Faculty Name: Bhawana Pillai,
Department: CSE,
Designation: A.P.
Subject: Analysis & Design of Algorithms,
Unit: IV; Topic: Branch & Bound Technique

15 Puzzel Problem by Branch and Bound (Least Cost Search)

The 15 Puzzel problem is invented by Sam Loyd in 1878.

The problem cinsist of 15 numbered (0-15) tiles on a square box with 16 tiles(one tile is blank or empty).

The objective of this problem is to change the arrangement of initial node to goal node by using series of legal moves.

The Initial and Goal node arrangement is shown by following figure.

1	2	4	15
2		5	12
7	6	11	14
8	9	10	13

Initial Arrangement

→

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Final Arrangement

In initial node four moves are possible. User can move any one of the tile like 2,or 3, or 5, or 6 to the empty tile. From this we have four possibilities to move from initial node.

The legal moves are for adjacent tile number is left, right, up, down, ones at a time.

Each and every move creates a new arrangement, and this arrangement is called state of puzzle problem.

By using different states, a state space tree diagram is created, in which edges are labeled according to the direction in which the empty space moves.

The state space tree is very large because it can be 16! Different arrangements.

In state space tree, nodes are numbered as per the level. In each level we must calculate the value or cost of each node by using given formula:

C(x)=f(x)+g(x),

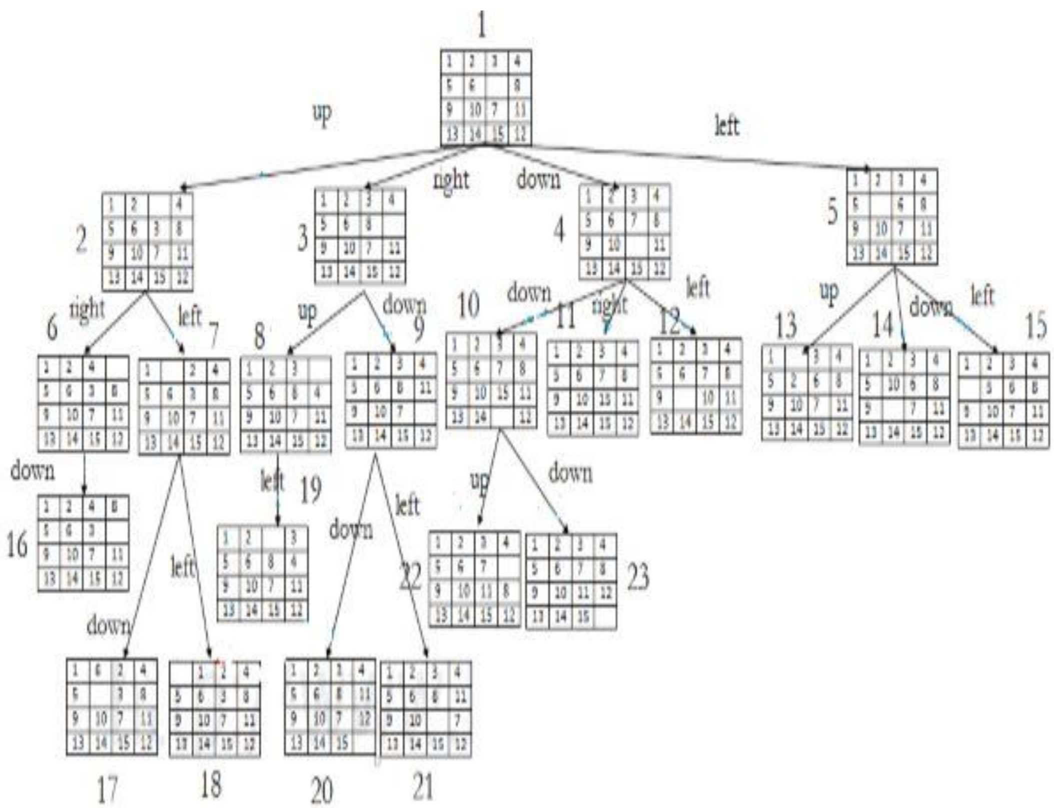
f(x) is length of path from root or initial node to node x,

g(x) is estimated length of path from x downward to the goal node. Number of non blank tile not in their correct position.

C(x)< Infinity.(initially set bound).

Each time node with smallest cost is selected for further expansion towards goal node. This node become the e-node.

State Space tree with node cost is shown in diagram.



Reference: Ques10;

Travelling Salesman Problem by Branch and Bound Technique

Given: A graph with nodes like A,B,C,D, called cities and connected with edges with their weight , called distance between cities.

