350~~ 300 | 400 | 200 | 1200 |

# CONTD…

From the remaining table the north-west corner cell is $(\boldsymbol{O_2}, \boldsymbol{D_2})$. The minimum among the supply from source $\boldsymbol{O_2}$ (i.e 400) and demand for column $\boldsymbol{D_2}$ (i.e 300) is **300**, so allocate **300** to the cell $(\boldsymbol{O_2}, \boldsymbol{D_2})$. The demand for the column $\boldsymbol{D_2}$ is completed so cancel the column and the remaining supply from source $\boldsymbol{O_2}$ is **400 – 300 = 100**.

Destination

| Source | D1 | D2 | D3 | D4 | Supply |
|---|---|---|---|---|---|
| O1 | 250 / 3 | 50 / 1 | 7 | 4 | ~~300~~ ~~50~~ 0 |
| O2 | 2 | 300 / 6 | 5 | 9 | ~~400~~ 100 |
| O3 | 8 | 3 | 3 | 2 | 500 |
| Demand: | ~~250~~ 0 | ~~350~~ ~~300~~ 0 | 400 | 200 | 1200 |

# CONTD…

Now from remaining table find the north-west corner i.e. $(\boldsymbol{O_2}, \boldsymbol{D_3})$ and compare the $\boldsymbol{O_2}$ supply (i.e. 100) and the demand for $\boldsymbol{D_2}$ (i.e. 400) and allocate the smaller (i.e. 100) to the cell $(\boldsymbol{O_2}, \boldsymbol{D_2})$. The supply from $\boldsymbol{O_2}$ is completed so cancel the row $\boldsymbol{O_2}$. The remaining demand for column $\boldsymbol{D_3}$ remains **400 – 100 = 300**.

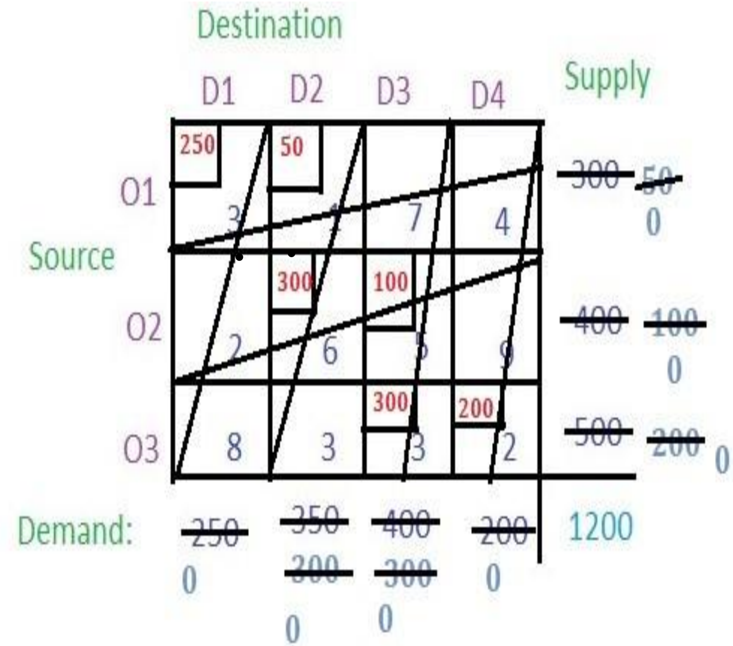
Destination

| Source | D1 | D2 | D3 | D4 | Supply |
|---|---|---|---|---|---|
| O1 | 250<br>3 | 50<br>1 | 7 | 4 | ~~300~~ ~~50~~<br>0 |
| O2 | 2 | 300<br>6 | 100<br>5 | 9 | ~~400~~ ~~100~~<br>0 |
| O3 | 8 | 3 | 3 | 2 | 500 |
| Demand: | ~~250~~<br>0 | ~~350~~<br>~~300~~<br>0 | ~~400~~<br>300 | 200 | 1200 |

# CONTD…

Proceeding in the same way, the final values of the cells will be:

Note: In the last remaining cell the demand for the respective columns and rows are equal which was cell $(\boldsymbol{O_3}, \boldsymbol{D_4})$. In this case, the supply from $\boldsymbol{O_3}$ and the demand for $\boldsymbol{D_4}$ was **200** which was allocated to this cell. At last, nothing remained for any row or column.




Initial basic feasible solution is given by

$$x_{11} = 250, x_{12} = 50, x_{22} = 300, x_{23} = 100, x_{33} = 300, x_{34} = 200$$

Min cost

$$= (250 \times 3) + (50 \times 1) + (300 \times 6) + (100 \times 5) + (300 \times 3) + (200 \times 2) = 4400$$

# Minimum cost method (Method of Matrix minima)

**Step 1:**

Select the cell having minimum unit cost $c_{ij}$ and allocate as much as possible, i.e. $\min(a_i, b_j)$.

**Step 2:**

(a) Subtract this min value from supply $a_i$ and demand $b_j$ .

(b) If the supply $a_i$ is 0, then cross (strike) that row and If the demand $b_j$ is 0 then cross (strike) that column.

(c) If min unit cost cell is not unique, then select the cell where maximum allocation can be possible.

**Step3:**

Repeat this steps for all uncrossed (unstriked) rows and columns until all supply and demand values are 0.

# 2. Method of Matrix minima

According to the Least Cost Cell method, the least cost among all the cells in the table has to be found which is **1** (i.e. cell **(O1, D2)**). Now check the supply from the row **O1** and demand for column **D2** and allocate the smaller value to the cell. The smaller value is **300** so allocate this to the cell. The supply from **O1** is completed so cancel this row and the remaining demand for the column **D2** is **350 – 300 = 50**.

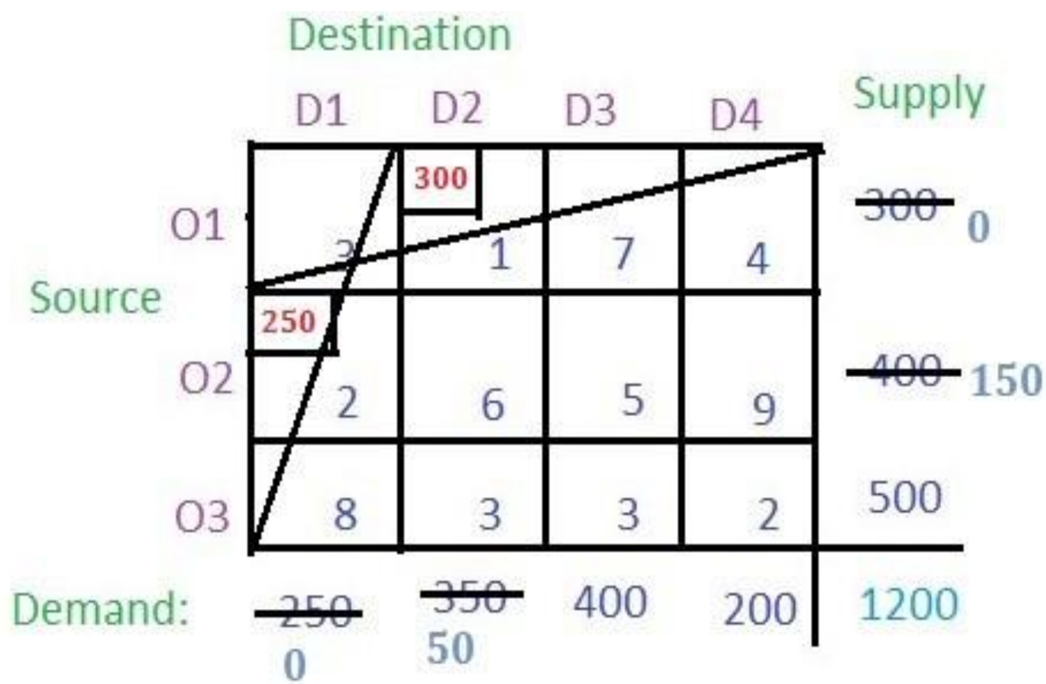
Destination

| Source | D1 | D2 | D3 | D4 | Supply |
|---|---|---|---|---|---|
| O1 | 3 | 300<br>1 | 7 | 4 | ~~300~~ 0 |
| O2 | 2 | 6 | 5 | 9 | 400 |
| O3 | 8 | 3 | 3 | 2 | 500 |
| Demand: | 250 | ~~350~~ 50 | 400 | 200 | 1200 |

# CONTD…

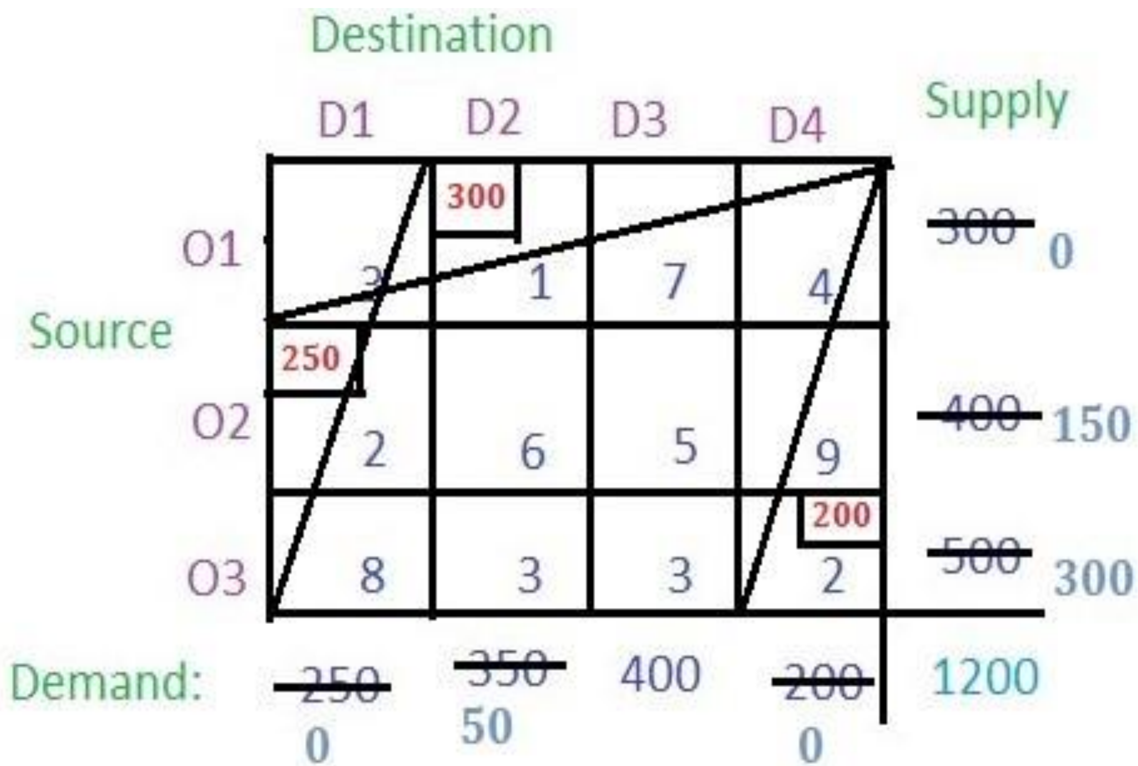
Now find the cell with the least cost among the remaining cells. There are two cells with the least cost i.e. $(\boldsymbol{O_2}, \boldsymbol{D_1})$ and $(\boldsymbol{O_3}, \boldsymbol{D_4})$ with cost **2**. Lets select $(\boldsymbol{O_2}, \boldsymbol{D_1})$.

Now find the demand and supply for the respective cell and allocate the minimum among them to the cell and cancel the row or column whose supply or demand becomes **0** after allocation.



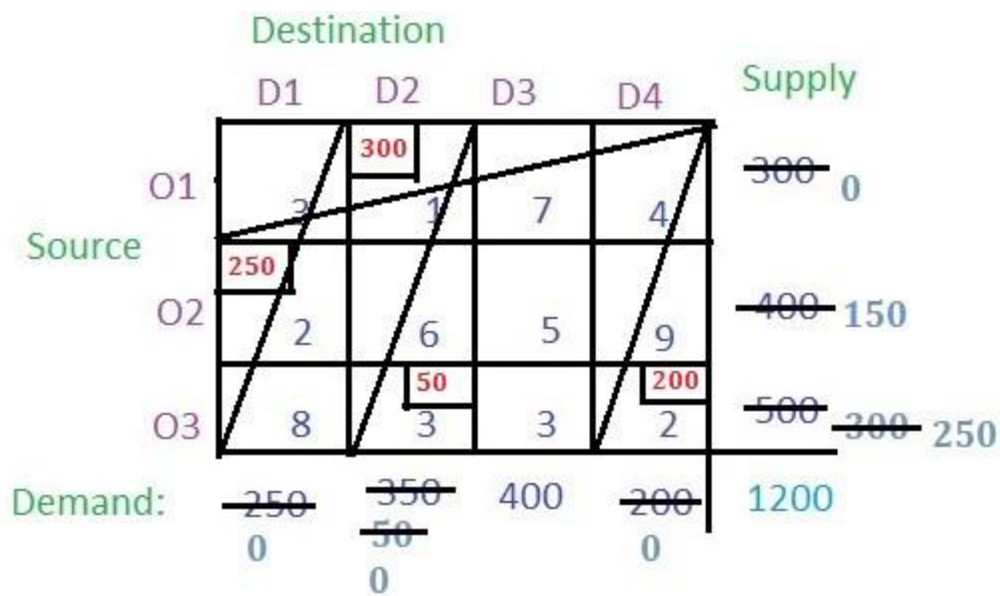
# CONTD…

Now the cell with the least cost is $(O_3, D_4)$ with cost **2**. Allocate this cell with **200** as the demand is smaller than the supply. So the column gets cancelled.



