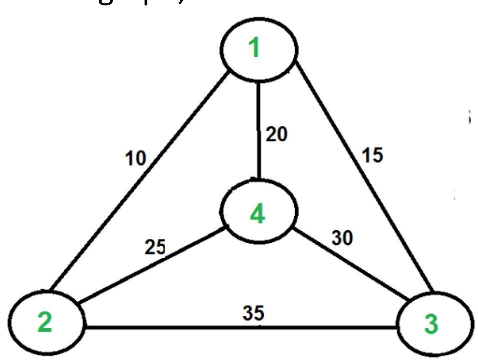
Branch and Bound – What is it?

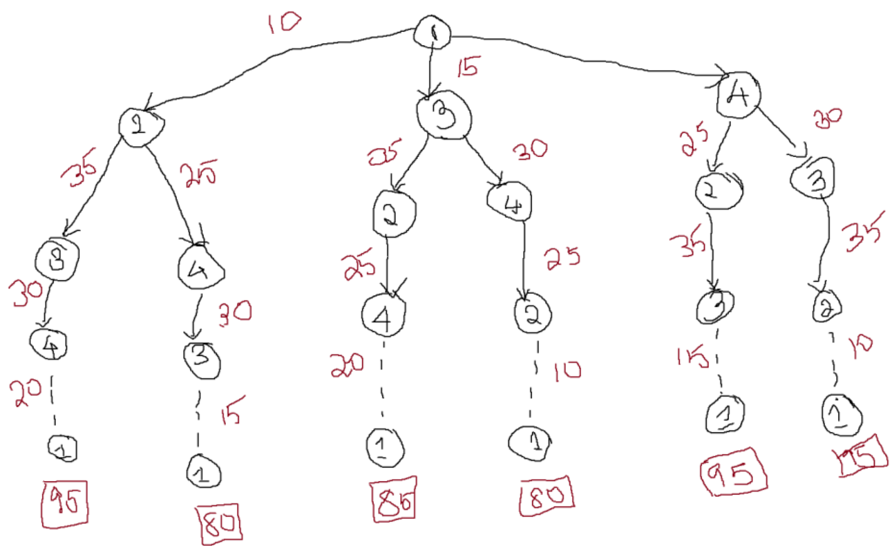
- The branch and bound technique is used to solve optimization problems
- Branch and bound uses the state space tree for solving the combinatorial optimization problem
- It has two major steps
  - **Branching** is technique involves division of the main problem into two or more subproblems.
  - **Bounding** operation restricts the state space tree to grow exponentially.
- Branch and bound build the state space tree and find the optimal solution quickly by pruning (removing) a few of the branches of the tree that do not satisfy the bound.
- Branch and bound can be useful where some other optimization techniques like greedy or dynamic programming fail. Such algorithms are typically slower than their counterparts. In the worst case, it may run in exponential time, but careful selection of bounds and branches makes an algorithm run reasonably faster.

State Space tree – What is it?

- A space state tree is a tree representing all the possible states (solution or nonsolution) of the problem from the root as an initial state to the leaf as a terminal state.
- Example: Consider the below graph,



- Consider you have to find out the path with minimum cost in which
  - If I start at vertex 1, your path should end on the vertex
  - Here, you should only visit each vertex exactly once
- For this, you should know all the possible paths and then you can find the path with minimum cost.
- State Space Tree for this problem looks like



- Possible Paths are (with their cost)
  - 1 – 2 – 3 – 4 – 1 = 95
  - 1 – 2 – 4 – 3 – 1 = 80
  - 1 – 3 – 2 – 4 – 1 = 85
  - 1 – 3 – 4 – 2 – 1 = 80
  - 1 – 4 – 2 – 3 – 1 = 95
  - 1 – 4 – 3 – 2 – 1 = 95
- In these possible paths, the paths with the minimum cost are:
  - 1 – 2 – 4 – 3 – 1 = 80
  - 1 – 3 – 4 – 2 – 1 = 80

Types of Branch and Bound

There are multiple types of the Branch and Bound method, based on the order in which the state space tree is to be searched.

