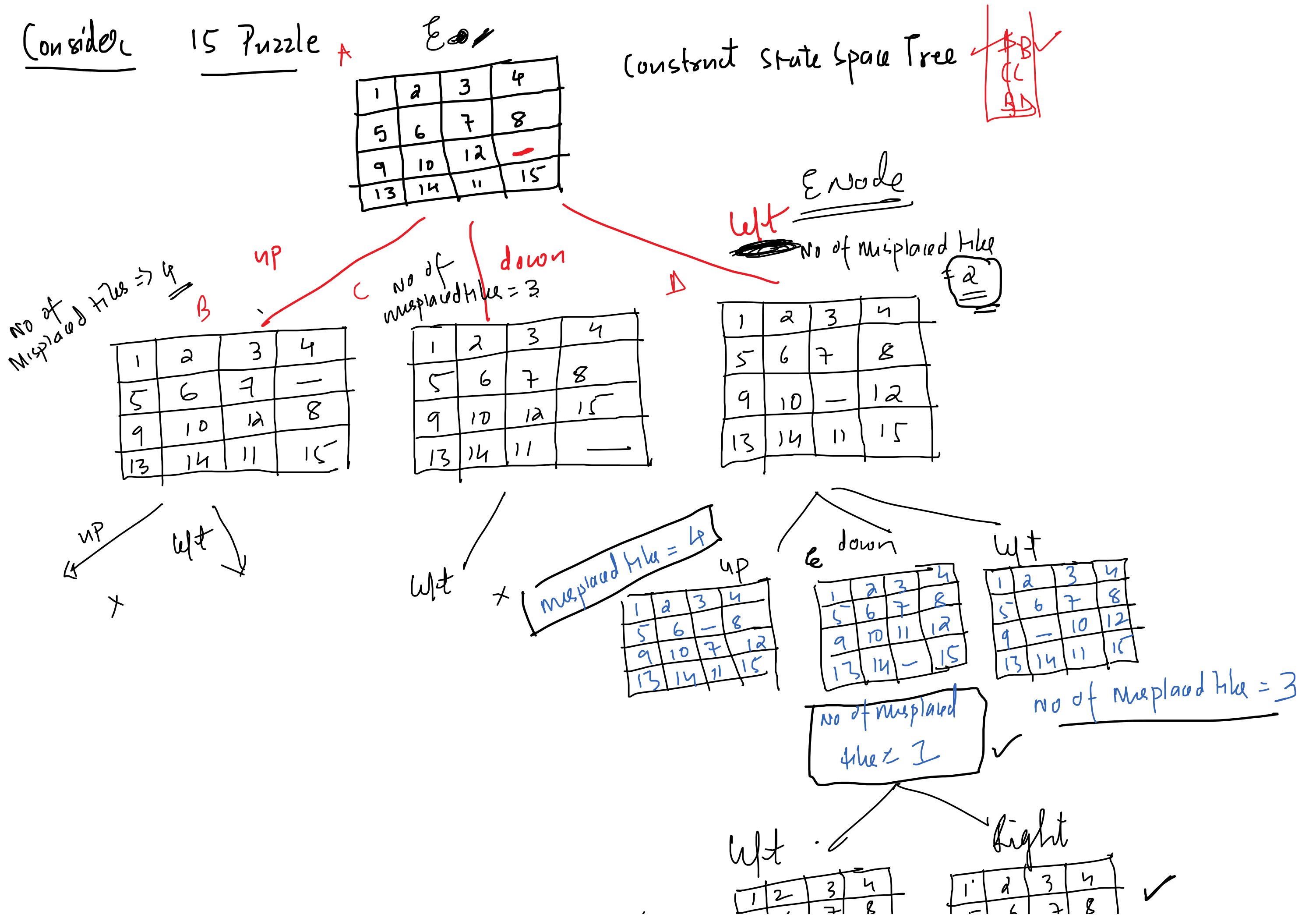
Branch and Bound -

It is a technique of exploring a directed graph or State Space Tree

* In this we calculate bound on each node which represents the cost of reaching to solution node (final) from that node.
* If the solution represented by bound, is worst than best solution found so far then that node is killed and corresponding branch is closed
* The bound can also be used to select most promising node out of available