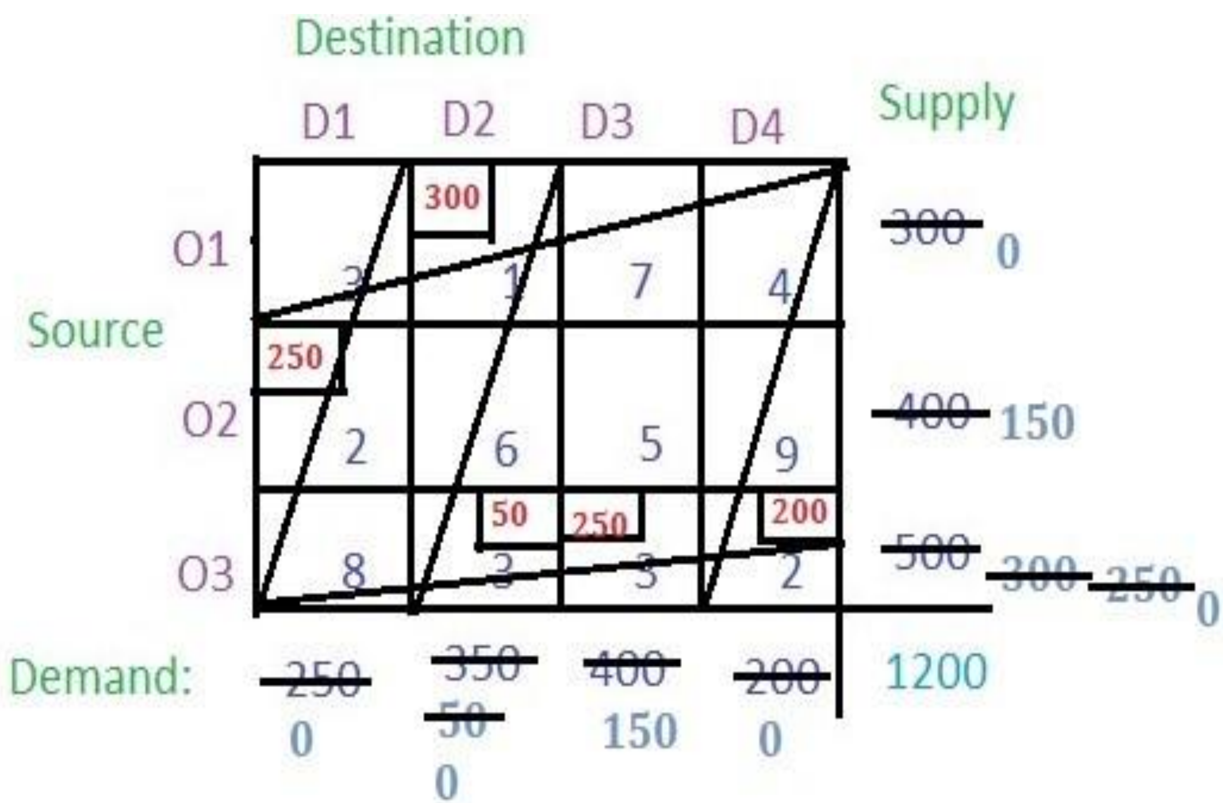
# CONTD…

There are two cells among the unallocated cells that have the least cost. Choose any at random say $(\boldsymbol{O_3}, \boldsymbol{D_2})$. Allocate this cell with a minimum among the supply from the respective row and the demand of the respective column. Cancel the row or column with zero value.



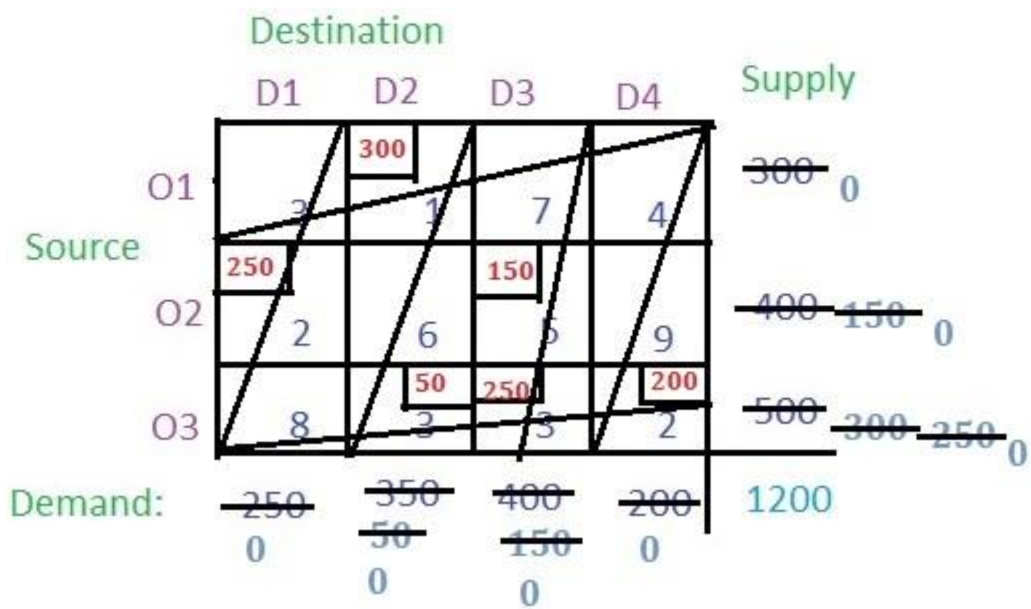
# CONTD…

Now the cell with the least cost is $(\boldsymbol{O_3}, \boldsymbol{D_3})$. Allocate the minimum of supply and demand and cancel the row or column with zero value.



# CONTD…

The only remaining cell is $(O_2, D3)$ with cost **5** and its supply is **150** and demand is **150** i.e. demand and supply both are equal. Allocate it to this cell.




Initial basic feasible solution is given by

$x_{12} = 300, x_{21} = 250, x_{23} = 150, x_{32} = 50, x_{33} = 250, x_{34} = 200$. Hence min cost = $(300 \times 1) + (250 \times 2) + (150 \times 5) + (50 \times 3) + (250 \times 3) + (200 \times 2) = 2850$

# Vogel's approximation method (Unit Cost Penalty Method)

This method is preferred over the method, because the initial basic feasible solution obtained by this method is either optimal solution or very nearer to the optimal solution.

**Step 1:** Find the cells having smallest and next to smallest cost in each row and write the difference (called penalty) along the side of the table in row penalty.

**Step 2:**Find the cells having smallest and next to smallest cost in each column and write the difference (called penalty) along the side of the table in each column penalty.

**Step 3:** Select the row or column with the maximum penalty and find cell that has least cost in selected row or column. Allocate as much as possible in this cell. If there is a tie in the values of penalties then select the cell where maximum allocation can be possible

**Step 4:** Adjust the supply & demand and cross out (strike out) the satisfied row or column.

**Step 5:** Repeat this steps until all supply and demand values are 0.

# 3. Vogel's approximation method

- For each row find the least value and then the second least value and take the absolute difference of these two least values and write it in the corresponding row difference as shown in the image below. In row $\boldsymbol{O_1}$, **1** is the least value and **3** is the second least value and their absolute difference is **2**. Similarly, for row $\boldsymbol{O_2}$ and $\boldsymbol{O_3}$ , the absolute differences are **3** and **1** respectively.
- For each column find the least value and then the second least value and take the absolute difference of these two least values then write it in the corresponding column difference as shown in the figure. In column $\boldsymbol{D_1}$, **2** is the least value and **3** is the second least value and their absolute difference is **1**. Similarly, for column $\boldsymbol{D_2}, \boldsymbol{D_3}$ and $\boldsymbol{D_4}$, the absolute differences are **2**, **2** and **2** respectively.

# CONTD…

Destination

| Source | D1 | D2 | D3 | D4 | Supply | Row Difference |
|---|---|---|---|---|---|---|
| O1 | 3 | 1 | 7 | 4 | 300 | **2** |
| O2 | 2 | 6 | 5 | 9 | 400 | **3** |
| O3 | 8 | 3 | 3 | 2 | 500 | **1** |
| Demand: | 250 | 350 | 400 | 200 | 1200 | |
| **Column Difference:** | **1** | **2** | **2** | **2** | | |

# CONTD…

- These value of row difference and column difference are also called as penalty. Now select the maximum penalty. The maximum penalty is **3** i.e. row **O2**. Now find the cell with the least cost in row **O2** and allocate the minimum among the supply of the respective row and the demand of the respective column. Demand is smaller than the supply so allocate the column's demand i.e. **250** to the cell. Then cancel the column **D1**

Destination

| Source | D1 | D2 | D3 | D4 | Supply | Row Difference |
|---|---|---|---|---|---|---|
| O1 | 3 | 1 | 7 | 4 | 300 | 2 |
| O2 | 250 / 2 | 6 | 5 | 9 | ~~400~~ 150 | 3 |
| O3 | 8 | 3 | 3 | 2 | 500 | 1 |
| Demand: | ~~250~~ 0 | 350 | 400 | 200 | 1200 | |
| Column Difference: | 1 | 2 | 2 | 2 | | |

# CONTD…

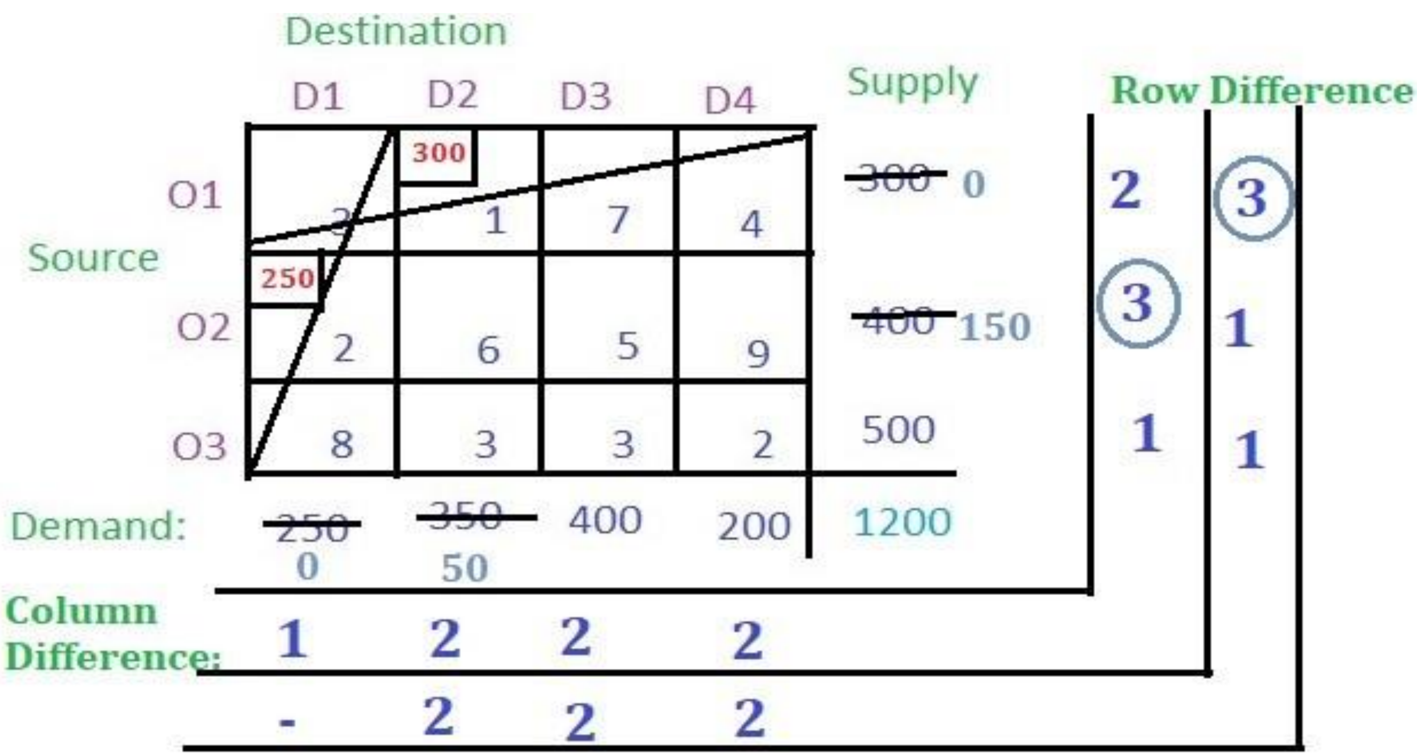
From the remaining cells, find out the row difference and column difference.

Destination; Source

| Source | D1 | D2 | D3 | D4 | Supply | Row Difference | |
|---|---|---|---|---|---|---|---|
| O1 | 3 | 1 | 7 | 4 | 300 | 2 | 3 |
| O2 | 250 / 2 | 6 | 5 | 9 | ~~400~~ 150 | (3) | 1 |
| O3 | 8 | 3 | 3 | 2 | 500 | 1 | 1 |
| Demand: | ~~250~~ 0 | 350 | 400 | 200 | 1200 | | |
| Column Difference: | 1 | 2 | 2 | 2 | | | |
| | - | 2 | 2 | 2 | | | |

# CONTD…

Again select the maximum penalty which is **3** corresponding to row **O1**. The least-cost cell in row **O1** is **(O1, D2)** with cost **1**. Allocate the minimum among supply and demand from the respective row and column to the cell. Cancel the row or column with zero value.



