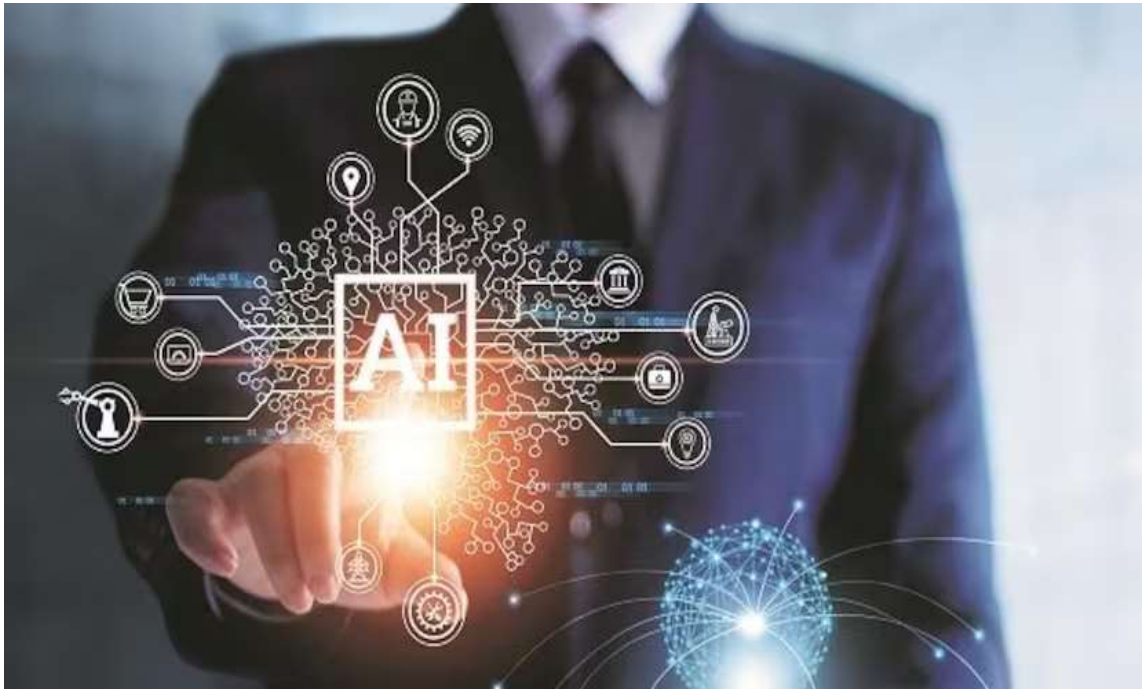
D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

Advanced AI



Dr.Vaishnaw G.Kale
Associate Professor
Department of Computer Science
Engineering & Applications
D.Y.Patil International University, Pune

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

About the Course

Name of the subject:Advanced AI
Course Code:MCA 301
Programme:Master in Computer Applications
Year of study:Second Year
Semester :III
Specialization:Artificial Intelligence and Data Science
Course Type:Specialization Course
Academic Year:2023-24

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Syllabus

Module-II:Problem Solving

Solving Problems by Searching, Problem-Solving Agents, Search Algorithms: Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Heuristic Functions, Local Search and Optimization Problems.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Advanced Artificial Intelligence

Module 02. Problem Solving

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

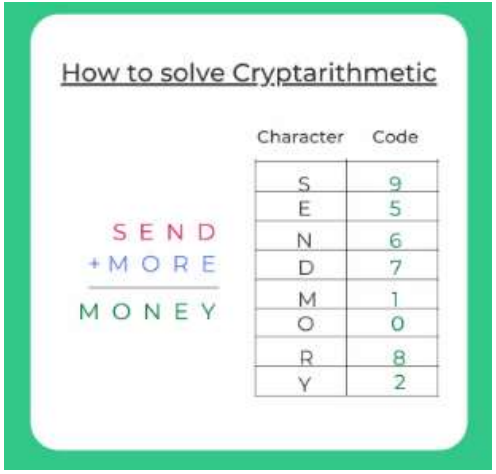
Problem Solving Through Search

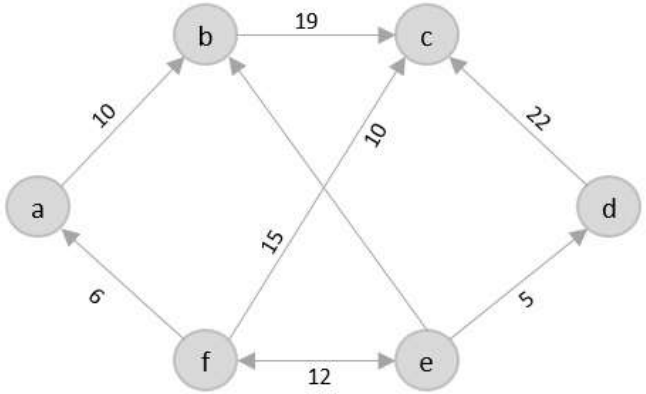
- In computer science, problem-solving refers to artificial intelligence techniques, including various techniques such as forming efficient algorithms, heuristics, and performing root cause analysis to find desirable solutions.
- In today's fast-paced digitized world, artificial intelligence techniques are used widely to automate systems that can use the resource and time efficiently.
- Some of the well-known problems experienced in everyday life are games and puzzles.
- Using AI techniques, we can solve these problems efficiently. In this sense some of the most common problems resolved by AI are

1) Travelling Salesman Problem,
2) Tower of Hanoi Problem,
3) Water-Jug Problem
4) N-Queen Problem
5) Chess, •Sudoku
6) Cryptarithmetic Problems,
7) Magic square, Logical Puzzles and so on….

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Through Search



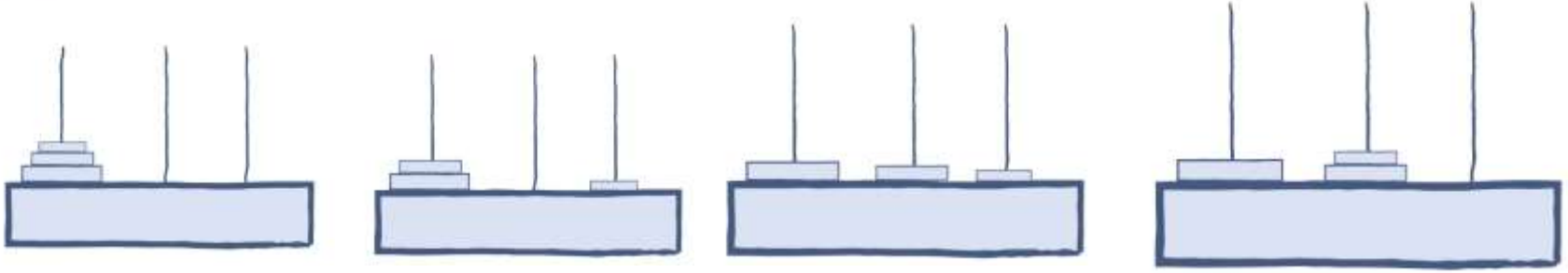


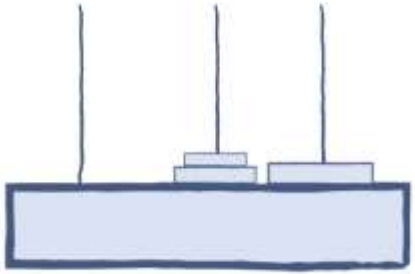
Travelling Salesman Problem

If you look at the graph below, considering that the salesman starts from the vertex ‘a’, they need to travel through all the remaining vertices b, c, d, e, f and get back to ‘a’ while making sure that the cost taken is minimum.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Through Search





The **Tower of Hanoi,** is a mathematical problem which consists of three rods and multiple disks. Initially, all the disks are placed on one rod, one over the other in ascending order of size similar to a cone-shaped tower.