Consider 15 Puzzle Eg

Construct State Space Tree



Left

1	2	3	4
5	6	7	8
9	10	11	12
13	–	14	15

No of misplaced tile = 2

Right

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	–

No of misplaced tile = 0

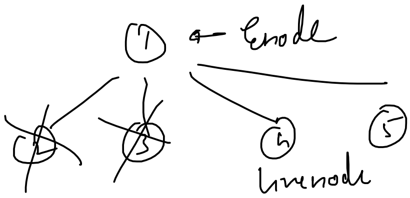
Goal State

Approach

E-Node ⇒ It is a node that is currently getting Expanded.

Live Node ⇒ It's a node which is created but not Expanded yet. In Branch and Bound at each E node we find all the live nodes that can be generated using single step

Those which are infeasible are killed and other are added to the list of live node



The following rules are followed in B & B ⇒

1. All the children of current E node are generated before selecting the next E-node
2. Each node may become an E-node exactly once.
3. Depending on policy of selecting next E-node B & B can be categorised as

A. LIFO BB / LIFO Search ⇒ Here next E node is selected using DFS.

(B) FIFO BB / FIFO Search ⇒ Here next E node is selected using BFS.

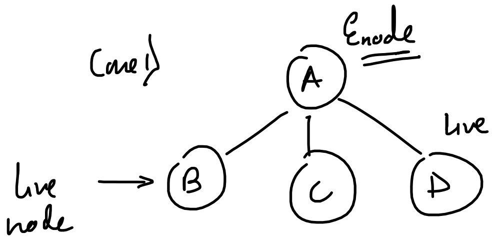
(C) LC B.B / LC Search [LC = Least Cost]

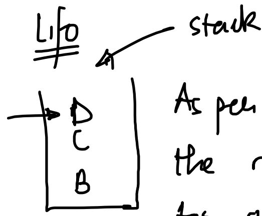
In this next E-node is selected from the live node having least cost.

* LC Search ~~was~~ uses Estimated Cost f^n

$\hat{c}(.)$

Case 1)

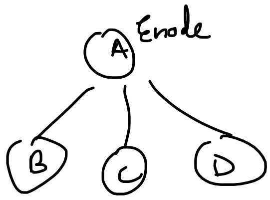


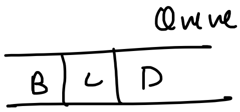


As per Lifo the node at top of stack will be next E-node (Here D)

Case 2)

Using FIFO

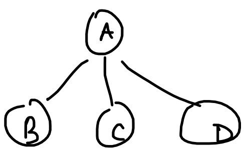




As per operation on queue data Structure the node B will be E-node

[But Both the above are not Efficient]

Least Cost



Next E-node = min(B, C, D) cost

⇒ It is Intelligent methodology and is efficient

Puzzle Problem

In this we have a square block frame of 4x4 dimension with 15 numbered tiles and an empty slot (ES)

The problem is to reach to goal state from Initial State

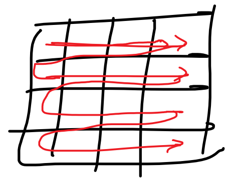
14	10	3	2
4	_	8	6
13	15	1	5
12	11	7	9

Initial State

$\longrightarrow$

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

Goal state



* A legal move can be represented as movement of ES
* Here there are maximum 4 legal moves possible

UP, DOWN, LEFT and RIGHT

* Hence we want to find set of legal moves to transform the initial state to goal state.
* A goal is reachable from initial state if and only if the initial state is reachable from goal state
* The Reachability can be as follows:

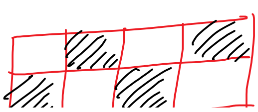
Checking Reachability

The initial state is reachable from goal state if

$$\boxed{\sum_{i=1}^{15} less(i) + X}$$ is Even

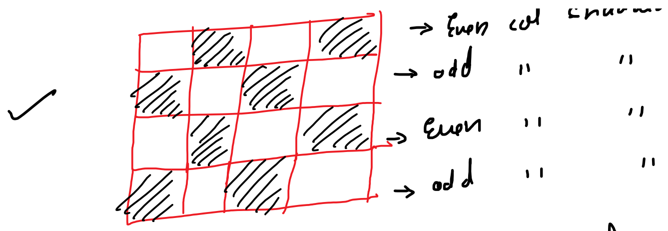
Here $\boxed{less(i)}$ = No of tiles having value < i but they are ahead of it.

$\boxed{X}$ can be computed by comparing initial state with following board arrangement



→ Even col shaded

→ odd " "

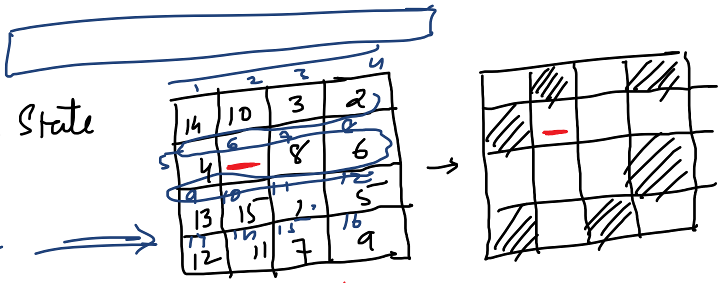


$x=1$ if ES in initial state matches with shaded slot

$x=0$ otherwise

Consider the Initial State

See here →



Initial state

Here the ES in Initial state does not matches with shaded slot. So $\boxed{X=0}$

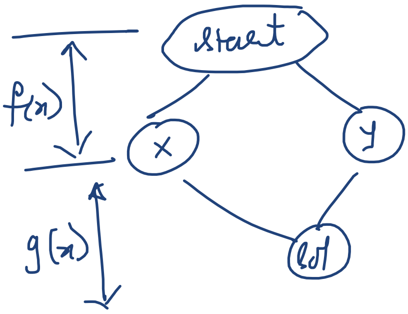
Now To calculate $\Rightarrow \sum_{i=1}^{15} \text{less}(i)$

i	less(i)
1	0
2	1
3	2
4	1
5	0
6	2
7	0
8	4
9	0
10	9
11	2
12	3
13	6
14	13
15	6
	49

$$\sum_{i=0}^{15} \text{less}(i) + X = 49 + 0 = \underline{\underline{49}} \quad \text{Not Even}$$

<u>The Goal state Here is not reachable</u>

<u>Compute</u> $\hat{c}(\,)$



$$\hat{c}(x) = f(x) + g(x).$$

In 15 puzzle problem

$f(x)$ = depth of node X.

$g(x)$ = No of Non black tiles not in their goal state

Travelling Salesman Problem Using Branch and Bound.

Approach →

* A better estimated cost function can be found using reduced cost matrix.

* [A row of the matrix is said to be reduced if it contains atleast one $\boxed{\text{Zero}}$ and the remaining elements are $\boxed{\text{Non-Zero}}$

Ex

10 2 7 ∞

↓ subs 2

8 0 5 ∞ ⇐ Reduced Row

* Similarly col can be reduced [atleast one Zero and other Non Zero]

A matrix is said to be reduced if all rows and all columns are reduced.]

To find $\hat{c}(n)$ → [Cost to reach destination from Current node n]

[Following steps are used to find $\hat{c}(n)$ of a node n representing an edge from i to j.