# CONTD…

Now find the row difference and column difference from the remaining cells.

Destination

| Source | D1 | D2 | D3 | D4 | Supply | Row Difference | | |
|---|---|---|---|---|---|---|---|---|
| O1 | 3 | **300** 1 | 7 | 4 | ~~300~~ 0 | 2 | (3) | - |
| O2 | **250** 2 | 6 | 5 | 9 | ~~400~~ 150 | (3) | 1 | 1 |
| O3 | 8 | 3 | 3 | 2 | 500 | 1 | 1 | 1 |
| Demand: | ~~250~~ 0 | ~~350~~ 50 | 400 | 200 | 1200 | | | |
| Column Difference: | 1 | 2 | 2 | 2 | | | | |
| | - | 2 | 2 | 2 | | | | |
| | - | 3 | 2 | 7 | | | | |

# CONTD…

Now select the maximum penalty which is **7** corresponding to column **D4**. The least cost cell in column **D4** is **(O3, D4)** with cost **2**. The demand is smaller than the supply for cell **(O3, D4)**. Allocate **200** to the cell and cancel the column.

Destination

| Source | D1 | D2 | D3 | D4 | Supply | Row Difference | | |
|---|---|---|---|---|---|---|---|---|
| O1 | 2 | 300<br>1 | 7 | 4 | ~~300~~ 0 | 2 | 3 | - |
| O2 | 250<br>2 | 6 | 5 | 9 | ~~400~~ 150 | 3 | 1 | 1 |
| O3 | 8 | 3 | 3 | 200<br>2 | ~~500~~ 300 | 1 | 1 | 1 |
| Demand: | ~~250~~<br>0 | ~~350~~<br>50 | 400 | ~~200~~<br>0 | 1200 | | | |
| Column Difference: | 1 | 2 | 2 | 2 | | | | |
| | - | 2 | 2 | 2 | | | | |
| | - | 3 | 2 | 7 | | | | |

# CONTD…

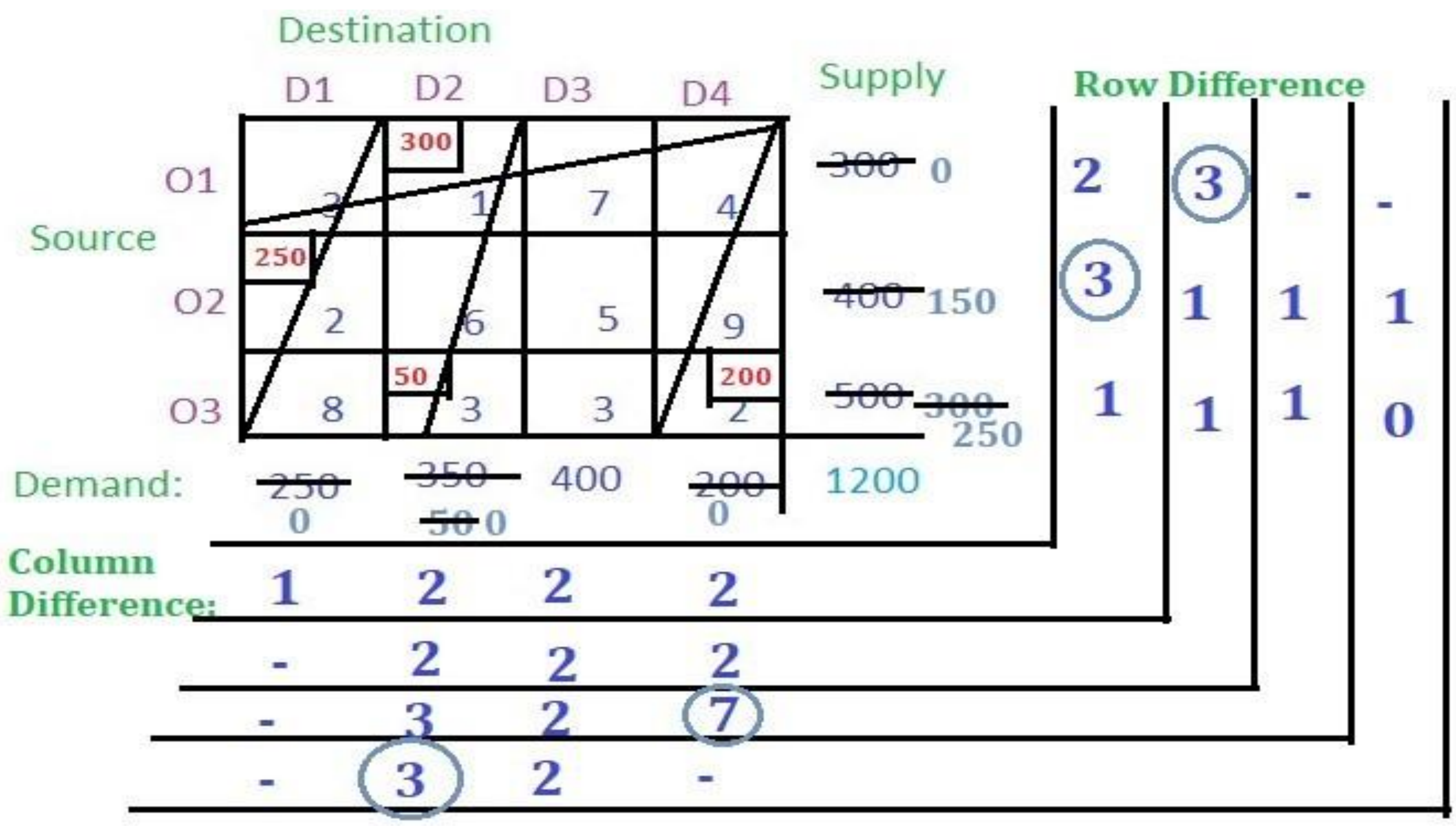
Find the row difference and the column difference from the remaining cells.

| Source \ Destination | D1 | D2 | D3 | D4 | Supply | Row Difference | | | |
|---|---|---|---|---|---|---|---|---|---|
| O1 | 2 | 1 (300) | 7 | 4 | ~~300~~ 0 | 2 | (3) | - | - |
| O2 | 2 (250) | 6 | 5 | 9 | ~~400~~ 150 | (3) | 1 | 1 | 1 |
| O3 | 8 | 3 | 3 | 2 (200) | ~~500~~ 300 | 1 | 1 | 1 | 0 |
| Demand: | ~~250~~ 0 | ~~350~~ 50 | 400 | ~~200~~ 0 | 1200 | | | | |

| Column Difference: | D1 | D2 | D3 | D4 |
|---|---|---|---|---|
| | 1 | 2 | 2 | 2 |
| | - | 2 | 2 | 2 |
| | - | 3 | 2 | (7) |
| | - | 3 | 2 | - |

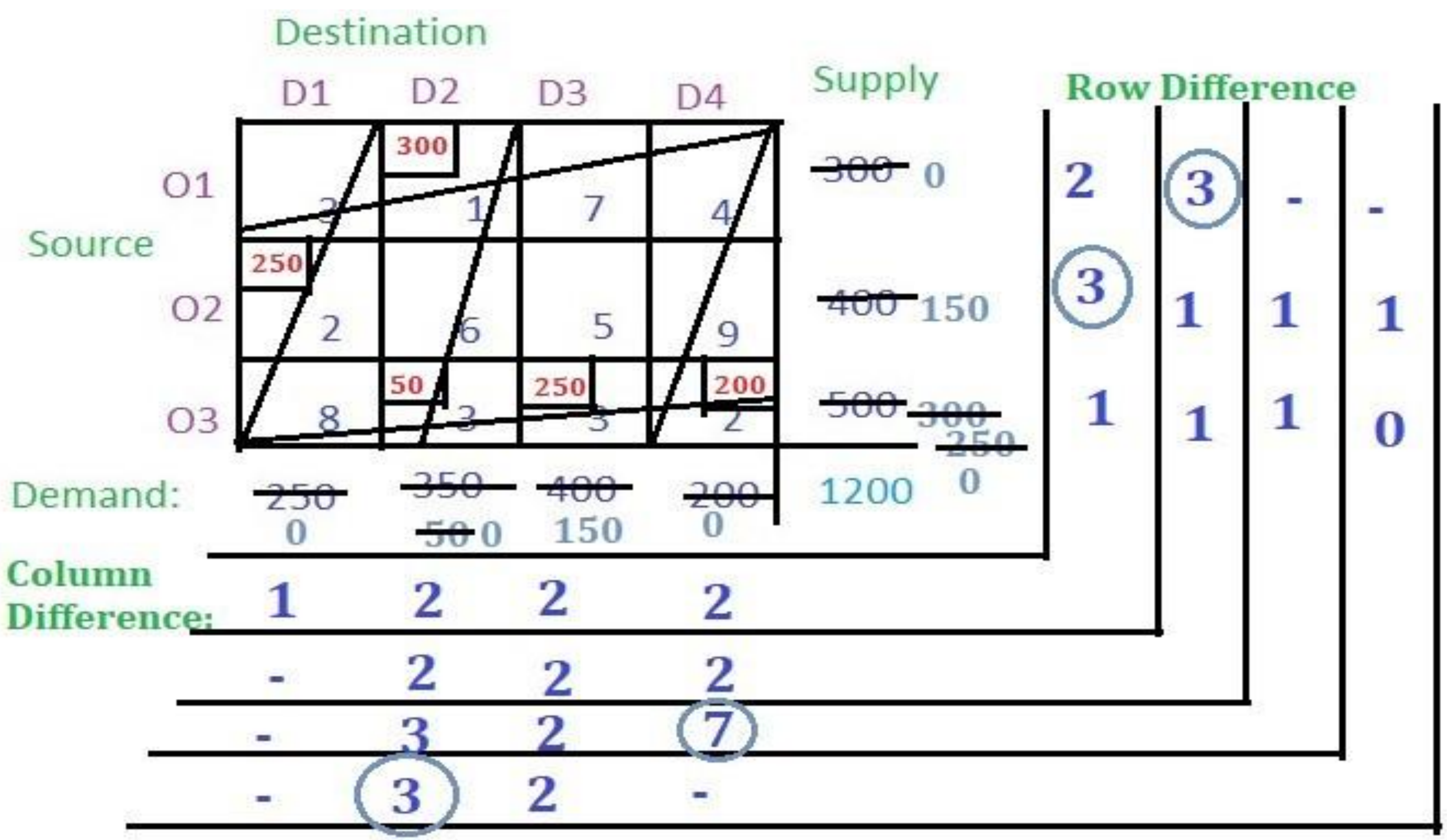
# CONTD…

Now the maximum penalty is **3** corresponding to the column **D2**. The cell with the least value in **D2** is **(O3, D2)**. Allocate the minimum of supply and demand and cancel the column.



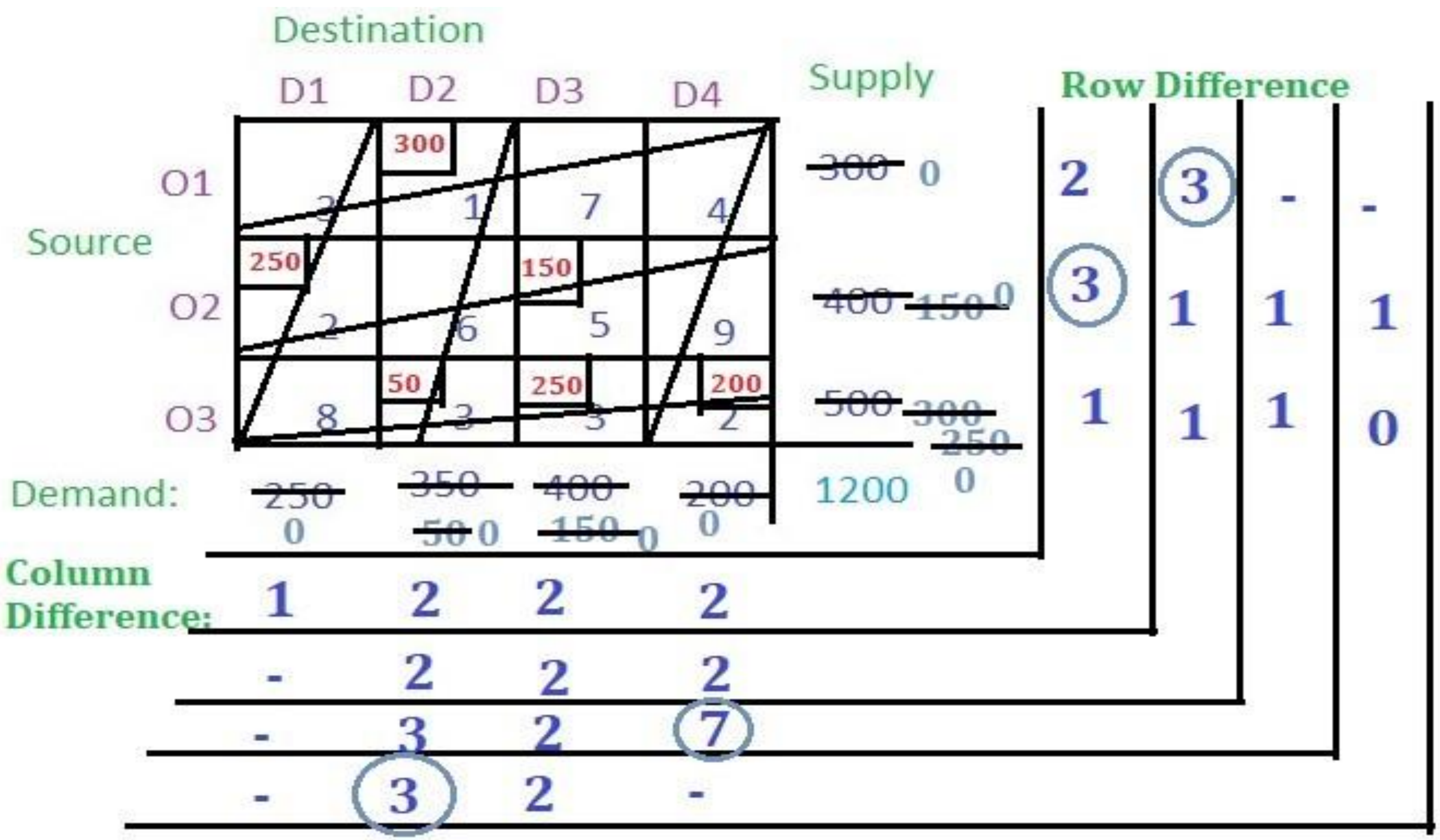
# CONTD…

Now there is only one column so select the cell with the least cost and allocate the value.