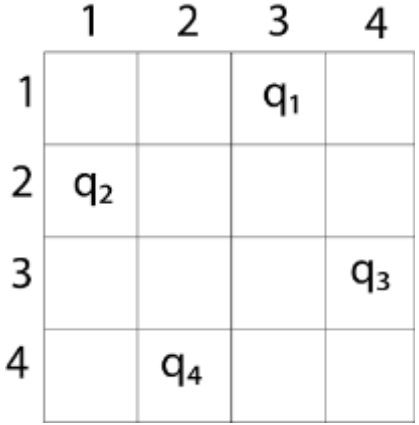
The objective of this problem is to move the stack of disks from the initial rod to another rod, following these rules:

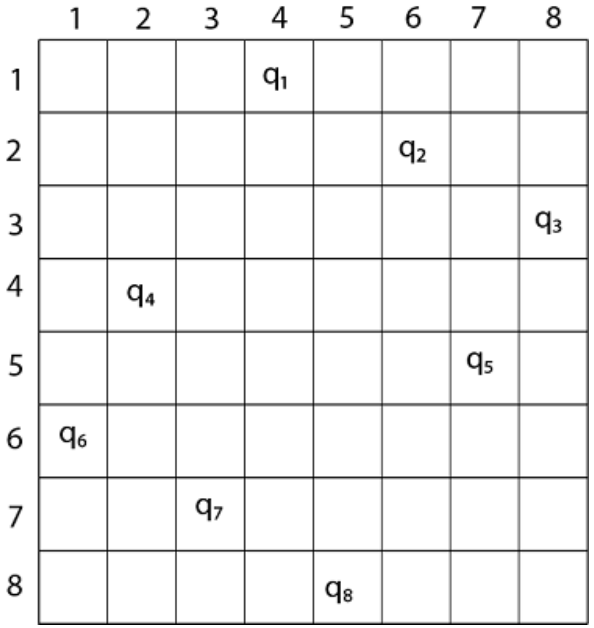
- A disk cannot be placed on top of a smaller disk
- No disk can be placed on top of the smaller disk.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Through Search

- Since, we have to place 4 queens such as q_1 q_2 q_3 and q_4 on the chessboard, such that no two queens attack each other. In such a conditional each queen must be placed on a different row,
- The implicit tree for 4 - queen problem for a solution (2, 4, 1, 3)





Thus, the solution **for** 8 - queen problem **for** (4, 6, 8, 2, 7, 1, 3, 5).

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Through Search

Water Jug Problem

You are on the side of the river. You are given a **m** liter jug and a **n** liter jug where **0 < m < n**.

Both the jugs are initially empty.

The jugs don't have markings to allow measuring smaller quantities.

You have to use the jugs to measure d liters of water where d < n.

Determine the minimum no of operations to be performed to obtain d liters of water in one of jug.

The operations you can perform are:

- Empty a Jug
- Fill a Jug
- Pour water from one jug to the other until one of the jugs is either empty or full.

Solution 01

- For example, if we have a jug J1 of 5 liters (n = 5) and another jug J2 of 3 liters (m = 3) and we have to measure 1 liter of water using them. The associated equation will be $5n + 3m = 1$.

Solution 02

- Suppose there are a 3 liter jug and a 5 liter jug to measure 4 liters water so m = 3,n = 5 and d = 4. The associated Diophantine equation will be $3m + 5n = 4$.
- Diophantine equation of the form $mx + ny = d$

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Through Search

The process of problem-solving using searching consists of the following steps.

1) Define the problem
2) Analyze the problem
3) Identification of possible solutions
4) Choosing the optimal solution
5) Implementation

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Agents

- Intelligent agents are supposed to maximize its performance measure. Achieving this can be simplified if the agent can adopt a goal and aim to satisfy it.
- Setting goals help the agent organize its behavior by limiting the objectives that the agent is trying to achieve and hence the actions it needs to consider.
- This Goal formulation based on the current situation and the agent's performance measure is the first step in problem solving.
- We consider the agent's goal to be a set of states.
- The agent's task is to find out actions in the present and in the future that could reach the goal state from the present state.
- Problem formulation is the process of deciding what actions and states to consider, given a goal.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Agents

- After Goal formulation and problem formulation, the agent has to look for a sequence of actions that reaches the goal. This process is called Search. A search algorithm takes a problem as input and returns a sequence of actions as output.
- After the search phase, the agent has to carry out the actions that are recommended by the search algorithm. This final phase is called execution phase.
- Before we get into more about problem formulating phase, we need to first understand what a problem is in terms of problem solving agents.The problem can be defined formally in five components:

1. Initial State
2. Actions
3. Transition Model
4. Goal Test
5. Path Cost

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Agents

1)Initial State

- The first component that describes the problem is the initial state that the agent starts in.
- For example, if a taxi agent needs to get to location(B) but the taxi is currently at location(A) then the initial state of the problem would be location(A).

2)Actions

- The second component that describes the problem is a description of the possible actions available to the agent.
- Given a state *s*, Actions(a) returns the set of actions that can be executed in *s*. We say that each of these actions is applicable in *s*.

3) Transition Model

- The third component is the description of what each action does which is called the transition model. It is specified by a function Result(s , a) that returns the state that results from doing action *a* in state s

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Problem Solving Agents

4)Goal Test

- The goal test determines whether a given state is a goal state or not.
- Sometimes there is an explicit set of possible goal states and the test simply checks whether the given state is one of them.
- Sometimes the goal is specified by an abstract property rather than an explicitly enumerated set of states.

5)Path Test

- The last component of the problem is the **path cost** which is a function that assigns a numeric cost to each path.
- The problem solving agent chooses a cost function that reflects its own performance measure.
- The **solution** to the problem is an action sequence that leads from initial state to goal state and the solution quality is measured by the path cost function.
- An optimal solution has the lowest path cost among all the solutions.