Travelling Salesman Problem states-

1. A salesman has to visit every city exactly once.
2. He has to come back to the city from where he starts his journey.
3. What is the shortest possible route that the salesman must follow to complete his tour?

If salesman starting city is A, then a TSP tour is like -

A → B → D → C → A

Step-01:

Given the initial cost matrix and reduce it-

	A	B	C	D
A	∞	4	12	7
B	5	∞	∞	18
C	11	∞	∞	6
D	10	2	3	∞

- To reduce a matrix, perform the row reduction and column reduction of the matrix one by one.

  A row or a column is reduced if it contains at least one entry '0' in it.

  Check each and every row, If the row already contains an entry '0', then-

  There is no need to reduce that row.

else

Reduce that particular row by

Selecting the smallest number from that row.

Subtract that number from each element of that row.

Following this, we have-

1. Reduce the elements of row-1 by 4.
2. Reduce the elements of row-2 by 5.
3. Reduce the elements of row-3 by 6.
4. Reduce the elements of row-4 by 2.

Performing this, we obtain the following row-reduced matrix-

	A	B	C	D
A	∞	0	8	3
B	0	∞	∞	13
C	5	∞	∞	0
D	8	0	1	∞

Now we reduce columns of above row-reduced matrix one by one.

If the column already contains an entry ‘0’, then-

There is no need to reduce that column.

Else

If the column does not contains an entry ‘0’, then-

Reduce that particular column, by selecting smallest number from each column

Subtract that number from each element of that column.

This will create an entry ‘0’ in that column

Following this, we have-

1. Column-1 already have 0.
2. Column-2 already have 0.
3. Reduce the elements of column-3 by 1.
4. Column-4 already have 0. No need to reduce.

we obtain the following column-reduced matrix-

	A	B	C	D
A	∞	0	7	3
B	0	∞	∞	13
C	5	∞	∞	0
D	8	0	0	∞

matrix is completely reduced.

Now, we calculate the cost of initial node by adding all the reduction elements.

Cost of initial node(node-1) is

= Sum of all reduction elements
= 4 + 5 + 6 + 2 + 1
= 18

<u>Step-02:</u>
We consider all remaining vertices one by one by considering low cost node as e node for further expansion of state space diagram.
To go to : Node-2 (Path node-1 → node-2)

From the reduced matrix of step-01, M[1,2] = 0
Set row-1 and column-2 to ∞
Set M[2,1] = ∞
Now, resulting cost matrix is-

	A	B	C	D
A	∞	∞	∞	∞
B	∞	∞	∞	13
C	5	∞	∞	0
D	8	∞	0	∞

Now We reduce this matrix, by applying same process as step-1.Then, we find out the cost of node-02.

Row Reduction-
1. Row-1 all its elements are ∞.
2. Reduce all the elements of row-2 by 13.
3. Row-3 already have 0.
4. Row-4 already have 0.

Now we obtain the following row-reduced matrix-

	A	B	C	D
A	∞	∞	∞	∞
B	∞	∞	∞	0
C	5	∞	∞	0
D	8	∞	0	∞

Column Reduction-
1. Reduce the elements of column-1 by 5.
2. Column-2 all its elements are ∞.
3. Ccolumn-3 already have 0.

4. Column-4 already have 0.

Performing this, we obtain the following column-reduced matrix-

	A	B	C	D
A	∞	∞	∞	∞
B	∞	∞	∞	0
C	0	∞	∞	0
D	3	∞	0	∞

Now, we calculate the cost of node-2.
= Cost of node-1 + Sum of reduction elements + M[1,2]
= 18 + (13 + 5) + 0
= 36

To go to: Node-3 (Path 1 → 3)

1. The reduced matrix of step-01, M[1,3] = 7
2. Set row-1 and column-3 to ∞
3. Set M[3,1] = ∞

Now, resulting cost matrix is-

	A	B	C	D
A	∞	∞	∞	∞
B	0	∞	∞	13
C	∞	∞	∞	0
D	8	0	∞	∞

Now, we find out the cost of node-3.

Row Reduction-

1. Row-1 all its elements are ∞.
2. Row-2 already have 0.
3. Row-3 already have 0.
4. Row-4 already have 0.

The matrix is already row-reduced.