# CONTD…

Now there is only one cell so allocate the remaining demand or supply to the cell



# CONTD…

Initial basic feasible solution is given by

- $x_{12} = 300, x_{21} = 250, x_{23} = 150, x_{32} = 50, x_{33} = 250, x_{34} = 200.$ Hence min cost = $(300 \times 1) + (250 \times 2) + (150 \times 5) + (50 \times 3) + (250 \times 3) + (200 \times 2) = 2850$

# Procedure for Optimality Test (MODI Method – UV Method)

There are two phases to solve the transportation problem. In the first phase, the initial basic feasible solution has to be found and the second phase involves optimization of the initial basic feasible solution that was obtained in the first phase.

**Step 1:** Find an initial basic feasible solution using any one of the three methods NWCM, LCM or VAM.

**Step 2:** Find $u_i$ and $v_j$ for rows and columns. To start

a) Assign 0 to $u_i$ or $v_j$ where maximum number of allocation in a row or column respectively.

b) Calculate other $u_i$'s and $v_j$'s using $c_{ij} = u_i + v_j$ , for all occupied cells.

**Step 3:** For all unoccupied cells, calculate $d_{ij} = c_{ij} - (u_i + v_j),.$

**Step 4:** Check the sign of $d_{ij}$

a) If $d_{ij} \geq 0$, then current basic feasible solution is optimal and stop this procedure. In case some $d_{ij} = 0$ then alternative solution exists, with different set allocation and same transportation cost. Now stop this procedure.

b) If one or more $dij < 0$, then the given solution is not an optimal solution and further improvement in the solution is possible.

**Step 5:** Select the unoccupied cell with the largest negative value of $d_{ij}$, and included in the next solution.

**Step 6:** Draw a closed path (or loop) from the unoccupied cell (selected in the previous step). The right angle turn in this path is allowed only at occupied cells and at the original unoccupied cell. Mark (+) and (-) sign alternatively at each corner, starting from the original unoccupied cell.

**Step 7:**

a) Select the minimum value from cells marked with (-) sign of the closed path.

b) Assign this value to selected unoccupied cell (So unoccupied cell becomes occupied cell).

c) Add this value to the other occupied cells marked with (+) sign.

d) Subtract this value to the other occupied cells marked with (-) sign.

**Step 8:** Repeat Step-2 to step-7 until optimal solution is obtained. This procedure stops when all $d_{ij} \geq 0$ for unoccupied cells.

# EXAMPLE…

## Example:

Find Solution using North west corner method, also find optimal solution using mc

Destination

| Source | D1 | D2 | D3 | D4 | Supply($S_i$) |
|---|---|---|---|---|---|
| O1 | 3 | 1 | 7 | 4 | 250 |
| O2 | 2 | 6 | 5 | 9 | 350 |
| O3 | 8 | 3 | 3 | 2 | 400 |
| Demand($D_j$): | 200 | 300 | 350 | 150 | |

Solution:

Here

$$\text{Total Demand} = \text{Total Supply} = 1000$$

Hence problem is balance and can be proceed for initial basic feasible solution.

# CONTD…

Using North west corner initial basic feasible solution is given by

**Destination**

| Source | D1 | D2 | D3 | D4 | Supply($S_i$) |
|---|---|---|---|---|---|
| O1 | (200) 3 | (50) 1 | 7 | 4 | ~~250~~ ~~50~~ 0 |
| O2 | 2 | (250) 6 | (100) 5 | 9 | ~~350~~ ~~100~~ 0 |
| O3 | 8 | 3 | (250) 3 | (150) 2 | ~~400~~ ~~150~~ 0 |
| Demand($D_j$): | ~~200~~ 0 | ~~300~~ ~~250~~ 0 | ~~350~~ ~~250~~ 0 | ~~150~~ 0 | 1000 |

Total transportation cost = $(200 \times 3) + (50 \times 1) + (250 \times 6) + (100 \times 5) + (250 \times 3) + (150 \times 2) = 3700$.

# U-V method to optimize the initial basic feasible solution

Now we use following formula to find $u_i$ and $v_j$,
$u_i + v_j = C_{ij}$ where $C_{ij}$ is the cost value only for the allocated cell.

Before applying the above formula we need to check whether $m + n - 1$ is equal to the total number of allocated cells or not where $m$ is the total number of rows and $n$ is the total number of columns.
In this case $m = 3, n = 4$ and total number of allocated cells is 6 so $m + n - 1 = 6$. The case when $m + n - 1$ is not equal to the total number of allocated cells will be discussed in the later slides(degenerate case).

Now to find the value for u and v we assign any of the three u or any of the four v as 0. Let we assign $u_1 = 0$ in this case. Then using the above formula we will get $v_1 = 3$ as $u_1 + v_1 = 3$ (i.e. $C_{11}$) and $v_2 = 1$ as $u_1 + v_2 = 1$ (i.e. $C_{12}$). Similarly, we have got the value for $v_2 = 3$ so we get the value for $u_2 = 5$ which implies $v_3 = 0$. From the value of $v_3 = 0$ we get $u_3 = 3$ which implies $v_4 = -1$.

# CONTD…

| | $v_1 = 3$ | $v_2 = 1$ | $v_3 = 0$ | $v_4 = -1$ |
|---|---|---|---|---|
| $u_1 = 0$ | 200 / 3 | 50 / 1 | 7 | 4 |
| $u_2 = 5$ | 2 | 250 / 6 | 100 / 5 | 9 |
| $u_3 = 3$ | 8 | 3 | 250 / 3 | 150 / 2 |

Now, compute penalties using the formula $d_{ij} = c_{ij} - (u_i + v_j)$ only for unallocated cells. We have two unallocated cells in the first row, two in the second row and two in the third row. Lets compute this one by one.

1. For $C_{13}, d_{13} = 7 - (0 + 0) = 7$ (here $C_{13} = 7, u_1 = 0$ and $v_3 = 0$)

1.For $C_{14}, d_{14} = 4 - (0 + (-1)) = 5$

2.For $C_{21}, d_{21} = 2 - (5 + 3) = -6$

3.For $C_{24}, d_{24} = 9 - (5 + (-1)) = 5$

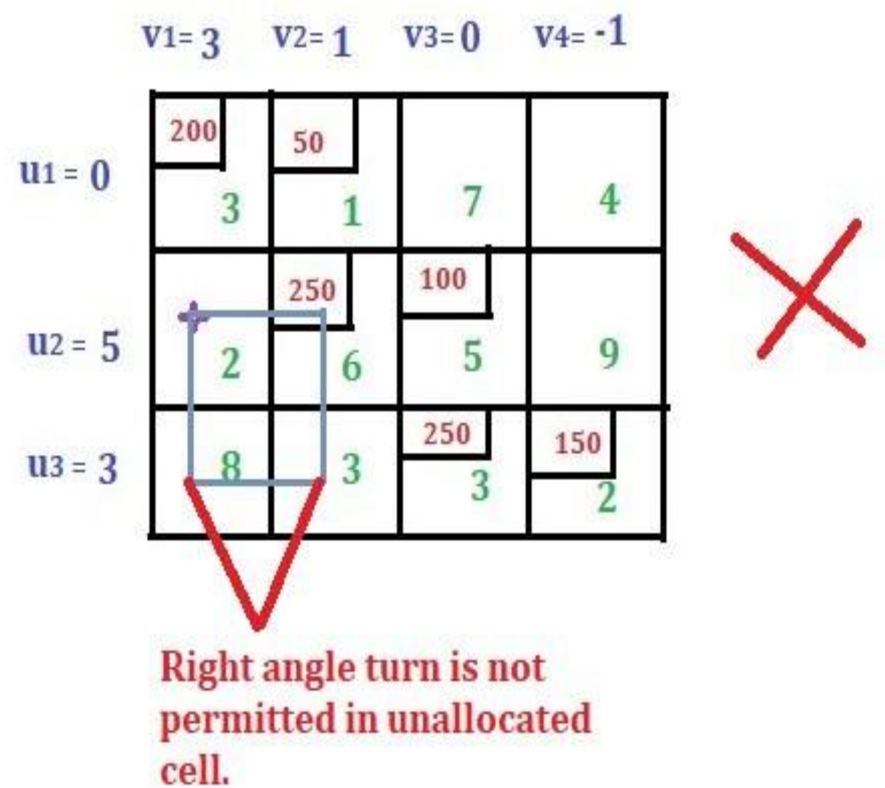
4.For $C_{31}, d_{31} = 8 - (3 + 3) = 2$

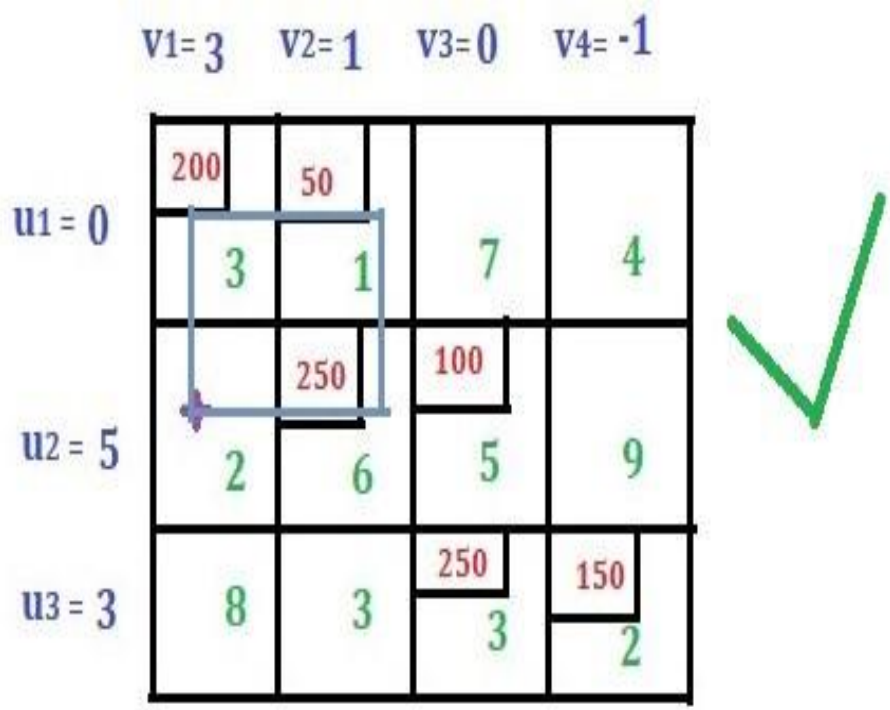
5.For $C_{32}, d_{32} = 3 - (3 + 1) = -1$

# CONTD…

Now most negative value is given by $-6$ for $\boldsymbol{C_{21}}$. Now this cell becomes new basic cell.

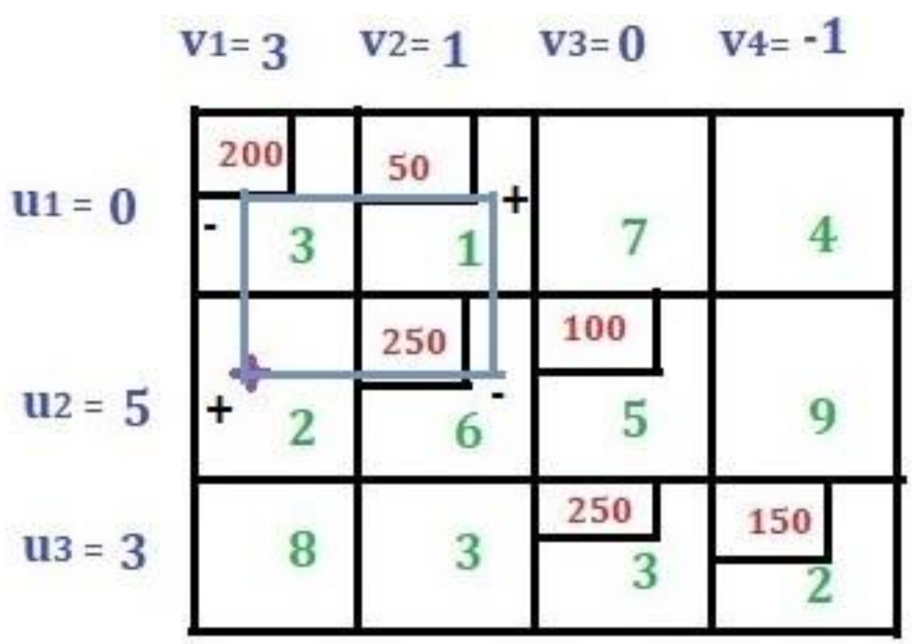
**The rule for drawing closed-path or loop.** Starting from the new basic cell draw a closed-path in such a way that the right angle turn is done only at the allocated cell or at the new basic cell. See the below images:



# CONTD…

Assign alternate plus-minus sign to all the cells with right angle turn (or the corner) in the loop with plus sign assigned at the new basic cell.