Example: There is a vacuum cleaner agent and it can move left or right and its jump is to suck up the dirt from the floor.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

There are two types of search Algorithms

1) Uninformed Search
2) Informed Search

1) Uninformed Search

- The uninformed search algorithm does not have any domain knowledge such as closeness, location of the goal state, etc. it behaves in a brute-force way.
- It only knows the information about how to traverse the given tree and how to find the goal state.
- This algorithm is also known as the Blind search algorithm or Brute -Force algorithm.
- Six Types of Uninformed Search

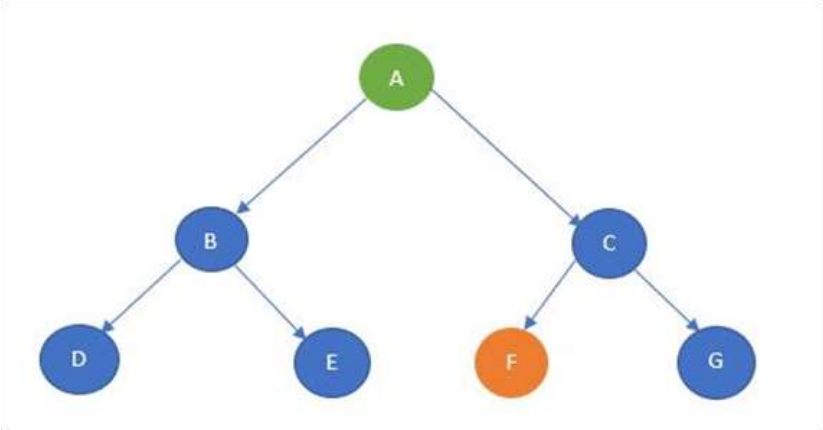
a) Breadth-first search b)Depth-first search c) Depth-limited search d) Iterative deepening depth-first search e) Bidirectional search f) Uniform cost search

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

a) Breadth First Search

- It is of the most common search strategies.
- It generally starts from the root node and examines the neighbor nodes and then moves to the next level.
- It uses First-in First-out (FIFO) strategy as it gives the shortest path to achieving the solution.
- BFS is used where the given problem is very small and space complexity is not considered. Now, consider the following tree.

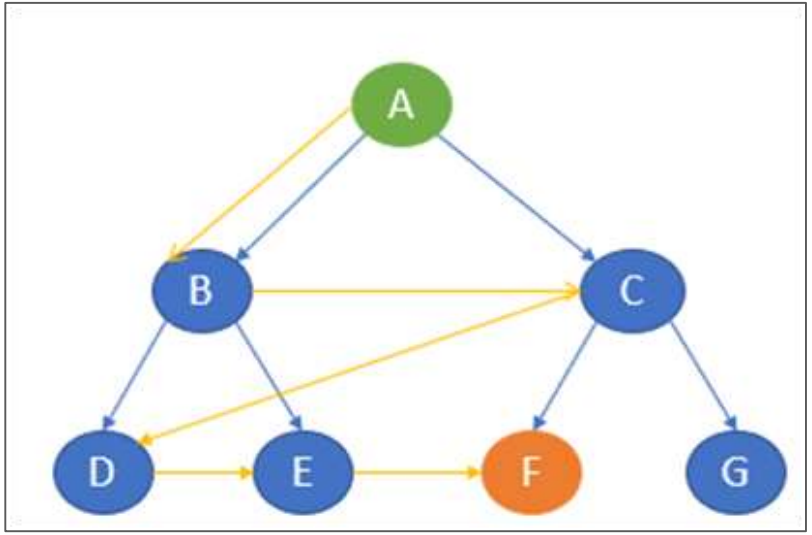


D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

a) Breadth First Search

- Here, let's take node A as the start state and node F as the goal state.
- The BFS algorithm starts with the start state and then goes to the next level and visits the node until it reaches the goal state.
- In this example, it starts from A and then travel to the next level and visits B and C and then
- travel to the next level and visits D, E, F and G. Here, the goal state is defined as F. So, the traversal will stop at F.



The path of traversal is:
A —-> B —-> C —-> D —-> E —-> F

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

a) **Breadth First Search**

Advantages of BFS

- BFS will never be trapped in any unwanted nodes.
- If the graph has more than one solution, then BFS will return the optimal solution which provides the shortest path.

Disadvantages of BFS

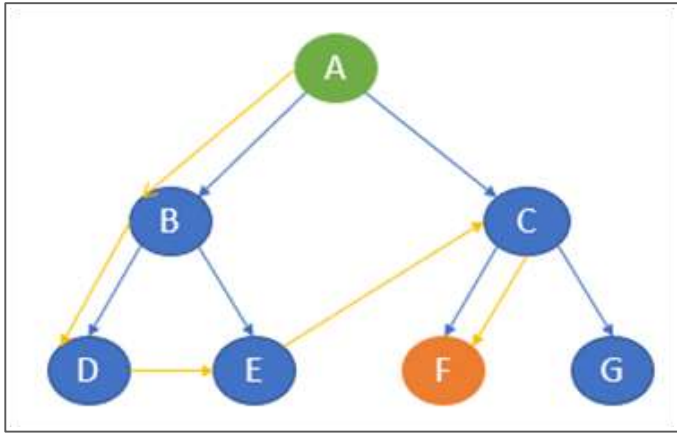
- BFS stores all the nodes in the current level and then go to the next level. It requires a lot of memory to store the nodes.
- BFS takes more time to reach the goal state which is far away.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

b) Depth First Search

- The depth-first search uses Last-in, First-out (LIFO) strategy and hence it can be implemented by using stack. DFS uses backtracking. That is, it starts from the initial state and explores each path to its greatest depth before it moves to the next path.
- DFS will follow Root node —-> Left node —-> Right node
- Here, it starts from the start state A and then travels to B and then it goes to D. After reaching
- D, it backtracks to B. B is already visited, hence it goes to the next depth E and then backtracks to B. as it is already visited, it goes back to A. A is already visited. So, it goes to C and then to F. F is our goal state and it stops there.



The path of traversal is:

A —-> B —-> D —-> E —->C—-> F

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

b) Depth First Search

Advantages of DFS

- It takes lesser memory as compared to BFS.
- The time complexity is lesser when compared to BFS.
- DFS does not require much more search.

Disadvantages of DFS

- DFS does not always guarantee to give a solution.
- As DFS goes deep down, it may get trapped in an infinite loop.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

c) Depth Limited Search

- Depth-limited works similarly to depth-first search. The difference here is that depth-limited search has a pre-defined limit up to which it can traverse the nodes. Depth-limited search solves one of the drawbacks of DFS as it does not go to an infinite path.
- DLS ends its traversal if any of the following conditions exists.

Standard Failure

- It denotes that the given problem does not have any solutions.

Cut off Failure Value

- It indicates that there is no solution for the problem within the given limit.

Advantages of DLS

- It takes lesser memory when compared to other search techniques.