Step 1 → set all elements of row i and col j to ∞

Step 2 set $A[j, 1]$ to ∞

Step 3 : Reduce all rows and col of Resultant matrix except the rows & col containing ∞

Step 4 : Compute the total Reduction value

R → S
S will be child of R

$$\hat{c}(S) = \hat{c}(R) + A(i, j) + r$$

S = Current node representing edge from i to j

r = Total Reduction value (row + col)

$A(i, j)$ = cost of edge (i, j)

R ⇒ current node (S).

16 April 2021 14:58

Dec-14 Consider

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & \infty & 10 & 15 & 20 \\ 2 & 5 & \infty & 9 & 10 \\ 3 & 6 & 13 & \infty & 12 \\ 4 & 8 & 8 & 9 & \infty \end{array}$$

$X \Rightarrow$ vertex

(n) $\Rightarrow n^{th}$ node in Tree

Solve TSP using Branch & Bound

Sol^n Given cost matrix $\Rightarrow$

$$\begin{bmatrix} \infty & 10 & 15 & 20 \\ 5 & \infty & 9 & 10 \\ 6 & 13 & \infty & 12 \\ 8 & 8 & 9 & \infty \end{bmatrix}$$

To find $\hat{c}(1) \Rightarrow$ cost to reach final node from node 1

[For $\hat{c}(1) \Rightarrow$ direct Reduce method]

$$\begin{array}{c} 10 \\ 5 \\ 6 \\ 8 \end{array}\begin{bmatrix} \infty & 10 & 15 & 20 \\ 5 & \infty & 9 & 10 \\ 6 & 13 & \infty & 12 \\ 8 & 8 & 9 & \infty \end{bmatrix}$$

$\underline{29}$

Row Red^n Total Row Red^n = $\underline{29}$

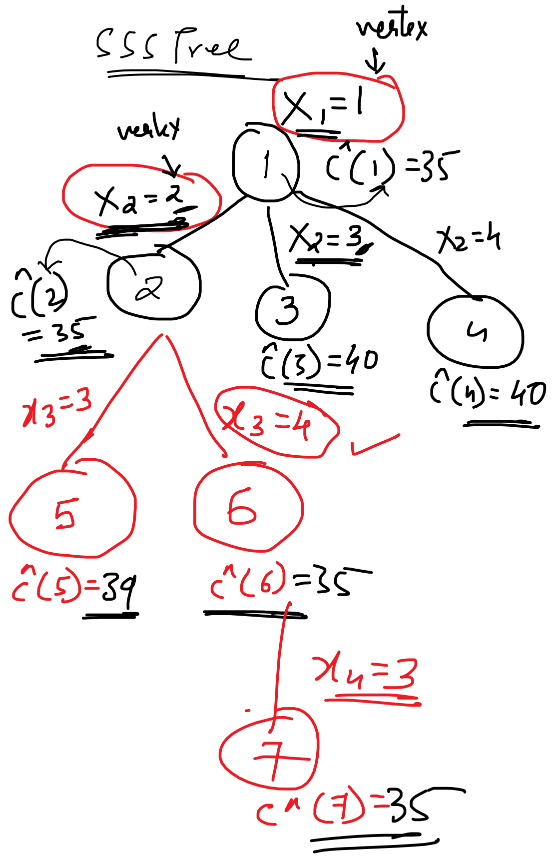
$$\begin{bmatrix} \infty & 0 & 5 & 10 \\ 0 & \infty & 4 & 5 \\ 0 & 7 & \infty & 6 \\ 0 & 0 & 1 & \infty \end{bmatrix}$$
$$\begin{array}{cccc} 0 & 0 & 1 & 5 \end{array}$$

Now Col^n Red^n $\Rightarrow$ $1+5 = \underline{6}$

$$\hat{c}(1) = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & \infty & 0 & 4 & 5 \\ 2 & 0 & \infty & 3 & 0 \\ 3 & 0 & 7 & \infty & 1 \\ 4 & 0 & 0 & 0 & \infty \end{array}$$