# CONTD…

Consider the cells with a negative sign. Compare the allocated value (i.e. 200 and 250 in this case) and select the minimum (i.e. select 200 in this case). Now subtract 200 from the cells with a minus sign and add 200 to the cells with a plus sign. And draw a new iteration. The work of the loop is over and the new solution looks as shown below.

| | | | |
|---|---|---|---|
| 3 | 250<br>1 | 7 | 4 |
| 200<br>2 | 50<br>6 | 100<br>5 | 9 |
| 8 | 3 | 250<br>3 | 150<br>2 |

# CONTD…

Check the total number of allocated cells is equal to $(m + n - 1)$. Again find $u$ values and $v$ values using the formula $u_i + v_j = C_{ij}$ where $C_{ij}$ is the cost value only for allocated cell. Assign $u_1 = 0$ then we get $v_2 = 1$. Similarly, we will get following values for $u_i$ and $v_j$.

| | v1=-3 | v2=1 | v3=0 | v4=-1 |
|---|---|---|---|---|
| u1=0 | 3 | 250 / 1 | 7 | 4 |
| u2=5 | 200 / 2 | 50 / 6 | 100 / 5 | 9 |
| u3=3 | 8 | 3 | 250 / 3 | 150 / 2 |

Find the penalties for all the unallocated cells using the formula $d_{ij} = C_{ij} - (u_i + v_j)$

1.For $C_{11}, d_{11} = 3 - \left(0 + (-3)\right) = 6$

2.For $C_{13}, d_{13} = 7 - (0 + 0) = 7$

3.For $C_{14}, d_{14} = 4 - (0 + (-1)) = 5$

4.For $C_{24}, d_{24} = 9 - (5 + (-1)) = 5$

5.For $C_{31}, d_{31} = 8 - (0 + (-3)) = 11$

6.For $C_{32}, d_{32} = 3 - (1 + 3) = -1$

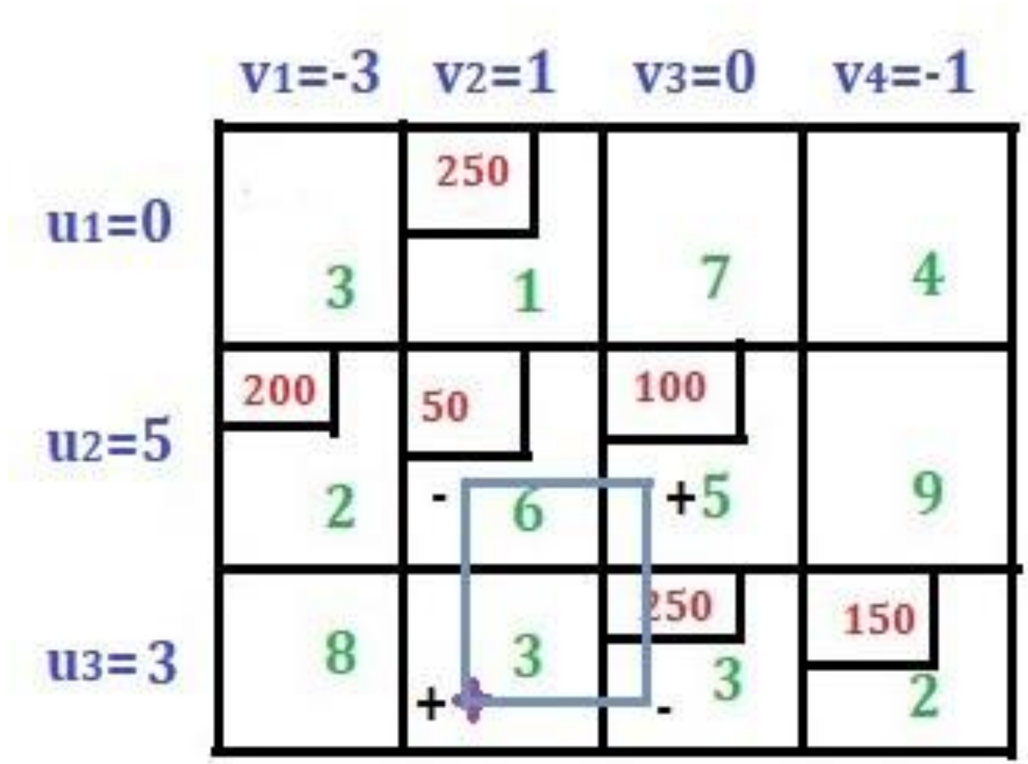
# CONTD…

There is one negative value i.e. $-1$ for $C_{32}$. Now this cell becomes new basic cell.

| | $v_1=-3$ | $v_2=1$ | $v_3=0$ | $v_4=-1$ |
|---|---|---|---|---|
| $u_1=0$ | 3 | 1 (250) | 7 | 4 |
| $u_2=5$ | 2 (200) | 6 (50) | 5 (100) | 9 |
| $u_3=3$ | 8 | 3 (+) | 3 (250) | 2 (150) |

# CONTD…

Now draw a loop starting from the new basic cell. Assign alternate plus and minus sign with new basic cell assigned as a plus sign.



# CONTD…

Select the minimum value from allocated values to the cell with a minus sign. Subtract this value from the cell with a minus sign and add to the cell with a plus sign. Now the solution looks as shown in the image below:

| | | | |
|---|---|---|---|
| 3 | 250<br>1 | 7 | 4 |
| 200<br>2 | 6 | 150<br>5 | 9 |
| 8 | 50<br>3 | 200<br>3 | 150<br>2 |

# CONTD…

Check if the total number of allocated cells is equal to $(m + n - 1)$. Find $u$ and $v$ values as above.

| | $v_1=-2$ | $v_2=1$ | $v_3=1$ | $v_4=0$ |
|---|---|---|---|---|
| $u_1=0$ | 3 | 250 / 1 | 7 | 4 |
| $u_2=4$ | 200 / 2 | 6 | 150 / 5 | 9 |
| $u_3=2$ | 8 | 50 / 3 | 200 / 3 | 150 / 2 |

Now again find the penalties for the unallocated cells as above.

1.For $d_{11} = 3 - (0 + (-2)) = 5$

2.For $d_{13} = 7 - (0 + 1) = 6$

3.For $d_{14} = 4 - (0 + 0) = 4$

4.For $d_{22} = 6 - (4 + 1) = 1$

5.For $d_{24} = 9 - (4 + 0) = 5$

6.For $d_{31} = 8 - (2 + (-2)) = 8$

# CONTD…

All the penalty values are +ve values. So the optimality is reached.

Hence optimal solution is given by Type equation here.

$x_{12} = 250, x_{21} = 200, x_{23} = 150, \quad x_{32} = 50, x_{33} = 200, x_{34} = 150$

Now, find the total cost $= (250 \times 1) + (200 \times 2) + (150 \times 5) + (50 \times 3) + (200 \times 3) + (150 \times 2) = 2450$

# Unbalanced Transportation Problem

**Example** Solve following transportation problem

| Origin | | Destination | | | | Supply |
|---|---|---|---|---|---|---|
| | | $D_1$ | $D_2$ | $D_3$ | $D_4$ | |
| | $O_1$ | 7 | 6 | 6 | 6 | 80 |
| | $O_2$ | 5 | 7 | 6 | 7 | 100 |
| | $O_3$ | 8 | 5 | 8 | 6 | 50 |
| Demand | | 50 | 40 | 60 | 40 | |

## Solution:

$$Total\ supply = \sum a_i = 230$$

$$\boldsymbol{Total\ demand = \sum b_i = 190}$$

# Contd..

Since $Total\ supply \neq Total\ demand$. Hence given problem is unbalanced.

Now we have to convert this problem into balanced transportation problem.

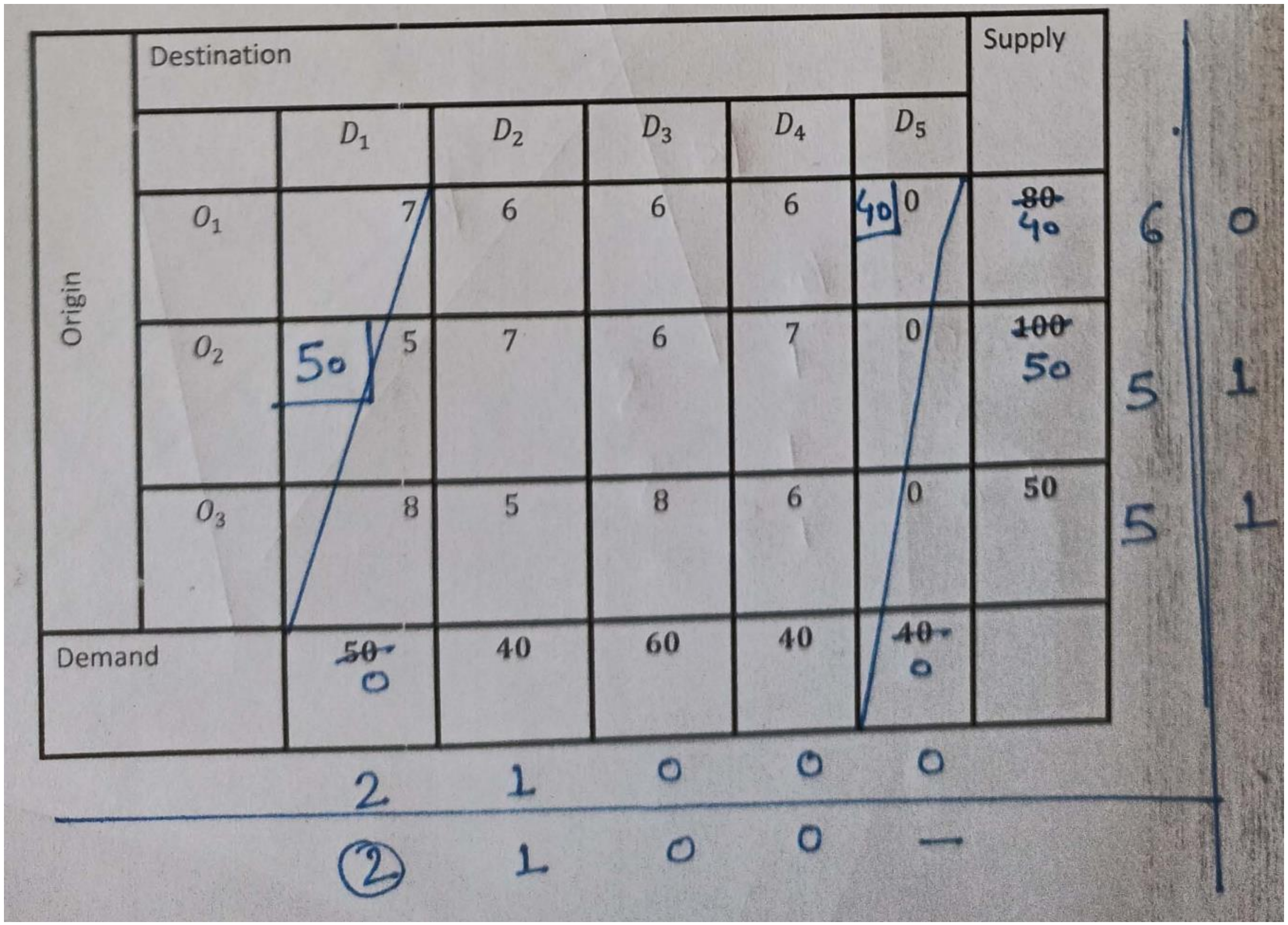
Since $Total\ supply - Total\ Demand = 40$. Hence for converting it into balance transportation problem we have to use dummy destination $D_5$ in the transportation table with demand 40 and transportation cost zero. t. Thus, the problem becomes balanced, i.e., the total capacity and total requirement are equal. The balanced problem is as follows:

| Origin | | Destination | | | | | Supply |
|---|---|---|---|---|---|---|---|
| | | $D_1$ | $D_2$ | $D_3$ | $D_4$ | $D_5$ | |
| | $O_1$ | 7 | 6 | 6 | 6 | 0 | **80** |
| | $O_2$ | 5 | 7 | 6 | 7 | 0 | **100** |
| | $O_3$ | 8 | 5 | 8 | 6 | 0 | **50** |
| Demand | | **50** | **40** | **60** | **40** | **40** | |