Column Reduction-

1. Column-1 already have 0.
2. Column-2 already have 0.
3. Column-3 all its elements are ∞.
4. Column-4 already have 0.

The matrix is already column reduced.
Now, the matrix is completely reduced.
Now, we calculate the cost of node-3.
= Cos of node-1 + Sum of reduction elements + M[1,3]
= 18 + 0 + 7
= 25

To go to node-4: (Path 1 → 4)

1. The reduced matrix of step-01, M[1,4] = 3
2. Set row-1 and column-4 to ∞
3. Set M[4,1] = ∞

	A	B	C	D
A	∞	∞	∞	∞
B	0	∞	∞	∞
C	5	∞	∞	∞
D	∞	0	0	∞

We find out the cost of node-4, by reducing this matrix

Row Reduction-

1. Row-1 all its elements are ∞.
2. Row-2 already have 0.
3. Reduce all the elements of row-3 by 5.
4. Row-4 already have 0.

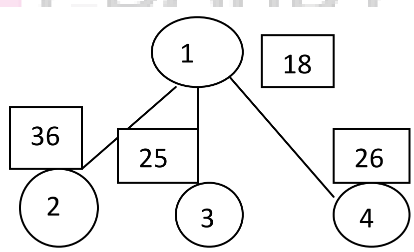
We obtain the following row-reduced matrix-

	A	B	C	D
A	∞	∞	∞	∞
B	0	∞	∞	∞
C	0	∞	∞	∞
D	∞	0	0	∞

Column Reduction-

1. Column-1 already have 0.
2. Column-2 already have 0.
3. Column-3 already have 0.
4. Column-4 all its elements are ∞.

The matrix is already column-reduced.
The matrix is completely reduced.
Now, we calculate the cost of node-4.
= Cost of node-1 + Sum of reduction elements + M[1,4]
= 18 + 5 + 3
= 26



Cost of node 2,3,4 is = 36, 25, 26 respectively as shown above state space diagram.
Circle represents Node and rectangle represents node cost.

We select the node with the lowest cost value. Cost for node-3 is lowest, we select node-3 to visit

i.e. path **1→ 3**.

Step-03:

We explore the node 2 and 4 from node-3. Start from the cost matrix at node-3 which is-

	A	B	C	D
A	∞	∞	∞	∞
B	0	∞	∞	13
C	∞	∞	∞	0
D	8	0	∞	∞

Cost of node 3 = 25

To go to node-2: Node-5 (Path 1 → 3 → 2)

1. The reduced matrix of step-02, M[3,2] = ∞
2. Set row-3 and column-2 to ∞
3. Set M[2,1] = ∞

Now, the cost matrix is-

	A	B	C	D
A	∞	∞	∞	∞
B	∞	∞	∞	13
C	∞	∞	∞	∞
D	8	∞	∞	∞

Now,We reduce this matrix,.we find out the cost of node-5.

Row Reduction-

1. Row-1 all its elements are ∞.
2. Reduce all the elements of row-2 by 13.
3. Row-3 all its elements are ∞.
4. Reduce all the elements of row-4 by 8.

To do this, we obtain the following reduced matrix-

	A	B	C	D
A	∞	∞	∞	∞
B	∞	∞	∞	0
C	∞	∞	∞	∞
D	0	∞	∞	∞

Column Reduction-

1. Column-1 already have 0.
2. Column-2 all its elements are ∞.
3. Column-3 all its elements are ∞.
4. Column-4 already have 0.

The matrix is already column reduced. The matrix is completely reduced.
we calculate the cost of node-5.
= cost of node 3 + Sum of reduction elements + M[3,2]
= 25 + (13 + 8) + ∞
= ∞

To go to node-4: Node-6 (Path 1 →3 → 4)

1. The reduced matrix of step-02, M[3,4] = ∞
2. Set row-3 and column-4 to ∞
3. Set M[4,1] = ∞

Now, resulting cost matrix is-

	A	B	C	D
A	∞	∞	∞	∞
B	0	∞	∞	∞
C	∞	∞	∞	∞
D	∞	0	∞	∞

Now, We reduce this matrix. We find out the cost of node-6.

Row Reduction-

1. Row-1 all its elements are ∞.
2. Row-2 already have 0.

3. Row-3 all its elements are ∞.
4. Row-4 all its elements are ∞.

the matrix is already reduced.

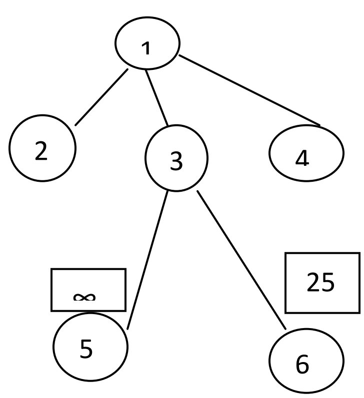
Column Reduction-

- column-1already have0.
- Column-2 all its elements are ∞.
- Column-3 all its elements are ∞.
- Column-4 all its elements are ∞.