- **FIFO Branch and Bound**
  - The First-In-First-Out approach to the branch and bound problem follows the queue approach in creating the state-space tree.
  - Here, breadth first search is performed, i.e., the elements at a particular level are all searched, and then the elements of the next level are searched, starting from the first child of the first node in the previous level.
- **LIFO Branch and Bound**
  - The Last-In-First-Out approach to this problem follows the stack approach in creating the state space tree.

- Here, when nodes get added to the state space tree, think of them as getting added to a stack. When all nodes of a level are added, we pop the topmost element from the stack and then explore it.

- Least Cost Branch and Bound
  - We give the preference to an answer node with minimum cost or least cost.
  - In this method, after the children of a particular node have been explored, the next node to be explored would be that node out of the unexplored nodes which has the least cost.

Travelling Salesman Problem – What is it?

- Travelling Salesman Problem (TSP) is an interesting problem.
- Problem is defined as **given n cities and distance between each pair of cities, find out the path which visits each city exactly once and come back to starting city, with the constraint of minimizing the travelling distance**.
- TSP has many practical applications. It is used in network design, and transportation route design. The objective is to minimize the distance. We can start tour from any random city and visit other cities in any order.
- With n cities, n! different permutations are possible. Exploring all paths using brute force attacks may not be useful in real life applications.
- **Least Common Branch and bound** is an effective way to solve TSP.

It works as follows

Consider directed weighted graph G = (V, E, W), where node represents cities and weighted directed edges represents direction and distance between two cities.

1. Initially, graph is represented by cost matrix C, where

   Cij = cost of edge, if there is a direct path from city i to city j

   Cij = ∞, if there is no direct path from city i to city j.
2. Convert cost matrix to reduced matrix by subtracting minimum values from appropriate rows and columns, such that each row and column contains at least one zero entry.
3. Find cost of reduced matrix. Cost is given by summation of subtracted amount from the cost matrix to convert it in to reduce matrix.
4. Prepare state space tree for the reduce matrix

5. Find least cost valued node A (i.e., E-node), by computing reduced cost node matrix with every remaining node.
6. If <i, j> edge is to be included, then do following:
   (a) Set all values in row i and all values in column j of A to ∞
   (b) Set A [j, 1] = ∞
   (c) Reduce A again, except rows and columns having all ∞ entries.
7. Compute the cost of newly created reduced matrix as,

   $$\text{Cost} = L + \text{Cost}(i, j) + r$$

   Where, L is cost of original reduced cost matrix and r is A [i, j].
8. If all nodes are not visited, then go to step 4.

Reduction

- Matrix M is called **reduced matrix** if each of its row and column has at least one zero entry or entire row or entire column has ∞ value.
- We will be doing row and column reduction.
- Matrix is called **row reduced** if each of its row has at least one zero entry or all ∞ entries.
- Matrix is called **column reduced** if each of its column has at least one zero entry or all ∞ entries.

Let M represents the distance matrix of 5 cities. Consider the following distance matrix:

M =

∞	20	30	10	11
15	∞	16	4	2
3	5	∞	2	4
19	6	18	∞	3
16	4	7	16	∞

Row Reduction

- M can be row reduced as follow:

$$M_{RowRed} = \{M_{ij} - \min \{M_{ij} \mid 1 \le j \le n, \text{ and } M_{ij} < \infty \}\}$$

- Find the minimum element from each row and subtract it from each cell of matrix.

$$M = \begin{array}{|c|c|c|c|c|}\hline \infty & 20 & 30 & 10 & 11 \\ \hline 15 & \infty & 16 & 4 & 2 \\ \hline 3 & 5 & \infty & 2 & 4 \\ \hline 19 & 6 & 18 & \infty & 3 \\ \hline 16 & 4 & 7 & 16 & \infty \\ \hline \end{array} \begin{array}{l} \rightarrow 10 \\ \rightarrow 2 \\ \rightarrow 2 \\ \rightarrow 3 \\ \rightarrow 4 \end{array}$$