Disadvantages of DLS

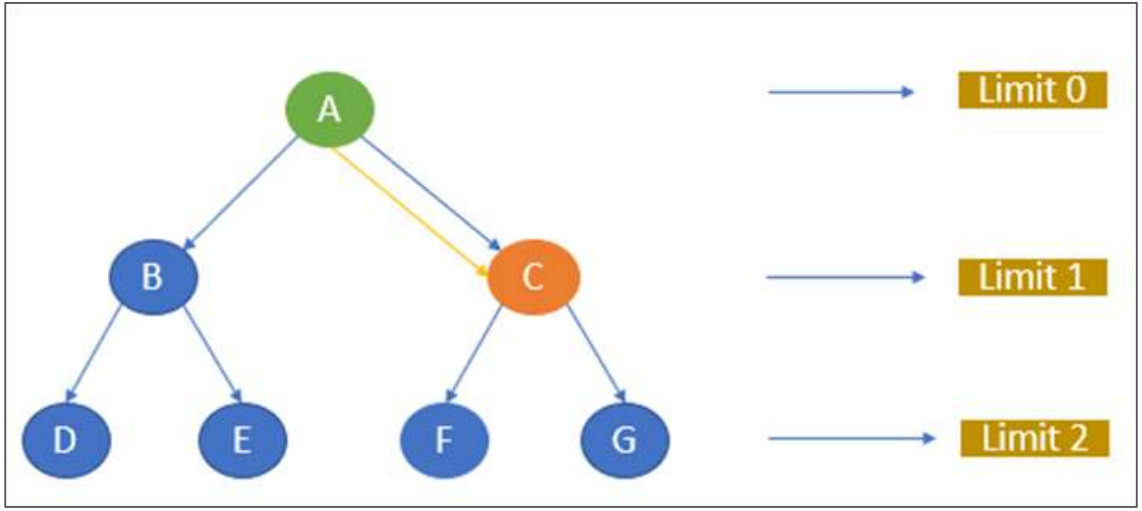
- DLS may not offer an optimal solution if the problem has more than one solution.
- DLS also encounters incompleteness.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

c) Depth Limited Search

- Let's take A as the start node and C as the goal state and limit as 1.
- The traversal first starts with node A and then goes to the next level 1 and the goal state C is there. It stops the traversal.



The path of traversal is:

A —-> C

If we give C as the goal node and the limit as 0, the algorithm will not return any path as the goal node is not available within the given limit. If we give the goal node as F and limit as 2, the path will be A, C, F.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

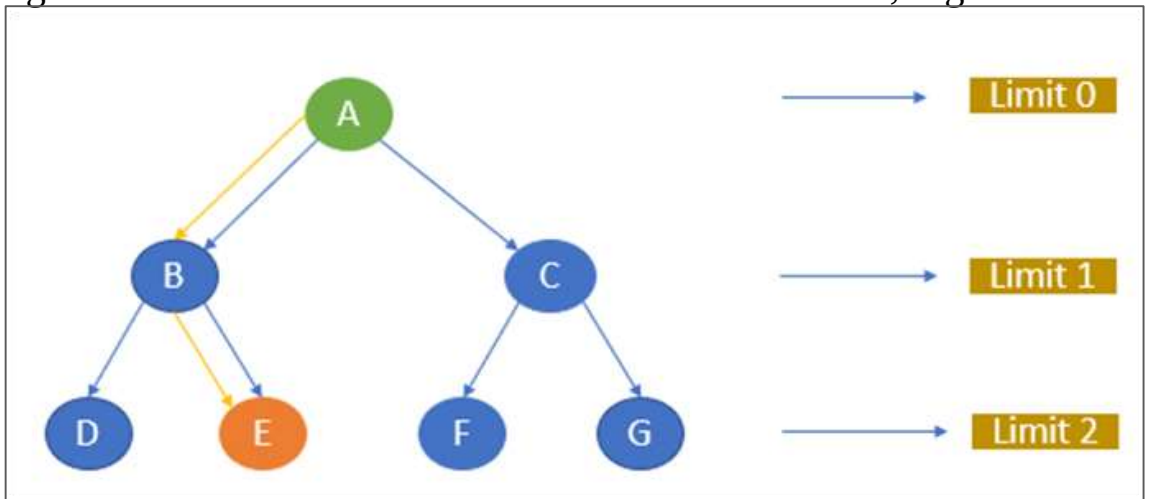
Search Algorithms

d) Iterative deepening depth-first search(IDDFS)

- Iterative deepening depth-first search is a combination of depth-first search and breadth-first search. IDDFS find the best depth limit by gradually adding the limit until the defined goal state is reached.
- Consider, A as the start node and E as the goal node. Let the maximum depth be 2.
- The algorithm starts with A and goes to the next level and searches for E. If not found, it goes to the next level and finds E.

The path of traversal is

A —-> B —-> E



D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

d) Iterative deepening depth-first search(IDDFS)

Advantages of IDDFS

- IDDFS has the advantages of both BFS and DFS.
- It offers fast search and uses memory efficiently.

Disadvantages of IDDFS

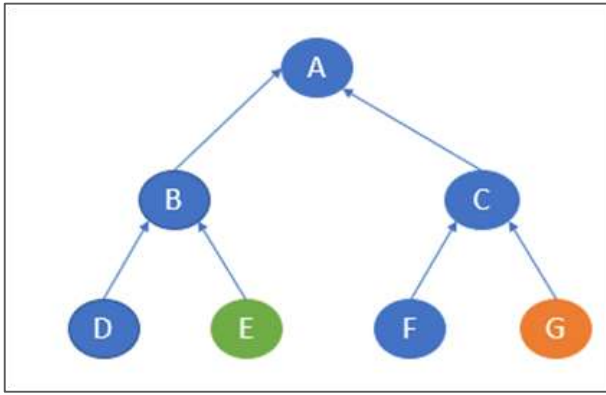
- It does all the works of the previous stage again and again.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

e)Bidirectional Search

- The bidirectional search algorithm is completely different from all other search strategies. It executes two simultaneous searches called forward-search and backwards-search and reaches the goal state. Here, the graph is divided into two smaller sub-graphs.
- In one graph, the search is started from the initial start state and in the other graph, the search is started from the goal state. When these two nodes intersect each other, the search will be terminated.
- Bidirectional search requires both start and goal start to be well defined and the branching factor to be the same in the two directions.

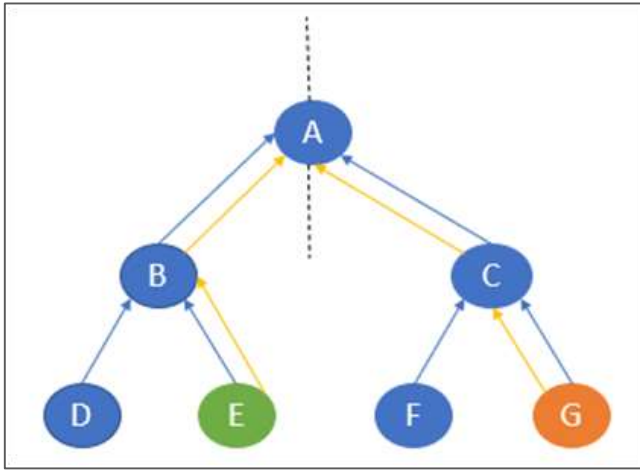


D Y PATIL INTERNATIONAL UNIVERSITY AKURDI PUNE

Search Algorithms

e)Bidirectional Search

- Here, the start state is E and the goal state is G.
- In one sub-graph, the search starts from E and in the other, the search starts from G.
- E will go to B and then A.
- G will go to C and then A.
- Here, both the traversal meets at A and hence the traversal ends.