# CONTD….

| Origin | $D_1$ | $D_2$ | $D_3$ | $D_4$ | $D_5$ | Supply | |
|---|---|---|---|---|---|---|---|
| $O_1$ | 7 | 6 | 6 | 6 | 40 \| 0 | ~~80~~ 40 | (6) |
| $O_2$ | 5 | 7 | 6 | 7 | 0 | 100 | 5 |
| $O_3$ | 8 | 5 | 8 | 6 | 0 | 50 | 5 |
| Demand | 50 | 40 | 60 | 40 | ~~40~~ 0 | | |
| | 2 | 1 | 0 | 0 | 0 | | |

# CONTD….



| Origin | $D_1$ | $D_2$ | $D_3$ | $D_4$ | $D_5$ | Supply | | |
|---|---|---|---|---|---|---|---|---|
| $O_1$ | 7 | 6 | 6 | 6 | 40 \| 0 | ~~80~~ 40 | 6 | 0 |
| $O_2$ | 50 \| 5 | 7 | 6 | 7 | 0 | ~~100~~ 50 | 5 | 1 |
| $O_3$ | 8 | 5 | 8 | 6 | 0 | 50 | 5 | 1 |
| Demand | ~~50~~ 0 | 40 | 60 | 40 | ~~40~~ 0 | | | |
| | 2 | 1 | 0 | 0 | 0 | | | |
| | (2) | 1 | 0 | 0 | — | | | |

# CONTD….

| Origin | $D_1$ | $D_2$ | $D_3$ | $D_4$ | $D_5$ | Supply | | | |
|---|---|---|---|---|---|---|---|---|---|
| $O_1$ | 7 | 6 | 6 | 6 | [40] 0 | ~~80~~ 40 | 6 | 0 | 0 |
| $O_2$ | [50] 5 | 7 | [50] 6 | 7 | 0 | ~~100~~ ~~50~~ 0 | 5 | 1 | (1) |
| $O_3$ | 8 | 5 | 8 | 6 | 0 | 50 | 5 | 1 | 1 |
| Demand | ~~50~~ 0 | 40 | ~~60~~ 10 | 40 | ~~40~~ 0 | | | | |
| | 2 | 1 | 0 | 0 | 0 | | | | |
| | 2 | 1 | 0 | 0 | — | | | | |
| | — | 1 | 0 | 0 | — | | | | |

# CONTD….

| Origin \ Destination | $D_1$ | $D_2$ | $D_3$ | $D_4$ | $D_5$ | Supply | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $O_1$ | 7 | 6 | [10] 6 | 6 | [40] 0 | ~~80~~ ~~40~~ 30 | 6 | 0 | 0 | 0 |
| $O_2$ | [50] 5 | 7 | [50] 6 | 7 | 0 | ~~100~~ ~~50~~ 0 | 5 | 1 | 1 | |
| $O_3$ | 8 | 5 | 8 | 6 | 0 | 50 | 5 | 1 | 1 | 1 |
| Demand | ~~60~~ 0 | 40 | ~~60~~ ~~10~~ 0 | 40 | ~~40~~ 0 | | | | | |
| | 2 | 1 | 0 | 0 | 0 | | | | | |
| | 2 | 1 | 0 | 0 | – | | | | | |
| | – | 1 | 0 | 0 | – | | | | | |
| | – | 1 | (2) | 0 | – | | | | | |

# CONTD….

| Origin | D1 | D2 | D3 | D4 | D5 | Supply | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| O1 | 7 | 6 | [10] 6 | [30] 6 | [40] 0 | ~~80~~ ~~40~~ 30 | 6 | 0 | 0 | 0 | 0 |
| O2 | [50] 5 | 7 | [50] 6 | 7 | 0 | 100 | 5 | 1 | 1 | — | — |
| O3 | 8 | [40] 5 | 8 | [10] 6 | 0 | 50 | 5 | 1 | 1 | 1 | (1) |
| Demand | ~~50~~ 0 | ~~40~~ 0 | ~~60~~ ~~10~~ 0 | 40 | ~~40~~ 0 | | | | | | |
| | 2 | 1 | 0 | 6 | 0 | | | | | | |
| | 2 | 1 | 0 | 0 | — | | | | | | |
| | — | 1 | 0 | 0 | — | | | | | | |
| | — | 1 | 2 | 0 | — | | | | | | |
| | — | 1 | — | 0 | — | | | | | | |

Here $m + n - 1 = 7$. Number of occupied cell $= 7$ which is equal to $m + n - 1$. Hence initial solution is nondegenerate and we can proceed for Modi Method for Optimality Test .

# CONTD…

| Origin | Destination $v_1=5$ $D_1$ | $v_2=5$ $D_2$ | $v_3=6$ $D_3$ | $u_4=6$ $D_4$ | $u_5=0$ $D_5$ | Supply | |
|---|---|---|---|---|---|---|---|
| $O_1$ | 7 | 6 | [10] 6 | [30] 6 | [40] 0 | 80 | $u_1=0$ |
| $O_2$ | [50] 5 | 7 | [50] 6 | 7 | 0 | 100 | $u_2=0$ |
| $O_3$ | 8 | [40] 5 | 8 | [10] 6 | 0 | 50 | $u_3=0$ |
| Demand | 50 | 40 | 60 | 40 | 40 | | |

$$d_{11} = 7 - (0+5) = 2$$
$$d_{12} = 6 - (0+5) = 1$$
$$d_{22} = 7 - (0+5) = 2$$
$$d_{24} = 7 - (0+6) = 1$$
$$d_{25} = 0 - (0+0) = 0$$
$$d_{31} = 8 - (0+5) = 3$$
$$d_{33} = 8 - (0+6) = 2$$
$$d_{35} = 0 - (0+0) = 0$$

Since all the $d_{ij} > 0$. Hence this solution is optimal solution.

$$x_{13} = 10, x_{14} = 30, x_{21} = 50, x_{23} = 50, x_{32} = 40, x_{34} = 10$$

Hence min cost $= 10 \times 6 + 30 \times 6 + 50 \times 5 + 50 \times 6 + 40 \times 5 + 10 \times 6 = 1050$

# ASSIGNMENT PROBLEM

## Special Cases of Linear Programming Problem

# Assignment Problem (AP)

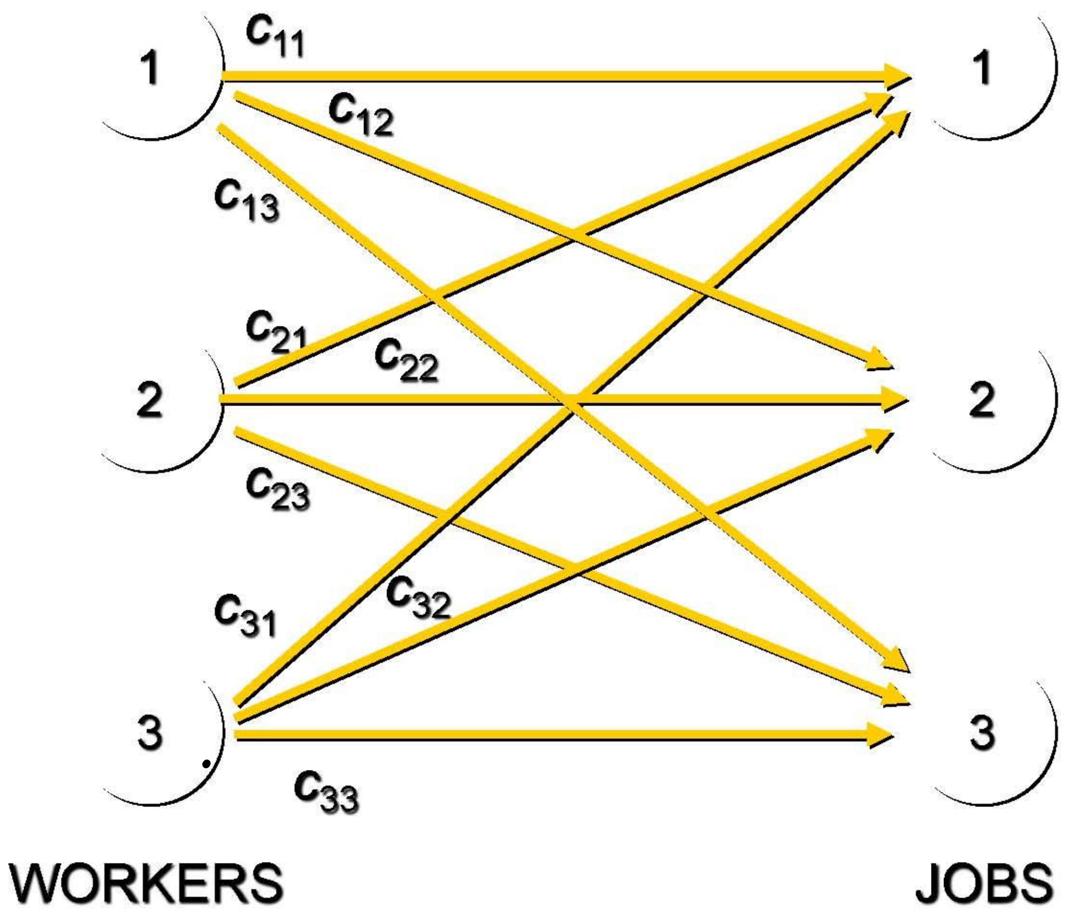
- In many business situations, management needs to assign personnel to jobs, jobs to machines, machines to job locations, or salespersons to territories.
- An assignment problem may be considered as a special type of transportation problem in which the number of sources and destinations are equal. The capacity of each source as well as the requirement of each destination is taken as 1.
- Consider the situation of assigning $n$ jobs to $n$ machines.

# CONTD…

- When a job $i\ (= 1,2, \ldots, n)$ is assigned to machine $j\ (= 1,2, \ldots n)$ that incurs a cost $C_{ij}$.
- The objective is to assign the jobs to machines at the least possible total cost.
- Here, jobs represent "sources" and machines represent "destinations."
- The supply available at each source is 1 unit and demand at each destination is 1 unit.

# CONTD…

- Network Representation



# ASSIGNMENT PROBLEM IN MATRIX FORM

| Workers | Jobs | | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | Demand |
| | 1 | $C_{11}$ | $C_{12}$ | $C_{13}$ | 1 |
| | 2 | $C_{21}$ | $C_{22}$ | $C_{23}$ | 1 |
| | 3 | $C_{31}$ | $C_{32}$ | $C_{33}$ | 1 |
| | Requirement | 1 | 1 | 1 | |

•

# ASSIGNMENT PROBLEM WITH $n$ PERSONS AND $n$ JOBS

| Person | JOB | | | | | |
|---|---|---|---|---|---|---|
| | | $J_1$ | $J_2$ | … | $J_n$ | supply |
| | $P_1$ | $C_{11}$ | $C_{12}$ | … | $C_{1n}$ | 1 |
| | $P_2$ | $C_{21}$ | $C_{22}$ | … | $C_{2n}$ | 1 |
| | … | … | … | … | … | … |
| | $P_n$ | $C_{n1}$ | $C_{n2}$ | … | $C_{nn}$ | 1 |
| | **Demand** | 1 | 1 | … | 1 | |

The assignment model can be expressed mathematically as follows:

Let $x_{ij}$ be assignment of $Person\ P_i$ to Job $J_j$ such that

$$x_{ij} = \begin{cases} 1, & \text{if the } i_{th} \text{ person to be assigned with } \mathrm{j_{th}} \text{ job} \\ 0, & \text{Otherwise} \end{cases}$$

# MATHEMATICAL FORMULATION

There are $n$ persons and $n$ jobs. The problem is to assign each person for one and only one job and each jobs requires only one person. The cost of assigning person $P_i$ to Job $J_j$ is $C_{ij}$. The supply available to each person is one ie only one person should be assigned with one job. Demand of person for each job is one ie only one job to be assigned for one person.

# CONTD…

Linear Programming Formulation is given by

$$Min\ z = \sum_{i=1}^{m}\sum_{j=1}^{n} C_{ij}x_{ij}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad \text{for } i = 1, 2, \ldots, n,$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad \text{for } j = 1, 2, \ldots, n,$$

$$x_{ij} \geq 0, \qquad \text{for all } i \text{ and } j$$

$$(x_{ij} \text{ binary}, \qquad \text{for all } i \text{ and } j).$$

Note: A modification to the right-hand side of the first constraint set can be made if a worker is permitted to work more than 1 job.

# TOTAL NUMBER OF ASSIGNMENT

| Person | Job | | |
|---|---|---|---|
| | $J_1$ | $J_2$ | $J_3$ |
| $P_1$ | $C_{11}$ | $C_{12}$ | $C_{13}$ |
| $P_2$ | $C_{21}$ | $C_{22}$ | $C_{23}$ |
| $P_3$ | $C_{31}$ | $C_{32}$ | $C_{33}$ |

$P_1 \rightarrow J_1$ ⇒ $P_2 \rightarrow J_2, P_3 \rightarrow J_3,$ Total cost $= C_{11} + C_{22} + C_{33}$

$P_2 \rightarrow J_3, P_3 \rightarrow J_2,$ Total cost $= C_{11} + C_{23} + C_{32}$

$P_1 \rightarrow J_2$ ⇒ $P_2 \rightarrow J_1, P_3 \rightarrow J_3,$ Total cost $= C_{12} + C_{21} + C_{33}$

$P_2 \rightarrow J_3, P_3 \rightarrow J_1$ Total cost $= C_{12} + C_{23} + C_{31}$

$P_1 \rightarrow J_3$ ⇒ $P_2 \rightarrow J_1, P_3 \rightarrow J_2,$ Total cost $= C_{13} + C_{21} + C_{32}$

$P_2 \rightarrow J_2, P_3 \rightarrow J_1,$ Total cost $= C_{13} + C_{22} + C_{31}$

# TOTAL NUMBER OF ASSIGNMENT

$P_1 \rightarrow$ 3 option for job

$P_2 \rightarrow$ 2 option for job

$P_3 \rightarrow$ 1 option for job

$\text{Total Option} = 3 \times 2 \times 1 = 6 = 3!$

$\text{Hence total no. of assignment} = 3!$

Hence an assignment problem with $n$ persons and $n$ jobs is represented by square matrix of order $n$, then possible way of making assignment is $n!$. If we enumerate all these $n!$ alternatives and evaluate the assignment cost of each one of them and select the assignment with minimum cost. The problem would be solved but this method is very slow and time consuming for small values of $n$ and hence it is not suitable at all. However a much and more efficient method of solving such problem is available . This method is known as Hungarian method for solving assignment problem.