- Reduced matrix would be:

$$M_{RowRed} = \begin{array}{|c|c|c|c|c|}\hline \infty & 10 & 20 & 0 & 1 \\ \hline 13 & \infty & 14 & 2 & 0 \\ \hline 1 & 3 & \infty & 0 & 2 \\ \hline 16 & 3 & 15 & \infty & 0 \\ \hline 12 & 0 & 3 & 12 & \infty \\ \hline \end{array}$$

- Row reduction cost is the summation of all the values subtracted from each row:

Row Reduction Cost (M) = 10 + 2 + 2 + 3 + 4 = 21

Column Reduction

- Matrix M_{RowRed} is row reduced but not the column reduced. M can be column reduced as follow:

$$\mathbf{M_{ColRed} = \{M_{ji} - \min \{M_{ji} \mid 1 \le j \le n, \text{ and } M_{ji} < \infty \}\}}$$

- To reduce above matrix, we will find the minimum element from each column and subtract it from each cell of matrix.

$$M_{RowRed} = \begin{array}{|c|c|c|c|c|}\hline \infty & 10 & 20 & 0 & 1 \\ \hline 13 & \infty & 14 & 2 & 0 \\ \hline 1 & 3 & \infty & 0 & 2 \\ \hline 16 & 3 & 15 & \infty & 0 \\ \hline 12 & 0 & 3 & 12 & \infty \\ \hline \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 0 & 3 & 0 & 0 \end{array}$$

- Column reduced matrix M_{ColRed} would be:

$$M_{ColRed} = \begin{array}{|c|c|c|c|c|} \hline \infty & 10 & 17 & 0 & 1 \\ \hline 12 & \infty & 11 & 2 & 0 \\ \hline 0 & 3 & \infty & 0 & 2 \\ \hline 15 & 3 & 12 & \infty & 0 \\ \hline 11 & 0 & 0 & 12 & \infty \\ \hline \end{array}$$

- Column reduction cost is the summation of all the values subtracted from each column:

Column Reduction Cost (M) = 1 + 0 + 3 + 0 + 0 = 4

Each row and column of M_{ColRed} has at least one zero entry, so this matrix is reduced matrix. Now, the reduction cost of the Matrix M is summation row reduction cost and column reduction cost.

Reduction Cost(M) = Row Reduction Cost(M) + Column Reduction Cost (M)= 21+ 4 = 25

Example: Find the solution of following travelling salesman problem using branch and bound method.

$$M = \begin{array}{|c|c|c|c|c|} \hline \infty & 20 & 30 & 10 & 11 \\ \hline 15 & \infty & 16 & 4 & 2 \\ \hline 3 & 5 & \infty & 2 & 4 \\ \hline 19 & 6 & 18 & \infty & 3 \\ \hline 16 & 4 & 7 & 16 & \infty \\ \hline \end{array}$$

Steps:

1. Draw state space tree with optimal reduction cost at root node.
2. Derive cost of path from node i to j by setting all entries in ith row and jth column as ∞. Also, Set M[j][i] = ∞
3. Cost of corresponding node N for path i to j is summation of optimal cost + reduction cost + M[j][i]

4. After exploring all nodes at level i, set node with minimum cost as E node and repeat the procedure until all nodes are visited.
5. Given matrix is not reduced. In order to find reduced matrix of it, we will first find the row reduced matrix followed by column reduced matrix if needed. We can find row reduced matrix by subtracting minimum element of each row from each element of corresponding row.