The path of traversal is

E —-> B —-> A —-> C —-> G

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

e)Bidirectional Search

Advantages of bidirectional search

- This algorithm searches the graph fast.
- It requires less memory to complete its action.

Disadvantages of bidirectional search

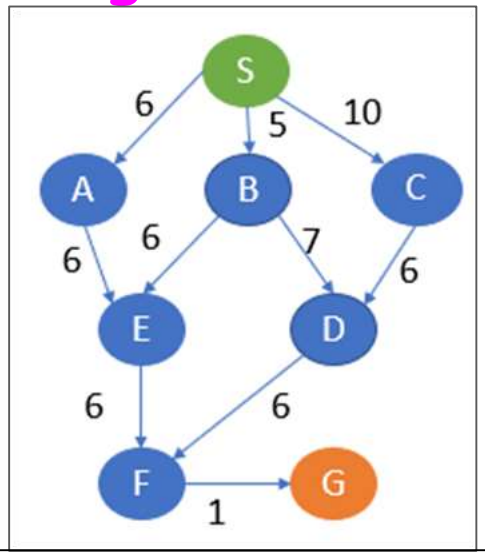
- The goal state should be predefined.
- The graph is quite difficult to implement.

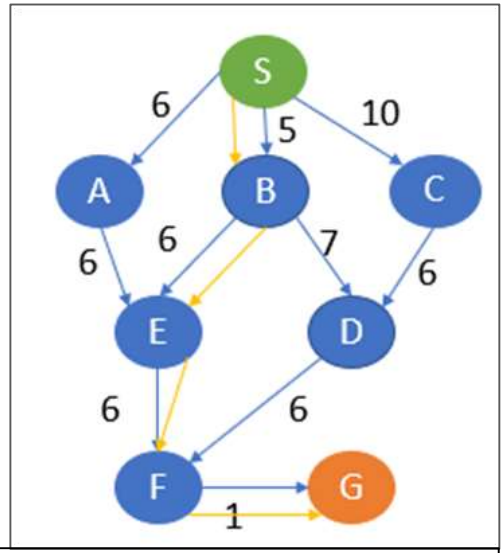
D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

f) Uniform Cost Search

- Uniform cost search is considered the best search algorithm for a weighted graph or graph with costs.
- It searches the graph by giving maximum priority to the lowest cumulative cost.
- Uniform cost search can be implemented using a priority queue.
- Consider the below graph where each node has a predefined cost.





Here, S is the start node and G is the goal node.

From S, G can be reached in the following ways.

S, A, E, F, G -> 19

S, B, E, F, G -> 18

S, B, D, F, G -> 19

S, C, D, F, G -> 23

Here, the path with the least cost is S, B, E, F, G.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

f) Uniform Cost Search

Advantages of UCS

- This algorithm is optimal as the selection of paths is based on the lowest cost.

Disadvantages of UCS

- The algorithm does not consider how many steps it goes to reach the lowest path.
- This may result in an infinite loop also.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

2)Informed Search

- The informed search algorithm is also called heuristic search or directed search.
- In contrast to uninformed search algorithms, informed search algorithms require details such as distance to reach the goal, steps to reach the goal, cost of the paths which makes this algorithm more efficient.
- Here, the goal state can be achieved by using the heuristic function.
- The heuristic function is used to achieve the goal state with the lowest cost possible.
- This function estimates how close a state is to the goal.
- Two Types of Informed search

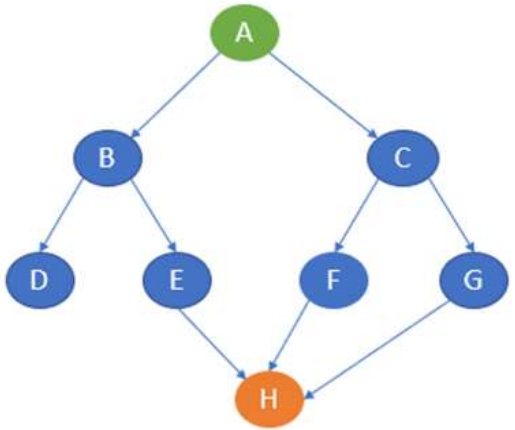
a)Greedy Best Fit Search Algorithm b) A* Search algorithm

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

a) Greedy Best Fit Search Algorithm

- Greedy best-first search uses the properties of both depth-first search and breadth-first search.
- Greedy best-first search traverses the node by selecting the path which appears best at the moment.
- The closest path is selected by using the heuristic function.
- Consider the below graph with the heuristic values



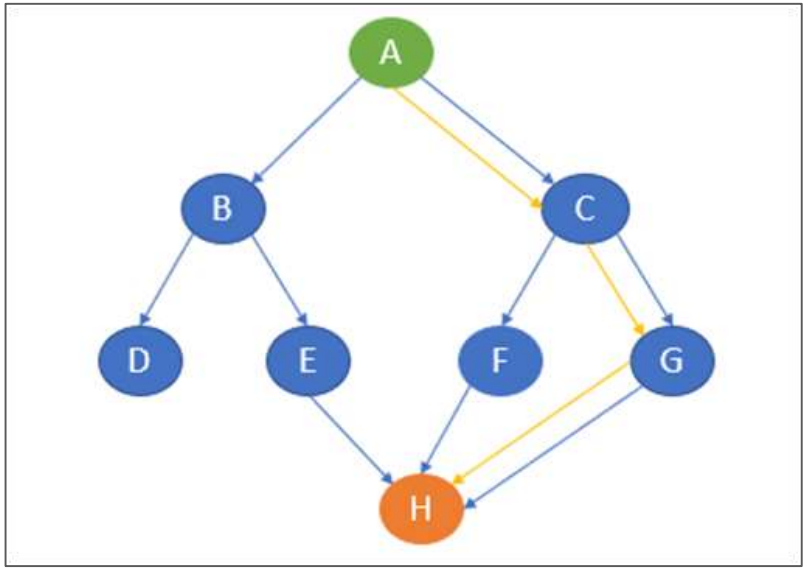
NODES	HEURISTICS
A	13
B	12
C	4
D	7
E	3
F	8
G	2
H	0

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

a) Greedy Best Fit Search Algorithm

- Here, A is the start node and H is the goal node.
- Greedy best-first search first starts with A and then examines the next neighbor B and C. Here, the heuristics of B is 12 and C is 4. The best path at the moment is C and hence it goes to C. From C, it explores the neighbors F and G. the heuristics of F is 8 and G is 2. Hence it goes to G. From G, it goes to H whose heuristic is 0 which is also our goal state.



The path of traversal is

A —-> C —-> G —-> H

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

a) Greedy Best Fit Search Algorithm

Advantages of Greedy best-first search

- Greedy best-first search is more efficient compared with breadth-first search and depth-first search.

Disadvantages of Greedy best-first search

- In the worst-case scenario, the greedy best-first search algorithm may behave like an unguided DFS.
- There are some possibilities for greedy best-first to get trapped in an infinite loop.
- The algorithm is not an optimal one.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

b)A* Search Algorithm

- A* search algorithm is a combination of both uniform cost search and greedy best-first search algorithms.
- It uses the advantages of both with better memory usage.
- It uses a heuristic function to find the shortest path. A* search algorithm uses the sum of both the cost and heuristic of the node to find the best path.