Total Red = $29 + 6 = 35$

SSS Tree



Total Red = 29 + 6 = 35

Step 2 find $\hat{C}(2)$ ⇒ cost of reaching final node from node 2.

$$\underset{i}{1} \rightarrow \underset{j}{2}$$

In $\hat{C}(1)$ make i^{th} row (1^{st} row) ⇒ ∞

j^{th} col (2^{nd} col) ⇒ ∞

make (2, 1) ⇒ ∞

$$\begin{array}{cc|cccc|}
 & & 1 & 2 & & 4 \\
0 & 1 & \infty & \infty & & \infty \\
0 & 2 & \infty & \infty & & 0 \\
0 & 3 & 0 & \infty & & 1 \\
0 & 4 & 0 & \infty & & \infty \\
 & & 0 & 0 & 0 & 0
\end{array}$$

Now perform Row and col Red^n

No Row and col^n reduction

(1→2) $r = 0$ → in $\hat{C}(1)$

$$\hat{C}(2) = \hat{C}(1) + A(i,j) + r$$
$$= 35 + 0 + 0 = 35$$

To find $\hat{C}(3)$ = (1→3) Cost to reach final node from node 3

$$\underset{1}{i} \rightarrow \underset{3}{j}$$

In $\hat{C}(1)$ = i^{th} row ⇒ 1^{st} row = ∞

j^{th} col ⇒ 3^{rd} col = ∞

$(j,1) = (3,1)$ ⇒ ∞

$$\begin{array}{c|cccc|}
 & 1 & 2 & 3 & 4 \\
1 & \infty & \infty & \infty & \infty \\
2 & 0 & \infty & \infty & 0 \\
3 & \infty & 7 & \infty & 1 \\
4 & 0 & 0 & \infty & \infty
\end{array}$$

$$\begin{array}{c} \\ 3 \\ 4 \end{array}\begin{bmatrix} 0 & & \infty & 0 \\ \infty & 7 & \infty & 1 \\ 0 & 0 & \infty & \infty \end{bmatrix}$$

$\underline{1}$ (No col Reduction)

$$\begin{bmatrix} \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & 0 \\ \infty & 6 & \infty & 0 \\ 0 & 0 & \infty & \infty \end{bmatrix}$$
$$\begin{array}{cccc} 0 & 0 & 0 & 0 \end{array}$$

Total Red: $1+0=\underline{1}$ $\boxed{r=1}$

$$\hat{c}(3) = \hat{c}(1) + A(1,3) + r$$

$A(1,3)$: In $\hat{c}(1)$

$$= 35 + 4 + 1$$
$$= \underline{\underline{40}}$$

$\hat{c}(4) \Rightarrow$ from $1 \to 4$

= cost of reaching final node from node 4

$$\begin{array}{ccc} 0 & & 0 \\ i & \xrightarrow{f} & j \\ 1 & & 4 \end{array}$$

In $\hat{c}(1) \Rightarrow$ 1st row $\Rightarrow \infty$

4th col $\Rightarrow \infty$

$(4,1) \Rightarrow \infty$

$$\begin{array}{c|c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{cc} 0 & 1 \\ 0 & 2 \\ 0 & 3 \\ 0 & 4 \end{array} & \begin{bmatrix} \infty & \infty & \infty & \infty \\ 0 & \infty & 3 & \infty \\ 0 & 7 & \infty & \infty \\ \infty & 0 & 0 & \infty \end{bmatrix} \\ & \begin{array}{cccc} 0 & 0 & 0 & 0 \end{array} \end{array}$$

No Row & col Reduction

$\underline{\underline{r=0}}$

$$\hat{c}(4) = \hat{c}(1) + A(1,4) + r = 35 + 5 + 0 = \underline{\underline{40}}$$