Solution

Let’s draw the state space tree from the Reduced Matrix

M is not reduced matrix. Reduce it subtracting minimum value from corresponding row. Doing this we get,

$$\begin{bmatrix} \infty & 20 & 30 & 10 & 11 \\ 15 & \infty & 16 & 4 & 2 \\ 3 & 5 & \infty & 2 & 4 \\ 19 & 6 & 18 & \infty & 3 \\ 16 & 4 & 7 & 16 & \infty \end{bmatrix} \begin{matrix} \rightarrow 10 \\ \rightarrow 2 \\ \rightarrow 2 \\ \rightarrow 3 \\ \rightarrow 4 \end{matrix} \Rightarrow \begin{array}{c} \begin{bmatrix} \infty & 10 & 20 & 0 & 1 \\ 13 & \infty & 14 & 2 & 0 \\ 1 & 3 & \infty & 0 & 2 \\ 16 & 3 & 15 & \infty & 0 \\ 12 & 0 & 3 & 12 & \infty \end{bmatrix} \\ \begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 0 & 3 & 0 & 0 \end{matrix} \end{array} = M_1'$$

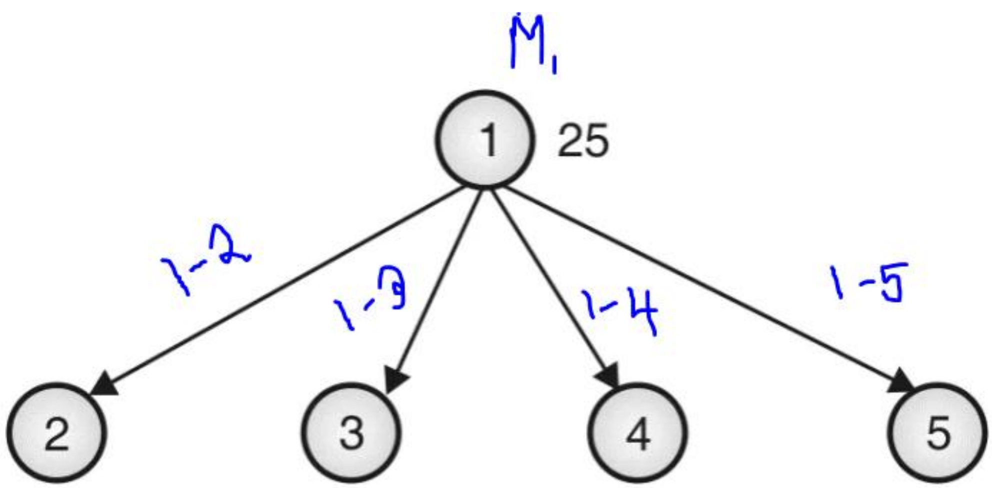
M_1' is not reduced matrix. Reduce it subtracting minimum value from corresponding column. Doing this we get,

$$\begin{bmatrix} \infty & 10 & 17 & 0 & 1 \\ 12 & \infty & 11 & 2 & 0 \\ 0 & 3 & \infty & 0 & 2 \\ 15 & 3 & 12 & \infty & 0 \\ 11 & 0 & 0 & 12 & \infty \end{bmatrix} = M_1$$

Cost of M1 = C (1) = Row reduction cost + Column reduction cost

= (10 + 2 + 2 + 3 + 4) + (1 + 3) = 25

This means all tours in graph has length at least **25**. This is the optimal cost of the path.



Let us find cost of edge from node 1 to 2, 3, 4, 5.

Select edge 1-2:

- Set M1 [1] [ ] = M1 [ ] [2] = ∞
- Set M1 [2] [1] = ∞

Reduce the resultant matrix if required.

$$\begin{array}{|c|c|c|c|c|l}
\hline
\infty & \infty & \infty & \infty & \infty & \rightarrow x \\
\hline
\infty & \infty & 11 & 2 & 0 & \rightarrow 0 \\
\hline
0 & \infty & \infty & 0 & 2 & \rightarrow 0 \quad = M_2 \\
\hline
15 & \infty & 12 & \infty & 0 & \rightarrow 0 \\
\hline
11 & \infty & 0 & 12 & \infty & \rightarrow 0 \\
\hline
\multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \\
\multicolumn{1}{c}{0} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{0} & \multicolumn{1}{c}{0} & \multicolumn{1}{c}{0} &
\end{array}$$

M2 is already reduced.