Advantages of A* search algorithm

- This algorithm is best when compared with other algorithms.
- This algorithm can be used to solve very complex problems also it is an optimal one.

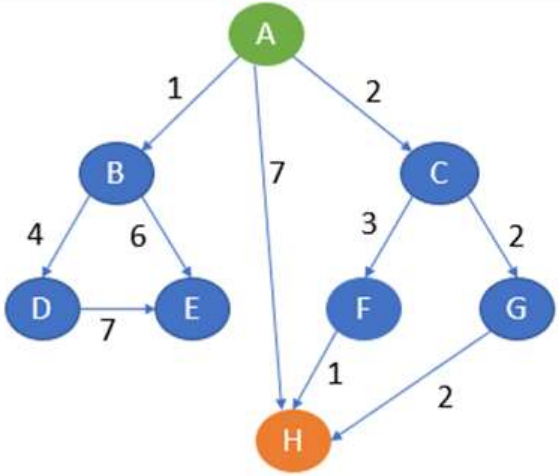
Disadvantages of A* search algorithm

- The A* search is based on heuristics and cost. It may not produce the shortest path.
- The usage of memory is more as it keeps all the nodes in the memory.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

b)A* Search Algorithm



NODES	HEURISTICS
A	5
B	3
C	4
D	2
E	6
F	3
G	1
H	0

Let A be the start node and H be the goal node.

First, the algorithm will start with A. From A, it can go to B, C, H.

Note the point that A* search uses the sum of path cost and heuristics value to determine the path.

Here, from A to B, the sum of cost and heuristics is 1 + 3 = 4.

From A to C, it is 2 + 4 = 6.

From A to H, it is 7 + 0 = 7.

Here, the lowest cost is 4 and the path A to B is chosen. The other paths will be on hold.

Now, from B, it can go to D or E.

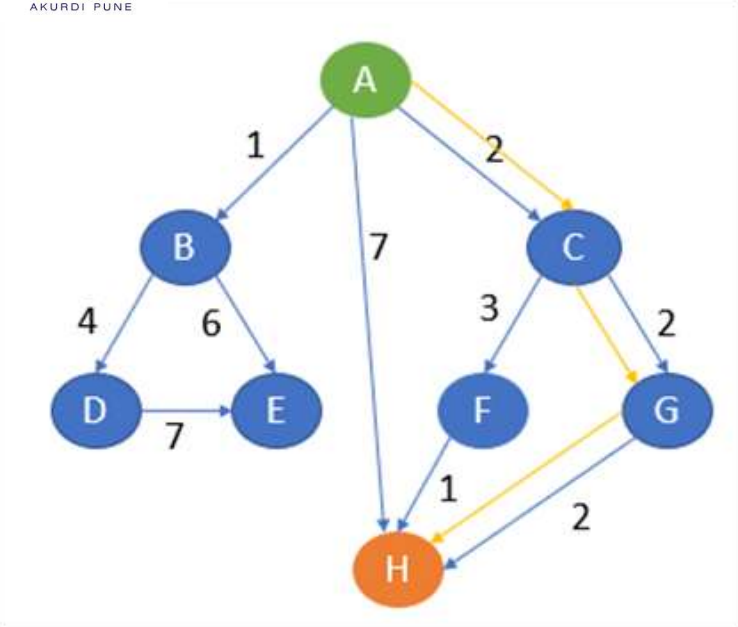
From A to B to D, the cost is 1 + 4 + 2 = 7.

From A to B to E, it is 1 + 6 + 6 = 13.

The lowest cost is 7. Path A to B to D is chosen and compared with other paths which are on hold.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms



The path of traversal is

A —-> C —-> G —-> H

Here, path A to C is of less cost. That is 6.

Hence, A to C is chosen and other paths are kept on hold.

From C, it can now go to F or G.

From A to C to F, the cost is 2 + 3 + 3 = 8.

From A to C to G, the cost is 2 + 2 + 1 = 5.

The lowest cost is 5 which is also lesser than other paths which are on hold. Hence, path A to G is chosen.

From G, it can go to H whose cost is 2 + 2 + 2 + 0 = 6.

Here, 6 is lesser than other paths cost which is on hold.

Also, H is our goal state. The algorithm will terminate here.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Search Algorithms

Comparison of uninformed and informed search algorithms

- Uninformed search is also known as blind search whereas informed search is also called heuristics search.
- Uninformed search does not require much information.
- Informed search requires domain-specific details.
- Compared to uninformed search, informed search strategies are more efficient and the time complexity of uninformed search strategies is more.
- Informed search handles the problem better than blind search.
- Search algorithms are used in games, stored databases, virtual search spaces, quantum computers, and so on.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

Heuristic Search

- **Heuristic search** is defined as a procedure of search that endeavors to upgrade an issue by iteratively improving the arrangement dependent on a given heuristic capacity or a cost measure.
- This technique doesn't generally ensure to locate an ideal or the best arrangement; however, it may rather locate a decent or worthy arrangement inside a sensible measure of time and memory space.
- This is a sort of an alternate route as we regularly exchange one of optimality, culmination, exactness, or accuracy for speed.
- It may be a function which is employed in Informed Search, and it finds the foremost promising path. It takes the present state of the agent as its input and produces the estimation of how close agent is from the goal.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

Heuristic Search

- A Heuristic (or a heuristic capacity) investigates search calculations. At each stretching step, it assesses the accessible data and settles on a choice on which branch to follow. It does as such by positioning other options. The Heuristic is any gadget that is frequently successful yet won't ensure work for each situation.
- We need heuristics in order to create, in a sensible measure of time, an answer that is sufficient for the issue being referred to. It doesn't need to be the best-an estimated arrangement will do since this is sufficiently quick. Most issues are exponential.
- A heuristic is a technique that is used to solve a problem faster than the classic methods. These techniques are used to find the approximate solution of a problem when classical methods do not. Heuristics are said to be the problem-solving techniques that result in practical and quick solutions.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

Heuristic Search

- Heuristics are strategies that are derived from past experience with similar problems. Heuristics use practical methods and shortcuts used to produce the solutions that may or may not be optimal, but those solutions are sufficient in a given limited time frame.

Why we need Heuristic Search?

- Heuristics are used in situations in which there is the requirement of a short-term solution. On facing complex situations with limited resources and time, Heuristics can help the companies to make quick decisions by shortcuts and approximated calculations. Most of the heuristic methods involve mental shortcuts to make decisions on past experiences.
- Based on context, there can be different heuristic methods that correlate with the problem's scope. The most common heuristic methods are - trial and error, guesswork, the process of elimination, historical data analysis.