The matrix is already reduced. The matrix is completely reduced.

Now, we calculate the cost of node-6.

= cost of node 3 + Sum of reduction elements + M[3,4]
= 25 + 0 + 0
= 25



Cost of node 5 = ∞ (Path 1 → 3 → 2)

Cost of node 6 = 25 (Path 1 → 3 → 4)

Select the node with the lowest cost. Cost for node-6 is lowest, we select to visit node-6. i.e. path 3 → 4.

Step-04:

We explore node 2 from node-6. Start with the cost matrix at node-6 which is-

	A	B	C	D
A	∞	∞	∞	∞
B	0	∞	∞	∞
C	∞	∞	∞	∞
D	∞	0	∞	∞

Cost of node 6 = 25

To go to node-2: Node-7 (Path 1 → 3 → 4 → 2)

1. The reduced matrix of step-03, M[4,2] = 0
2. Set row-4 and column-2 to ∞
3. Set M[2,1] = ∞

Now, the cost matrix is-

	A	B	C	D
A	∞	∞	∞	∞
B	∞	∞	∞	∞
C	∞	∞	∞	∞
D	∞	∞	∞	∞

Now, We reduce this matrix and find out the cost of node-7.

Row Reduction-

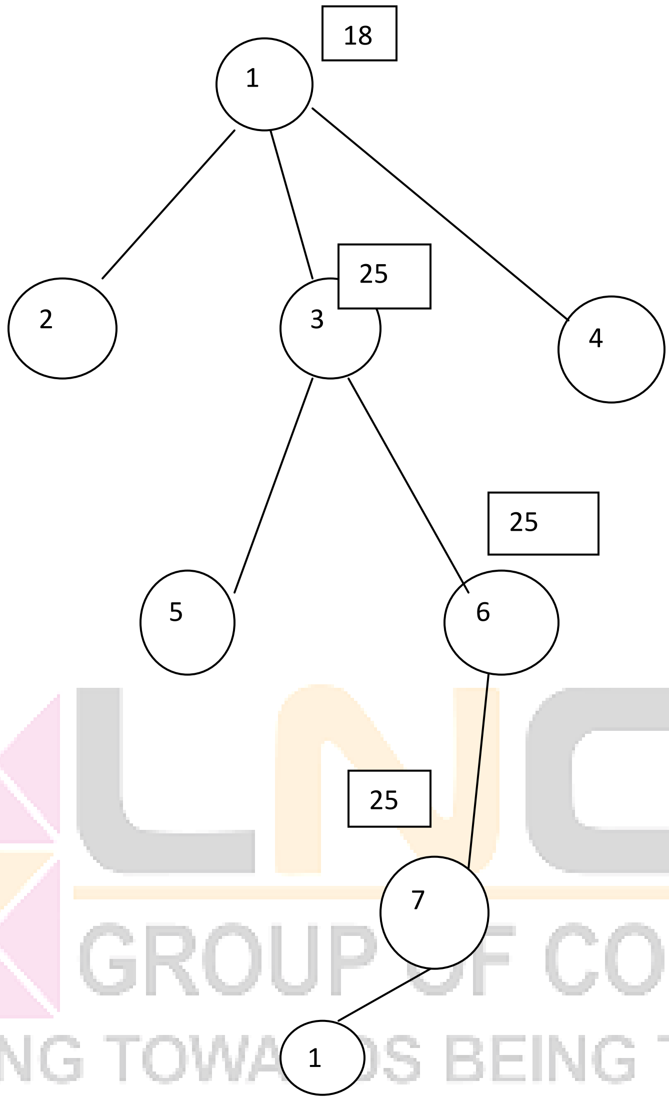
1. Row-1 all its elements are ∞.
2. Row-2 all its elements are ∞.
3. Row-3 all its elements are ∞.
4. Row-4 all its elements are ∞.

Column Reduction-

- Column-1 all its elements are ∞.
- Column-2 all its elements are ∞.
- Column-3 all its elements are ∞.
- Column-4 all its elements are ∞.

Finally, the matrix is completely reduced. All the entries have become ∞.

Now, we calculate the cost of node-7.
= cost of node 6 + Sum of reduction elements + M[4,2]
= 25 + 0 + 0
=25
Optimal path is 1—3—4—2—1. Cost of Optimal path = 25



State space diagram

Reference: **Author:** Akshay Singhal, **Publisher Name:** Gate Vidyalay

By : Bhawana Pillai