Cost of node 2 = C (2) = C (1) + Reduction cost + M1 [1] [2] = 25 + 0 + 10 = 35

Select edge 1-3

- Set M1 [1][ ] = M1 [ ] [3] = ∞
- Set M1 [3][1] = ∞

Reduce the resultant matrix if required.

$$
M_1 \Rightarrow \begin{array}{|c|c|c|c|c|l}
\hline
\infty & \infty & \infty & \infty & \infty & \rightarrow x \\
\hline
12 & \infty & 11 & 2 & 0 & \rightarrow 0 \\
\hline
0 & \infty & \infty & 0 & 2 & \rightarrow 0 \\
\hline
15 & \infty & 12 & \infty & 0 & \rightarrow 0 \\
\hline
11 & \infty & 0 & 12 & \infty & \rightarrow 0 \\
\hline
\multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \\
\multicolumn{1}{c}{11} & \multicolumn{1}{c}{0} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{0} & \multicolumn{1}{c}{0} &
\end{array}
\Rightarrow
\begin{array}{|c|c|c|c|c|}
\hline
\infty & \infty & \infty & \infty & \infty \\
\hline
1 & \infty & \infty & 2 & 0 \\
\hline
\infty & 3 & \infty & 0 & 2 \\
\hline
4 & 3 & \infty & \infty & 0 \\
\hline
0 & 0 & \infty & 12 & \infty \\
\hline
\end{array}
= M_3
$$

Cost of node 3 = C (3) = C (1) + Reduction cost + M1[1] [3] = 25 + 11 + 17 = 53

Select edge 1-4:

- Set M1 [1][ ] = M1[ ][4] = ∞
- Set M1 [4][1] = ∞

Reduce resultant matrix if required.

$$
M_1 \Rightarrow \begin{array}{|c|c|c|c|c|l}
\hline
\infty & \infty & \infty & \infty & \infty & \rightarrow x \\
\hline
12 & \infty & 11 & \infty & 0 & \rightarrow 0 \\
\hline
0 & 3 & \infty & \infty & 2 & \rightarrow 0 \\
\hline
\infty & 3 & 12 & \infty & 0 & \rightarrow 0 \\
\hline
11 & 0 & 0 & \infty & \infty & \rightarrow 0 \\
\hline
\multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{\downarrow} & \\
\multicolumn{1}{c}{0} & \multicolumn{1}{c}{0} & \multicolumn{1}{c}{0} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{0} &
\end{array}
= M_4
$$

Matrix M4 is already reduced.

Cost of node 4 = C(4) = C(1) + Reduction cost + M1 [1] [4] = 25 + 0 + 0 = 25

Select edge 1-5:

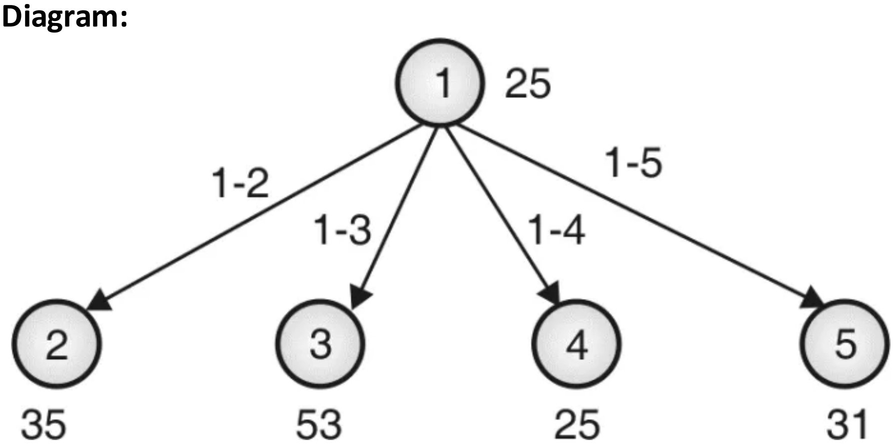
- Set M1 [1] [ ] = M1 [ ] [5] = ∞
- Set M1 [5] [1] = ∞

Reduce the resultant matrix if required.