Heuristic Search Strategies

Heuristic Search Strategies

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

1) Hill Climbing

- It is a technique for optimizing the mathematical problems
- Hill Climbing is widely used when a good heuristic is available.
- It is a local search algorithm that continuously moves in the direction of increasing elevation/value to find the mountain's peak or the best solution to the problem.
- It terminates when it reaches a peak value where no neighbor has a higher value.
- Traveling-salesman Problem is one of the widely discussed examples of the Hill climbing algorithm, in which we need to minimize the distance traveled by the salesman.
- It is also called greedy local search as it only looks to its good immediate neighbor state and not beyond that.
- The steps of a simple hill-climbing algorithm are listed below

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

1) Hill Climbing

Step 1: Evaluate the initial state. If it is the goal state, then return success and Stop.

Step 2: Loop Until a solution is found or there is no new operator left to apply.

Step 3: Select and apply an operator to the current state.

Step 4: Check new state:

If it is a goal state, then return to success and quit.

Else if it is better than the current state, then assign a new state as a current state.

Else if not better than the current state, then return to step 2.

Step 5: Exit

Heuristic Search Strategies

1) Hill Climbing

Blocks World

Local heuristic:

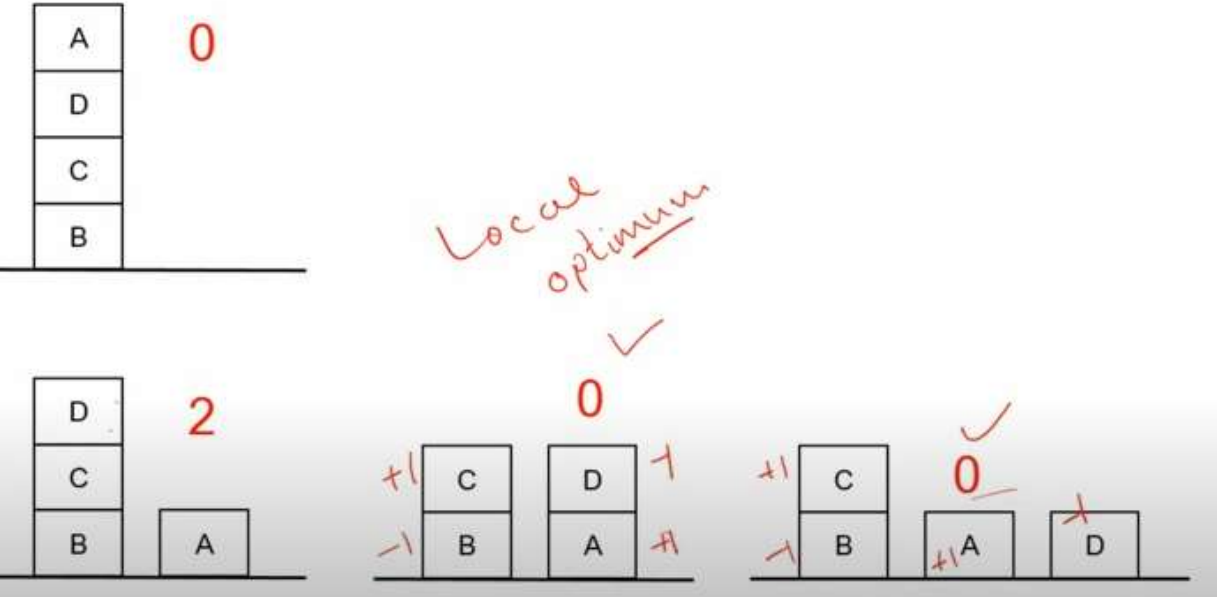
+1 for each block that is resting on the thing it is supposed to be resting on.

–1 for each block that is resting on a wrong thing.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

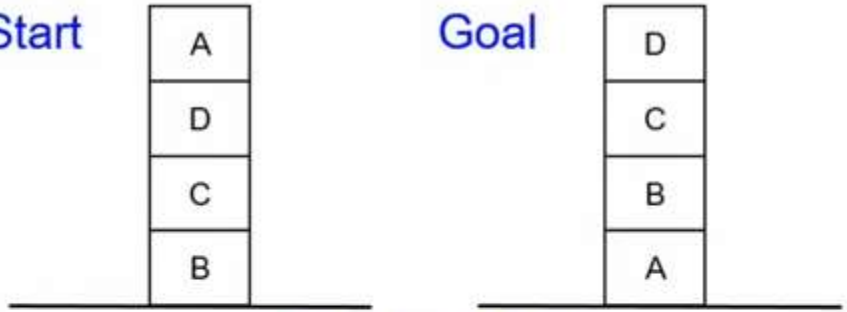
Heuristic Search Strategies

1) **Hill Climbing by local search**



Heuristic Search Strategies

1) Hill Climbing by Global search



Blocks World

Global heuristic:

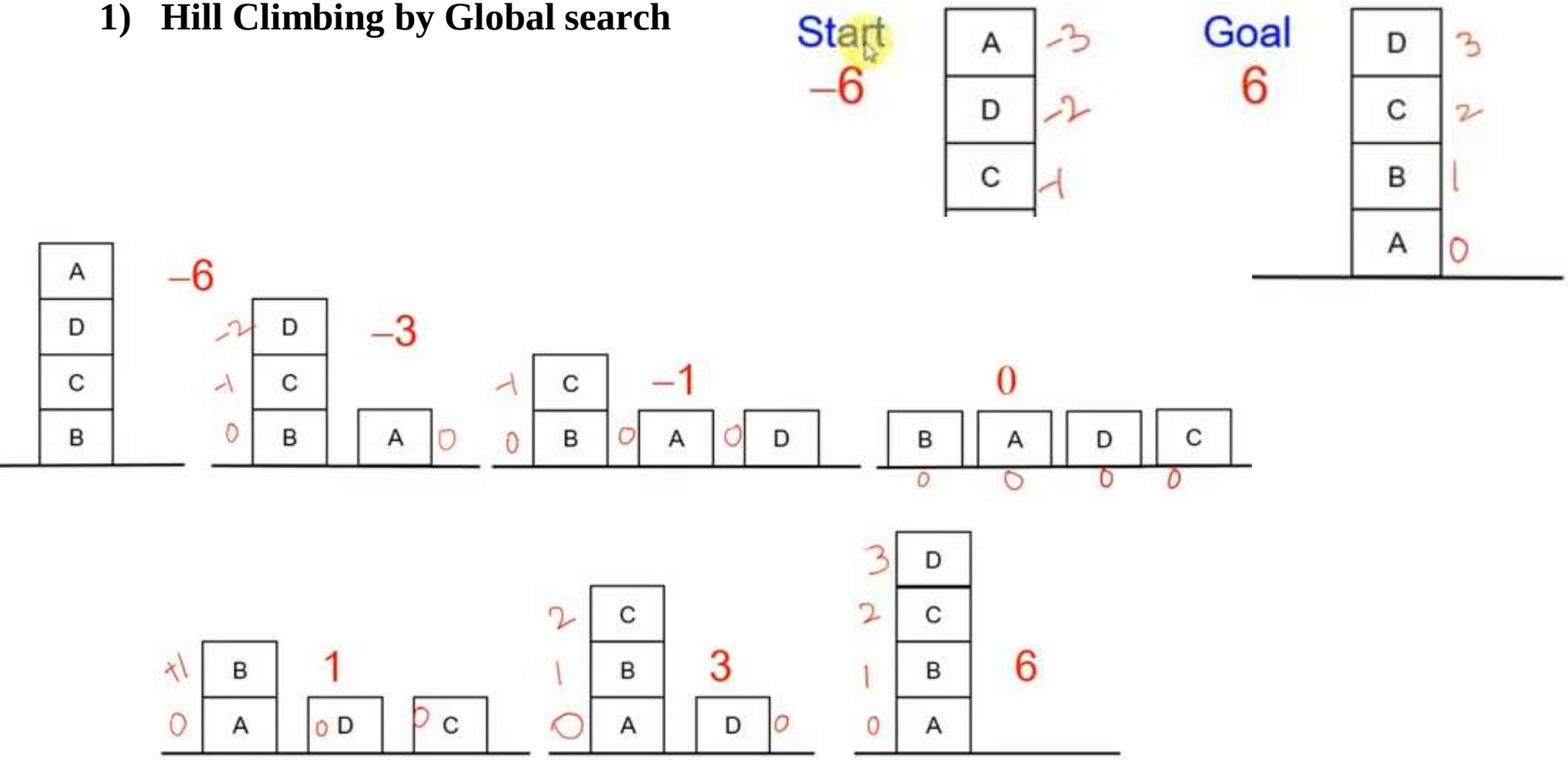
For each block that has the correct support structure: +1 to every block in the support structure.