# Hungarian Method for Solving Assignment Problem

- The Hungarian method solves minimization assignment problems with $n$ workers and $m$ jobs.
- Special considerations can include:
  - Number of workers does not equal the number of jobs — add dummy workers/jobs with 0 assignment costs as needed
  - worker $i^{th}$ cannot do job $j^{th}$ assign $c_{ij} = +M$
  - Maximization Assignment Problem — Convert into minimization problem by subtracting all the values in the cost matrix from the largest value in the matrix before beginning the Hungarian method

# Hungarian Method for Solving Assignment Problem

Step 1: Identify the smallest element in each row of the given cost matrix and subtract it from each element of that row.

Step 2: In the reduced matrix obtained from step 1, Identify the smallest element in each column and then subtract it from each element of that column. Now each row and column have at least one zero element.

Step 3: The procedure of making assignment as follows

(a) First round for making assignment

- Identify the rows successively from top to bottom until a row with exactly one zero element is found. Make an assignment to this single zero by making square (□) around it, then cross off (×) all other zeros in corresponding column.
- Apply similar procedure to the columns from left to right.

(b) Second round for making assignment

- ❑ If a row and/or column has two or more unmarked zeros one can by chosen by inspection, then choose zero element arbitrary for the assignment.

(c) Repeat step (a) and (b) successively until all the zero in the cost matrix are either marked with square (□) or crossed off and following situation arise

Step 4:

(a) If there is exactly one assignment (□) to each row and each column, then it is optimal . The total cost associated with solution is obtained by adding the original cost in the assigned cells (□)

(b) If a zero element in a row or column are chosen arbitrary for assignment in step 3(b) and condition 4(a) achieved , there exist an alternative solution.

(c) If there is no assignment in a row or column ie total no. of assignment are less than the number of rows /columns in square cost matrix, then this solution is not optimal. Now we proceed to step (5)

Step 5: Revise the cost matrix:

Draw a set of horizontal and vertical lines to cover all the zeros in revised cost matrix from step 3, by using following procedure.

(a) For each row in which no assignment made, mark a tick (✓)

(b) Examine the marked rows. If any zero element is present in these rows , mark a tick (✓) to respective columns containing zeros.

(c) Examine marked columns. If any assigned zero element is present in these columns , tick (✓) the respective rows containing assigned zeros

(d) Repeat this process until no more rows or columns can be marked.

(e) Draw a straight line through each marked column and each unmarked row. If no. of lines drawn is equal to number of rows (columns) then current solution is optimal otherwise go to step 6.

# Hungarian Method for Solving Assignment Problem

Step 6: Determine the smallest element of uncovered number (call it *d*) after drawing the lines as given in step 5.

- Subtract *d* from each uncovered numbers.
- Add *d* to numbers covered by two lines.
- Numbers covered by one line remain the same.
- This process gives modified cost matrix with more zeros.
- Then, GO TO STEP 3.

# CONTD…

## Example 1:

| | | Machine | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | |
| | 1 | 5 | 7 | 9 | 1 |
| Job | 2 | 14 | 10 | 12 | 1 |
| | 3 | 15 | 13 | 16 | 1 |
| | | 1 | 1 | 1 | |

How should the contractors be assigned to minimize total costs?

| | | | | | |
|---|---|---|---|---|---|
| | | Machine | | | |
| | | 1 | 2 | 3 | |
| | 1 | 5 | 7 | 9 | 1 |
| Job | 2 | 14 | 10 | 12 | 1 |
| | 3 | 15 | 13 | 16 | 1 |
| | | 1 | 1 | 1 | |

•

# Solution:

- Step 1: For each row, subtract the minimum number in that row from all numbers in that row.
- Step 2: For each column, subtract the minimum number in that column from all numbers in that column.
- 

| | | Machine | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| Job | 1 | **5** | 7 | 9 |
| | 2 | 14 | **10** | 12 |
| | 3 | 15 | **13** | 16 |

| | | Machine | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| Job | 1 | 0 | 2 | 4 |
| | 2 | 4 | 0 | **2** |
| | 3 | 2 | 0 | 3 |

# CONTD…

Step 3: Draw the minimum number of lines to cover all zeroes.

| | Machine | | |
|---|---|---|---|
| | 1 | 2 | 3 |
| 1 | [0] | 2 | 2 |
| 2 | 4 | ~~0~~ | [0] |
| 3 | 2 | [0] | 1 |

Since we have one assigned zero in each row and each column. Hence this table gives optimal solution. Now Assignments are made at assigned zero values. Therefore, we assign job 1 to machine 1; job 2 to machine 3, and job 3 to machine 2.

Total cost is $5 + 12 + 13 = 30$.

It is not always possible to obtain a feasible assignment as here.

# EXAMPLE

**Example:** Four Machines are to assigned to four different Jobs. The cost (in rupees) of producing $i_{th}$ machine on the $j_{th}$ job is given below

| Machine | Jobs | | | | |
|---|---|---|---|---|---|
| | | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
| | $M_1$ | 14 | 5 | 8 | 7 |
| | $M_2$ | 2 | 12 | 6 | 5 |
| | $M_3$ | 7 | 8 | 3 | 9 |
| | $M_4$ | 2 | 4 | 6 | 10 |

Determine how Machines should be assigned to jobs to <u>minimize</u> cost.

# Solution:

- Step 1: For each row, subtract the minimum number in that row from all numbers in that row.

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| | | | | |
| $M_1$ | 14 | **5** | 8 | 7 |
| $M_2$ | **2** | 12 | 6 | 5 |
| $M_3$ | 7 | 8 | **3** | 9 |
| $M_4$ | **2** | 4 | 6 | 10 |

**Row Reduction**

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| | | | | |
| $M_1$ | 9 | 0 | 3 | 2 |
| $M_2$ | 0 | 10 | 4 | 3 |
| $M_3$ | 4 | 5 | 0 | 6 |
| $M_4$ | 0 | 2 | 4 | 8 |

# Solution:

- Step 2: For each column, subtract the minimum number in that column from all numbers in that column.

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| $M_1$ | 9 | 0 | 3 | 2 |
| $M_2$ | 0 | 10 | 4 | 3 |
| $M_3$ | 4 | 5 | 0 | 6 |
| $M_4$ | 0 | 2 | 4 | 8 |

**Column Reduction**

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| $M_1$ | 9 | 0 | 3 | 0 |
| $M_2$ | 0 | 10 | 4 | 1 |
| $M_3$ | 4 | 5 | 0 | 4 |
| $M_4$ | 0 | 2 | 4 | 6 |

# CONTD…

Step 3: The procedure of making assignment as follows

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| $M_1$ | 9 | [0] | 3 | ~~0~~ |
| $M_2$ | [0] | 10 | 4 | 1 |
| $M_3$ | 4 | 5 | [0] | 4 |
| $M_4$ | ~~0~~ | 2 | 4 | 6 |

Step 4: Since each rows and columns have not assigned zero. Hence this not optimal. Now go to step 5

# CONTD…

Step 5: Draw the minimum number of lines to cover all zeroes. If 4 lines are required, then an optimal solution is available among the covered zeros in the matrix. Otherwise, continue to Step 6.

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ | |
|---|---|---|---|---|---|
| $M_1$ | 9 | 0 | 3 | 0 | |
| $M_2$ | 0 | 10 | 4 | 1 | ✓ |
| $M_3$ | 4 | 5 | 0 | 4 | |
| $M_4$ | 0 | 2 | 4 | 6 | ✓ |
| | ✓ | | | | |

We have 3 < 4 lines covering all the zero , so continue to Step 6

# CONTD…

Step 4: Step 6: Determine the smallest element of uncovered number (call it *d*) after drawing the lines as given in step 5.

- Subtract *d* from each uncovered numbers.
- Add *d* to numbers covered by two lines.
- Numbers covered by one line remain the same.
- This process gives modified cost matrix with more zeros.
- Then, GO TO STEP 3.

Here 1 is the smallest element from uncovered elements. Subtract 1 from all uncovered element and add 1 to the element lying in the intersection of the vertical and horizontal line ie numbers covered by two lines.

# CONTD…

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| $M_1$ | 9 | 0 | 3 | 0 |
| $M_2$ | 0 | 10 | 4 | 1 |
| $M_3$ | 4 | 5 | 0 | 4 |
| $M_4$ | 0 | 2 | 4 | 6 |

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| $M_1$ | **10** | 0 | 3 | 0 |
| $M_2$ | 0 | **9** | **3** | **0** |
| $M_3$ | **5** | 5 | 0 | 4 |
| $M_4$ | 0 | **1** | **3** | **5** |

# CONTD…

## Repeat the Step 3

| | $J_1$ | $J_2$ | $J_3$ | $J_4$ |
|---|---|---|---|---|
| $M_1$ | 10 | [0] | 3 | ~~0~~ |
| $M_2$ | ~~0~~ | 9 | 3 | [0] |
| $M_3$ | 5 | 5 | [0] | 4 |
| $M_4$ | [0] | 1 | 3 | 5 |

Since we have assigned zero in each row and column. Hence this table will give the optimal solution and we assign, $\boldsymbol{M_1} \rightarrow \boldsymbol{J_2}, \boldsymbol{M_2} \rightarrow \boldsymbol{J_4}, \boldsymbol{M_3} \rightarrow \boldsymbol{J_3}, \boldsymbol{M_4} \rightarrow \boldsymbol{J_1}$.

Min Cost$= 5 + 5 + 3 + 2 = 15$

# Example

A firm has four plants each of which can manufacture any of the four products 1, 2, 3 and 4. Production cost differ from plant to plant as do sales revenue. The revenue and cost data given below. Determine from the following data which product should each plant produce in order to maximize the profit.

**Sales Revenue (in ’000 Rs.)**

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 50 | 68 | 49 | 62 |
| | **B** | 60 | 70 | 51 | 74 |
| | **C** | 55 | 67 | 53 | 70 |
| | **D** | 58 | 65 | 54 | 69 |

**Production Cost (in ’000 Rs.)**

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 49 | 60 | 45 | 61 |
| | **B** | 55 | 63 | 45 | 69 |
| | **C** | 52 | 62 | 49 | 68 |
| | **D** | 55 | 64 | 48 | 66 |

# Solution

Since profit =Sales Revenue-Product cost. So profit matrix is given by

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 1 | 8 | 4 | 1 |
| | **B** | 5 | 7 | 6 | 5 |
| | **C** | 3 | 5 | 4 | 2 |
| | **D** | 3 | 1 | 6 | 3 |

Now we change the maximization problem in minimization cost. Subtracting each element of the Profit matrix from the largest element of 8 of the matrix . Hence equivalent loss matrix is given by

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 7 | 0 | 4 | 7 |
| | **B** | 3 | 1 | 2 | 3 |
| | **C** | 5 | 3 | 4 | 6 |
| | **D** | 5 | 7 | 2 | 5 |

# Solution

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 7 | 0 | 4 | 7 |
| | **B** | 3 | 1 | 2 | 3 |
| | **C** | 5 | 3 | 4 | 6 |
| | **D** | 5 | 7 | 2 | 5 |

**Row Reduction**

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 7 | 0 | 4 | 7 |
| | **B** | 2 | 0 | 1 | 2 |
| | **C** | 2 | 0 | 1 | 3 |
| | **D** | 3 | 5 | 0 | 3 |

# Solution

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 7 | 0 | 4 | 7 |
| | **B** | 2 | 0 | 1 | 2 |
| | **C** | 2 | 0 | 1 | 3 |
| | **D** | 3 | 5 | 0 | 3 |

**Column Reduction**

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 5 | 0 | 4 | 5 |
| | **B** | 0 | 0 | 1 | 0 |
| | **C** | 0 | 0 | 1 | 1 |
| | **D** | 1 | 5 | 0 | 1 |

# Solution

| | Product | | | | |
|---|---|---|---|---|---|
| Plant | | **1** | **2** | **3** | **4** |
| | **A** | 5 | [0] | 4 | 5 |
| | **B** | ~~0~~ | ~~0~~ | 1 | [0] |
| | **C** | [0] | ~~0~~ | 1 | 1 |
| | **D** | 1 | 5 | [0] | 1 |

In above table there is an assignment in each row and column.