$$
M_1 \Rightarrow \begin{array}{|c|c|c|c|c|}\hline \infty & \infty & \infty & \infty & \infty \\ \hline 12 & \infty & 11 & 2 & \infty \\ \hline 0 & 3 & \infty & 0 & \infty \\ \hline 15 & 3 & 12 & \infty & \infty \\ \hline \infty & 0 & 0 & 12 & \infty \\ \hline \end{array}
\begin{array}{c} \rightarrow x \\ \rightarrow 2 \\ \rightarrow 0 \\ \rightarrow 3 \\ \rightarrow 0 \end{array}
\Rightarrow
\begin{array}{c} \begin{array}{|c|c|c|c|c|}\hline \infty & \infty & \infty & \infty & \infty \\ \hline 10 & \infty & 9 & 0 & \infty \\ \hline 0 & 3 & \infty & 0 & \infty \\ \hline 12 & 0 & 9 & \infty & \infty \\ \hline \infty & 0 & 0 & 12 & \infty \\ \hline \end{array} \\ \begin{array}{ccccc} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & 0 & 0 & 0 & x \end{array} \end{array}
= M_5
$$

Cost of node 5 = C(5) = C(1) + reduction cost + M1 [1] [5]   = 25 + 5 + 1 = 31

State Space Diagram:



Node 4 has minimum cost for path 1-4. We can go to vertex 2, 3 or 5. Let's explore all three nodes.

Select path 1-4-2 : (Add edge 4-2)

- Set M4 [1] []   =   M4 [4] [] = M4 [] [2] = ∞
- Set M4 [2] [1]   =   ∞

Reduce resultant matrix if required.

$$M_4 \Rightarrow \begin{array}{c} \begin{array}{|c|c|c|c|c|} \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & 11 & \infty & 0 \\ \hline 0 & \infty & \infty & \infty & 2 \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 11 & \infty & 0 & \infty & \infty \\ \hline \end{array} \begin{array}{l} \rightarrow x \\ \rightarrow 0 \\ \rightarrow 0 \\ \rightarrow x \\ \rightarrow 0 \end{array} \\ \begin{array}{ccccc} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & 0 & 0 & x & 0 \end{array} \end{array} = M_6$$

Matrix M_6 is already reduced.

Cost of node 6 = C(6) = C(4) + Reduction cost + M4 [4] [2] = 25 + 0 + 3 = 28

Select edge 4-3 (Path 1-4-3):

- Set M4 [1] [ ] = M4 [4] [ ] = M4 [ ] [3] = ∞
- Set M [3][1] = ∞

Reduce the resultant matrix if required.

$$M_4 \Rightarrow \begin{array}{|c|c|c|c|c|} \hline \infty & \infty & \infty & \infty & \infty \\ \hline 12 & \infty & \infty & \infty & 0 \\ \hline \infty & 3 & \infty & \infty & 2 \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 11 & 0 & \infty & \infty & \infty \\ \hline \end{array} \begin{array}{l} \rightarrow x \\ \rightarrow 0 \\ \rightarrow 2 \\ \rightarrow \infty \\ \rightarrow 0 \end{array} \Rightarrow \begin{array}{c} \begin{array}{|c|c|c|c|c|} \hline \infty & \infty & \infty & \infty & \infty \\ \hline 12 & \infty & \infty & \infty & 0 \\ \hline \infty & 1 & \infty & \infty & 0 \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 11 & 0 & \infty & \infty & \infty \\ \hline \end{array} \\ \begin{array}{ccccc} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 11 & 0 & x & x & 0 \end{array} \end{array} = M_7'$$

$M_7^{`}$ is not reduced. Reduce it by subtracting 11 from column 1.

$$\therefore M_7' \Rightarrow \begin{array}{|c|c|c|c|c|} \hline \infty & \infty & \infty & \infty & \infty \\ \hline 1 & \infty & \infty & \infty & 0 \\ \hline \infty & 1 & \infty & \infty & 2 \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 0 & 0 & \infty & \infty & \infty \\ \hline \end{array} = M_7$$

Matrix M_7 is reduced

Cost of node 7 = C(7) = C(4) + Reduction cost + M4 [4] [3] = 25 + 2 + 11 + 12 = 50