$\hat{c}(4)$: 4th node of SSS Tree

$\hat{c}(1)$: root

$A(1,4)$: In $\hat{c}(1)$

$\hat{c}(2) = 35 \to$ i.e cost to visit vertex 2 from vertex 1

$\hat{c}(3) = 40$ i.e " " " " 3 " " 1

$\hat{c}(4) = 40$ i.e " " " " 4 from " 1

Least cost is $\underline{\hat{c}(2)}$ i.e To reach 2nd node in SSST ie now we visited from vertex 1 $\longrightarrow$ 2

now at vertex (at $\hat{c}(2)$ we have options to

Visit vertex 3 (generate node 5 in Tree)

or vertex 4 (generate node 6 in Tree)

$\hat{c}(5)$ = Here $\underline{\underline{2 \rightarrow 3}}$ (i, j)

In $\hat{c}(2)$ make 2^{nd} row $= \infty$

make 3^{rd} col $= \infty$

$(3,1) \Rightarrow \infty$

$$
\begin{array}{cc|cccc|}
 & & 1 & 2 & 3 & 4 \\
0 & 1 & \infty & \infty & \infty & \infty \\
0 & 2 & \infty & \infty & \infty & \infty \\
1 & 3 & \infty & \infty & \infty & 1 \\
0 & 4 & 0 & \infty & \infty & \infty
\end{array}
\xrightarrow[\text{Redn}]{\text{After Row}}
\begin{array}{c|cccc|}
 & 1 & 2 & 3 & 4 \\
1 & \infty & \infty & \infty & \infty \\
2 & \infty & \infty & \infty & \infty \\
3 & \infty & \infty & \infty & 0 \\
4 & 0 & \infty & \infty & \infty
\end{array}
$$

No col Redn

$\underline{\underline{r = 1}}$

$$\hat{c}(5) = \underset{\text{source}}{\hat{c}(2)} + \underset{\text{in } \hat{c}(2)}{A(2,3)} + r = 35 + 3 + 1 = \underline{\underline{39}}$$

$\hat{c}(6)$ going from vertex 2 to 4 (i, j)

In $\hat{c}(2)$ make 2^{nd} row ∞

4^{th} col ∞

$(4,1)$ ∞

$$C^n(6) = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & \infty & \infty & \infty & \infty \\ 2 & \infty & \infty & \infty & \infty \\ 3 & 0 & \infty & \infty & \infty \\ 4 & \infty & \infty & 0 & \infty \end{array}$$

Here no row & col Reduction $\underline{\underline{r = 0}}$

$$C^n(6) = C^n(2) + \overset{\rightarrow \text{In } C^n(2)}{A(2,4)} + r = 35 + 0 + 0 = \underline{\underline{35}}$$

Now $C^n(5)$ i.e from vertex $2 \rightarrow 3 \Rightarrow 39$ cost
$C^n(6)$ i.e from vertex $2 \rightarrow 4 \Rightarrow 35$ cost

So we will visit vertex 4 from 2

Now we are at vertex 4 i.e <u>node 6</u> of Tree
We have only one node to visit i.e 3 from 4
and this will generate <u>node 7</u> of Tree

$C^n(7)$ will be created using $\underline{\underline{C^n(6)}}$ $\quad \underset{4}{i} \rightarrow \underset{3}{j}$

In $C^n(6)$ $\Rightarrow$ make 4^{th} row $= \infty$
3^{rd} col $= \infty$
$(3,1)$ –

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & \infty & \infty & \infty & \infty \\ 2 & \infty & \infty & \infty & \infty \\ 3 & \infty & \infty & \infty & \infty \\ 4 & \infty & \infty & \infty & \infty \end{array}$$

No row & Col Reduction $r = 0$

No row & Col Reduction $r = 0$

$$\hat{c}(7) = \hat{c}(6) + A(4,3) + r$$

$\rightarrow$ In $\hat{c}(2)$

$= 35 + 0 + 0$

$= \underline{\underline{35}}$

Path $\quad \underline{\underline{1 \rightarrow 2 \rightarrow 4 \rightarrow 3 - 1}}$