Hence optimal assignment for maximizing profit is

$$A \to 2, B \to 4, C \to 1, D \to 4$$

Hence maximum profit $= (8 + 5 + 3 + 6) \times 1000 = \text{Rs.}\, 220000$

# UNBALANCED ASSIGNMENT PROBLEM

**Example:** A company faced with problem of assigning six different machines to five different jobs. The cost are estimated as follows (in hundred of rupees)

| | Job | | | | | |
|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** |
| | **1** | 2.5 | 5 | 1 | 6 | 1 |
| | **2** | 2 | 5 | 1.5 | 7 | 3 |
| | **3** | 3 | 6.5 | 2 | 8 | 3 |
| | **4** | 3.5 | 7 | 2 | 9 | 4.5 |
| | **5** | 4 | 7 | 3 | 9 | 6 |
| | **6** | 6 | 9 | 5 | 10 | 6 |

Solve the problem assuring that the objective is to minimise total cost.

# SOLUTION

Since given Assignment problem is not Balanced. So we a add dummy job 6 with 0 assignment cost to make it balanced. Thus given cost matrix become

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 2.5 | 5 | 1 | 6 | 1 | 0 |
| | **2** | 2 | 5 | 1.5 | 7 | 3 | 0 |
| | **3** | 3 | 6.5 | 2 | 8 | 3 | 0 |
| | **4** | 3.5 | 7 | 2 | 9 | 4.5 | 0 |
| | **5** | 4 | 7 | 3 | 9 | 6 | 0 |
| | **6** | 6 | 9 | 5 | 10 | 6 | 0 |

Subtract smallest element of each row from the each element of that row and then in reduced matrix subtract smallest element of each column from each element of that column.

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 2.5 | 5 | 1 | 6 | 1 | 0 |
| | **2** | 2 | 5 | 1.5 | 7 | 3 | 0 |
| | **3** | 3 | 6.5 | 2 | 8 | 3 | 0 |
| | **4** | 3.5 | 7 | 2 | 9 | 4.5 | 0 |
| | **5** | 4 | 7 | 3 | 9 | 6 | 0 |
| | **6** | 6 | 9 | 5 | 10 | 6 | 0 |

**Row Reduction**


| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 2.5 | 5 | 1 | 6 | 1 | 0 |
| | **2** | 2 | 5 | 1.5 | 7 | 3 | 0 |
| | **3** | 3 | 6.5 | 2 | 8 | 3 | 0 |
| | **4** | 3.5 | 7 | 2 | 9 | 4.5 | 0 |
| | **5** | 4 | 7 | 3 | 9 | 6 | 0 |
| | **6** | 6 | 9 | 5 | 10 | 6 | 0 |

| Machine | Job | | | | | |
|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | 2.5 | 5 | 1 | 6 | 1 | 0 |
| **2** | 2 | 5 | 1.5 | 7 | 3 | 0 |
| **3** | 3 | 6.5 | 2 | 8 | 3 | 0 |
| **4** | 3.5 | 7 | 2 | 9 | 4.5 | 0 |
| **5** | 4 | 7 | 3 | 9 | 6 | 0 |
| **6** | 6 | 9 | 5 | 10 | 6 | 0 |

**Column Reduction**

| Machine | Job | | | | | |
|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | 0.5 | 0 | 0 | 0 | 0 | 0 |
| **2** | 0 | 0 | 0.5 | 1 | 2 | 0 |
| **3** | 1 | 1.5 | 1 | 2 | 2 | 0 |
| **4** | 1.5 | 2 | 1 | 3 | 3.5 | 0 |
| **5** | 2 | 2 | 2 | 3 | 5 | 0 |
| **6** | 4 | 4 | 4 | 4 | 5 | 0 |

# SOLUTION

| | | Job | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 0.5 | [0] | ~~0~~ | ~~0~~ | ~~0~~ | ~~0~~ |
| | **2** | [0] | ~~0~~ | 0.5 | 1 | 2 | ~~0~~ |
| | **3** | 1 | 1.5 | 1 | 2 | 2 | ~~0~~ |
| | **4** | 1.5 | 2 | 1 | 3 | 3.5 | [0] |
| | **5** | 2 | 2 | 2 | 3 | 5 | ~~0~~ |
| | **6** | 4 | 4 | 4 | 4 | 5 | ~~0~~ |

Since there is no assigned zero in third, fifth and sixth column. So this matrix will not give the optimal solution. So have to proceed further steps.

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 0.5 | 0 | 0 | 0 | 0 | 0 |
| | **2** | 0 | 0 | 0.5 | 1 | 2 | 0 |
| | **3** | 1 | 1.5 | 1 | 2 | 2 | 0 ✓ |
| | **4** | 1.5 | 2 | 1 | 3 | 3.5 | 0 ✓ |
| | **5** | 2 | 2 | 2 | 3 | 5 | 0 ✓ |
| | **6** | 4 | 4 | 4 | 4 | 5 | 0 ✓ |

✓

Here 1 is the smallest element from uncovered elements. Subtract 1 from all uncovered element and add 1 to the element lying in the intersection of the vertical and horizontal line ie numbers covered by two lines.

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 0.5 | 0 | 0 | 0 | 0 | 1 |
| | **2** | 0 | 0 | 0.5 | 1 | 2 | 1 |
| | **3** | 0 | 0.5 | 0 | 1 | 1 | 0 |
| | **4** | 0.5 | 1 | 0 | 2 | 2.5 | 0 |
| | **5** | 1 | 1 | 1 | 2 | 4 | 0 |
| | **6** | 3 | 3 | 3 | 3 | 4 | 0 |

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | 1 | 2 | 3 | 4 | 5 | 6 |
| | 1 | 0.5 | 0 | 0 | 0 | 0 | 1 |
| | 2 | 0 | 0 | 0.5 | 1 | 2 | 1 |
| | 3 | 0 | 0.5 | 0 | 1 | 1 | 0 |
| | 4 | 0.5 | 1 | 0 | 2 | 2.5 | 0 |
| | 5 | 1 | 1 | 1 | 2 | 4 | 0 ✓ |
| | 6 | 3 | 3 | 3 | 3 | 4 | 0 ✓ |

✓

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 0.5 | ~~0~~ | ~~0~~ | [0] | ~~0~~ | 2 |
| | **2** | ~~0~~ | ~~0~~ | 0.5 | 1 | 2 | 2 |
| | **3** | [0] | 0.5 | ~~0~~ | 1 | 1 | 1 |
| | **4** | 0.5 | 1 | [0] | 2 | 2.5 | 1 |
| | **5** | ~~0~~ | [0] | ~~0~~ | 1 | 3 | ~~0~~ |
| | **6** | 2 | 2 | 2 | 2 | 3 | [0] |

Since there is no assigned zero in second column. So this matrix will not give the optimal solution. So have to proceed further steps.

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 0.5 | 0 | 0 | 0 | 0 | 2 |
| | **2** | 0 | 0 | 0.5 | 1 | 2 | 2 |
| | **3** | 0 | 0.5 | 0 | 1 | 1 | 1 |
| | **4** | 0.5 | 1 | 0 | 2 | 2.5 | 1 |
| | **5** | 0 | 0 | 0 | 1 | 3 | 0 |
| | **6** | 2 | 2 | 2 | 2 | 3 | 0 |

Here 1 is the smallest element from uncovered elements. Subtract 1 from all uncovered element and add 1 to the element lying in the intersection of the vertical and horizontal line ie numbers covered by two lines.

# FINAL SOLUTION MATRIX

Giving zero assignment in the usual manner we get the optimal assignment given by the following table

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 1.5 | 1 | 1 | [0] | ~~0~~ | 2 |
| | **2** | [0] | ~~0~~ | 0.5 | ~~0~~ | 1 | 2 |
| | **3** | ~~0~~ | 0.5 | ~~0~~ | ~~0~~ | [0] | 1 |
| | **4** | 0.5 | 1 | [0] | 1 | 1.5 | 1 |
| | **5** | ~~0~~ | [0] | ~~0~~ | ~~0~~ | 2 | ~~0~~ |
| | **6** | 2 | 2 | 2 | 1 | 2 | [0] |

**$1 \rightarrow 4$, $2 \rightarrow 1$, $3 \rightarrow 5$, $4 \rightarrow 3$, $5 \rightarrow 2$**

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 1.5 | 1 | 1 | [0] | ~~0~~ | 2 |
| | **2** | ~~0~~ | [0] | 0.5 | ~~0~~ | 1 | 2 |
| | **3** | ~~0~~ | 0.5 | ~~0~~ | ~~0~~ | [0] | 1 |
| | **4** | 0.5 | 1 | [0] | 1 | 1.5 | 1 |
| | **5** | [0] | ~~0~~ | ~~0~~ | ~~0~~ | 2 | ~~0~~ |
| | **6** | 2 | 2 | 2 | 1 | 2 | [0] |

$1 \rightarrow 4, \quad 2 \rightarrow 2, \quad 3 \rightarrow 5, \quad 4 \rightarrow 3, \quad 5 \rightarrow 1$

# SOLUTION

| | Job | | | | | |
|---|---|---|---|---|---|---|
| Machine | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | 1.5 | 1 | 1 | ~~0~~ | [0] | 2 |
| **2** | [0] | ~~0~~ | 0.5 | ~~0~~ | 1 | 2 |
| **3** | ~~0~~ | 0.5 | ~~0~~ | [0] | ~~0~~ | 1 |
| **4** | 0.5 | 1 | [0] | 1 | 1.5 | 1 |
| **5** | ~~0~~ | [0] | ~~0~~ | ~~0~~ | 2 | ~~0~~ |
| **6** | 2 | 2 | 2 | 1 | 2 | [0] |

**$1 \rightarrow 5$, $2 \rightarrow 1$, $3 \rightarrow 4$, $4 \rightarrow 3$, $5 \rightarrow 2$**

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 1.5 | 1 | 1 | ~~0~~ | 0 | 2 |
| | **2** | ~~0~~ | 0 | 0.5 | ~~0~~ | 1 | 2 |
| | **3** | 0 | 0.5 | ~~0~~ | ~~0~~ | ~~0~~ | 1 |
| | **4** | 0.5 | 1 | 0 | 1 | 1.5 | 1 |
| | **5** | ~~0~~ | ~~0~~ | ~~0~~ | 0 | 2 | ~~0~~ |
| | **6** | 2 | 2 | 2 | 1 | 2 | 0 |

$$1 \rightarrow 5, \quad 2 \rightarrow 2, \quad 3 \rightarrow 1, \quad 4 \rightarrow 3, \quad 5 \rightarrow 4$$

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 1.5 | 1 | 1 | ~~0~~ | [0] | 2 |
| | **2** | ~~0~~ | [0] | 0.5 | ~~0~~ | 1 | 2 |
| | **3** | ~~0~~ | 0.5 | ~~0~~ | [0] | ~~0~~ | 1 |
| | **4** | 0.5 | 1 | [0] | 1 | 1.5 | 1 |
| | **5** | [0] | ~~0~~ | ~~0~~ | ~~0~~ | 2 | ~~0~~ |
| | **6** | 2 | 2 | 2 | 1 | 2 | [0] |

$$1 \rightarrow 5, \quad 2 \rightarrow 2, \quad 3 \rightarrow 4, \quad 4 \rightarrow 3, \quad 5 \rightarrow 1$$

# SOLUTION

| | Job | | | | | | |
|---|---|---|---|---|---|---|---|
| Machine | | **1** | **2** | **3** | **4** | **5** | **6** |
| | **1** | 1.5 | 1 | 1 | ~~0~~ | [0] | 2 |
| | **2** | ~~0~~ | 0 | 0.5 | [0] | 1 | 2 |
| | **3** | [0] | 0.5 | ~~0~~ | ~~0~~ | ~~0~~ | 1 |
| | **4** | 0.5 | 1 | [0] | 1 | 1.5 | 1 |
| | **5** | ~~0~~ | [0] | ~~0~~ | ~~0~~ | 2 | ~~0~~ |
| | **6** | 2 | 2 | 2 | 1 | 2 | [0] |

$$1 \rightarrow 5, \quad 2 \rightarrow 4, \quad 3 \rightarrow 1, \quad 4 \rightarrow 3, \quad 5 \rightarrow 2$$

# SOLUTION

Since each row and column of above given all 6 solution matrices have assigned zero. All above written 6 matrices will give the optimal solution (Alternative Solution). Hence optimal solution are

$$1 \rightarrow 4, \quad 2 \rightarrow 1, \quad 3 \rightarrow 5, \quad 4 \rightarrow 3, \quad 5 \rightarrow 2$$
$$1 \rightarrow 4, \quad 2 \rightarrow 2, \quad 3 \rightarrow 5, \quad 4 \rightarrow 3, \quad 5 \rightarrow 1$$
$$1 \rightarrow 5, \quad 2 \rightarrow 1, \quad 3 \rightarrow 4, \quad 4 \rightarrow 3, \quad 5 \rightarrow 2$$
$$1 \rightarrow 5, \quad 2 \rightarrow 2, \quad 3 \rightarrow 1, \quad 4 \rightarrow 3, \quad 5 \rightarrow 4$$
$$1 \rightarrow 5, \quad 2 \rightarrow 2, \quad 3 \rightarrow 4, \quad 4 \rightarrow 3, \quad 5 \rightarrow 1$$
$$1 \rightarrow 5, \quad 2 \rightarrow 4, \quad 3 \rightarrow 1, \quad 4 \rightarrow 3, \quad 5 \rightarrow 2$$

In all the optimal solution $6^{th}$ machine will not do any job and minimum total cost in all cases $= (6 + 2 + 3 + 2 + 7) \times 100 = 2000.$

# THANK YOU