Select edge 4-5 (Path 1-4-5):

- Set M4 [1] [ ] = M4 [4] [ ] = M4 [ ] [5] = ∞
- Set M [5][1] = ∞

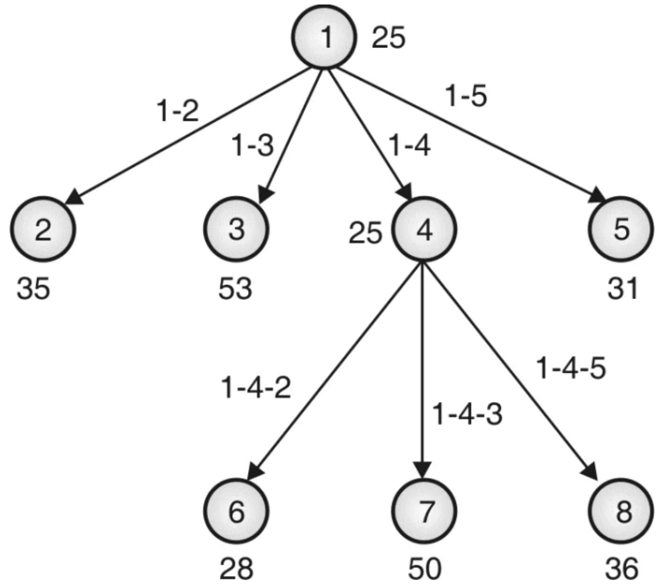
The resultant matrix is not reduced. Reduce it by subtracting 11 from row 2.

$$
M_4 \Rightarrow \begin{array}{|c|c|c|c|c|} \hline \infty & \infty & \infty & \infty & \infty \\ \hline 12 & \infty & 11 & \infty & \infty \\ \hline 0 & 3 & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & 0 & 0 & \infty & \infty \\ \hline \end{array}
\begin{array}{l} \rightarrow x \\ \rightarrow 11 \\ \rightarrow 0 \\ \rightarrow x \\ \rightarrow 0 \end{array}
\Rightarrow
\begin{array}{c} \begin{array}{|c|c|c|c|c|} \hline \infty & \infty & \infty & \infty & \infty \\ \hline 1 & \infty & 0 & \infty & \infty \\ \hline 0 & 3 & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & 0 & 0 & \infty & \infty \\ \hline \end{array} \\ \begin{array}{ccccc} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & 0 & 0 & x & x \end{array} \end{array}
= M_8
$$

Matrix M_8 is reduced.

Cost of node 8 = C (8) = C(4) + Reduction cost + M4 [4][5] = 25 + 11 + 0 = 36

State space tree



Path 1-4-2 leads to minimum cost. Let’s find the cost for two possible paths.

Add edge 2-3 (Path 1-4-2-3):

- Set M6 [1][ ] = M6 [4][ ] = M6 [2][ ]= M6 [][3] = ∞
- Set M6 [3][1] = ∞

Reduce the resultant matrix if required.

$$M_6 \Rightarrow \begin{array}{|c|c|c|c|c|}\hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 0 & \infty & \infty & \infty & 2 \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 11 & \infty & \infty & \infty & \infty \\ \hline \end{array} \begin{array}{l} \rightarrow x \\ \rightarrow x \\ \rightarrow 0 \\ \rightarrow x \\ \rightarrow 11 \end{array} \Rightarrow \begin{array}{c} \begin{array}{|c|c|c|c|c|}\hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 0 & \infty & \infty & \infty & 2 \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 0 & \infty & \infty & \infty & \infty \\ \hline \end{array} \\ \begin{array}{ccccc} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & x & x & x & 2 \end{array} \end{array} = M_9'$$

$M_9^{`}$ is not reduced. Reduce it by subtracting 11 from row 5 and then reducing 2 from the column 5.

$$\therefore M_9' \Rightarrow \begin{array}{|c|c|c|c|c|} \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 0 & \infty & \infty & \infty & 0 \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 0 & \infty & \infty & \infty & \infty \\ \hline \end{array} = M_9$$

