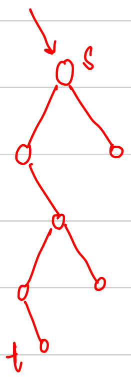


Lecture - III 9th Jan 2023

What's the proof that BFS gives us the shortest path?

Proof: By contradiction.

What if we use weighted edges? → We use Dijsktra Algorithm. $\curvearrowright$ Time Compl. = 2

Bellman Ford Algo:

1. For each vertex v, assign to it the distance d_v from the source vertex. Initialize d_i to ∞ $\forall i$, except for the source vertex $d_s = 0$

2. Repeat $|V|-1$ times: $\longrightarrow$ why $|V|-1$

```
        for every edge (u,v)
              if $d_u + w < d_v$
                    then: $d_v \leftarrow d_u + w$
```

3. If for any edge (u,v), $d_u + v < d_v$, then report the existence of a negative cycle. Else each d_v gives the distance of v from the source. Path can be generated by assigning a neighbour u as predecessor of v s/t $d_u + w = d_v$.

Lecture - V

16th Jan '23

We've seen two types of search algorithms : Tree- Search (problem) and Graph-Search (problem)

We cannot use DFS nor BFS in case of a game where both depth & breadth are infinite :

eg: We use BFS for each a no. x from y using only $\sqrt{\ }$ and !

But if $y = 4^k$ and $x = 4$ then the breadth & depth are both infinite in this case, so neither BFS nor DFS are possible.

Sol^n to this can be we can incrementally go on with increasing both depth & breadth.

§ Informed Search Methods :

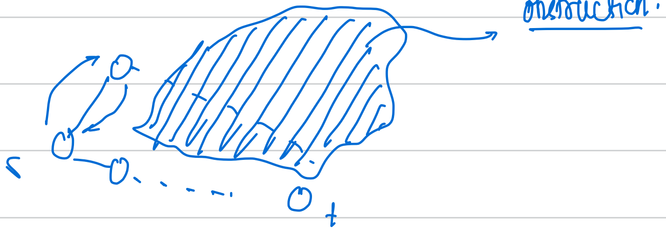
→ This uses problem specific knowledge (efficient than uninformed search strategies)

Best first search :

First of all we need to consider an evaluation function based on the conditions.

We shall greedily try to expand the node based on which one of them is closest to the goal.

But what if :



A* Search : So, we need to combine h(n) = cost to get from the node to the goal

g(n) = cost to reach the given node.

A* Search:

1) h(n) : should be an admissible heuristic (for tree search)
2) For graph search: consistency is needed.

Disadvantages of A*:

i) Breadth is a problem cuz A* is basically Dijkstra + heuristic function, and dijkstra is basically BFS. (You'll run out of space)

→ We can use Recursive Best first search instead to prevent space overflow.