Matrix M_9 is reduced

Cost of node 9 = C(9) = C(6) + Reduction cost + M6 [2][3] = 28 + 11 + 2 + 11 = 52

Add edge 2-5 (Path 1-4-2-5):

- Set M6 [1][ ] = M6 [4][ ] = M6 [2][ ] = M6 [ ][5] = ∞
- Set M6 [5][1] = ∞

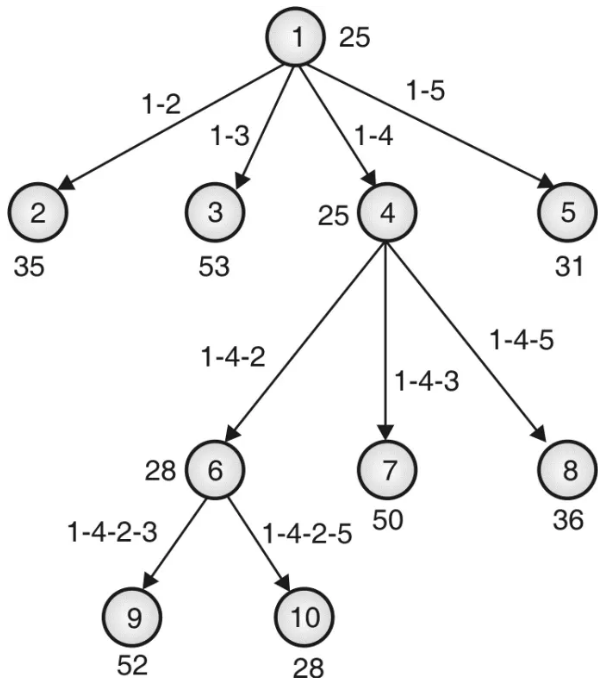
Reduce resultant matrix if required.

$$\therefore M_6 \Rightarrow \begin{array}{|c|c|c|c|c|} \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline 0 & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & 0 & \infty & \infty \\ \hline \end{array} = M_{10}$$

Matrix M_{10} is reduced

Cost of node 10 = C(10) = C(6) + Reduction cost + M6 [2][5] = 28 + 0 + 0 = 28

State space tree



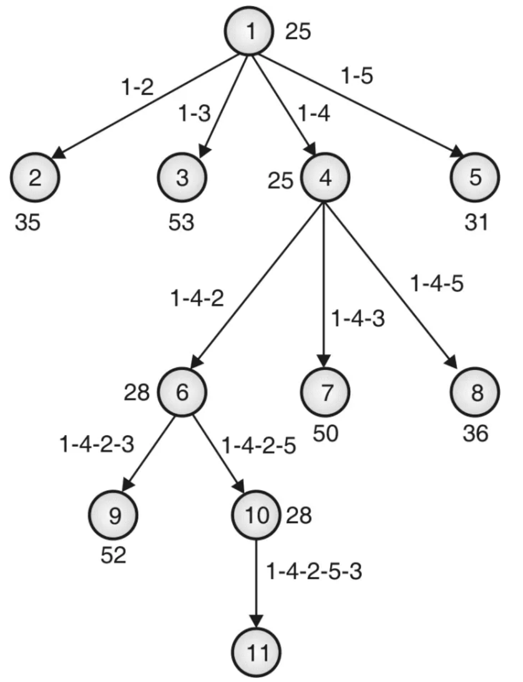
Path 1-4-2-5 leads to minimum cost. Let's find the cost by adding edge 5-3.

Add edge 5-3 (Path 1-4-2-5-3):

$$\therefore\ M_{10} \Rightarrow \begin{array}{|c|c|c|c|c|}\hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline \infty & \infty & \infty & \infty & \infty \\ \hline \end{array} = M_{11}$$

Cost of node 11 = C(11) = C(10) + Reduction cost + M10 [5][3] = 28 + 0 + 0 = 28

State space tree:



Thus, the final path is 1 – 4 – 2 – 5 – 3 – 1 which has cost of 28.