For each block that has a wrong support structure: –1 to every block in the support structure.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

1) Hill Climbing by Global search



D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

2)Best First Search

- This algorithm always chooses the path which appears best at that moment.
- It is the combination of depth-first search and breadth-first search algorithms.
- It lets us to take the benefit of both algorithms.
- It uses the heuristic function and search.
- With the help of the best-first search, at each step, we can choose the most promising node.

Best first search algorithm:

Step 1: Place the starting node into the OPEN list.

Step 2: If the OPEN list is empty, Stop and return failure.

Heuristic Search Strategies

2)Best First Search

Step 3: Remove the node n from the OPEN list, which has the lowest value of h(n), and places it in the CLOSED list.

Step 4: Expand the node n, and generate the successors of node n.

Step 5: Check each successor of node n, and find whether any node is a goal node or not. If any successor node is the goal node, then return success and stop the search, else continue to next step.

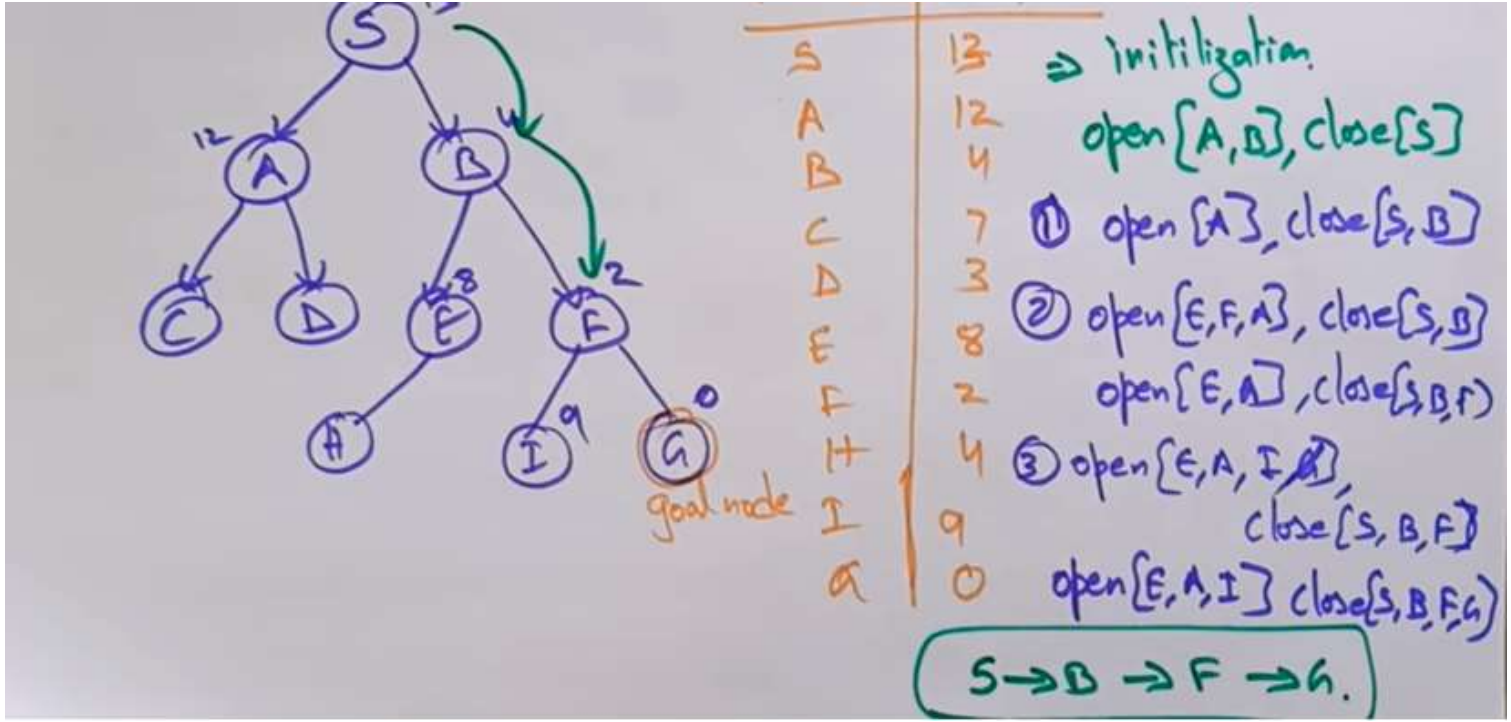
Step 6: For each successor node, the algorithm checks for evaluation function f(n) and then check if the node has been in either OPEN or CLOSED list. If the node has not been in both lists, then add it to the OPEN list.

Step 7: Return to Step 2.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

2)Best First Search



Heuristic Search Strategies

3)A* Search

- A* search is the most commonly known form of best-first search.
- It uses the heuristic function h(n) and cost to reach the node n from the start state g(n).
- It has combined features of UCS and greedy best-first search, by which it solve the problem efficiently.
- It finds the shortest path through the search space using the heuristic function.
- This search algorithm expands fewer search trees and gives optimal results faster.

Algorithm of A* search:

Step 1: Place the starting node in the OPEN list.

Step 2: Check if the OPEN list is empty or not. If the list is empty, then return failure and stops.

Heuristic Search Strategies

3)A* Search

Algorithm of A* search:

Step 3: Select the node from the OPEN list which has the smallest value of the evaluation function (g+h). If node n is the goal node, then return success and stop, otherwise.

Step 4: Expand node n and generate all of its successors, and put n into the closed list. For each successor n', check whether n' is already in the OPEN or CLOSED list. If not, then compute the evaluation function for n' and place it into the Open list.

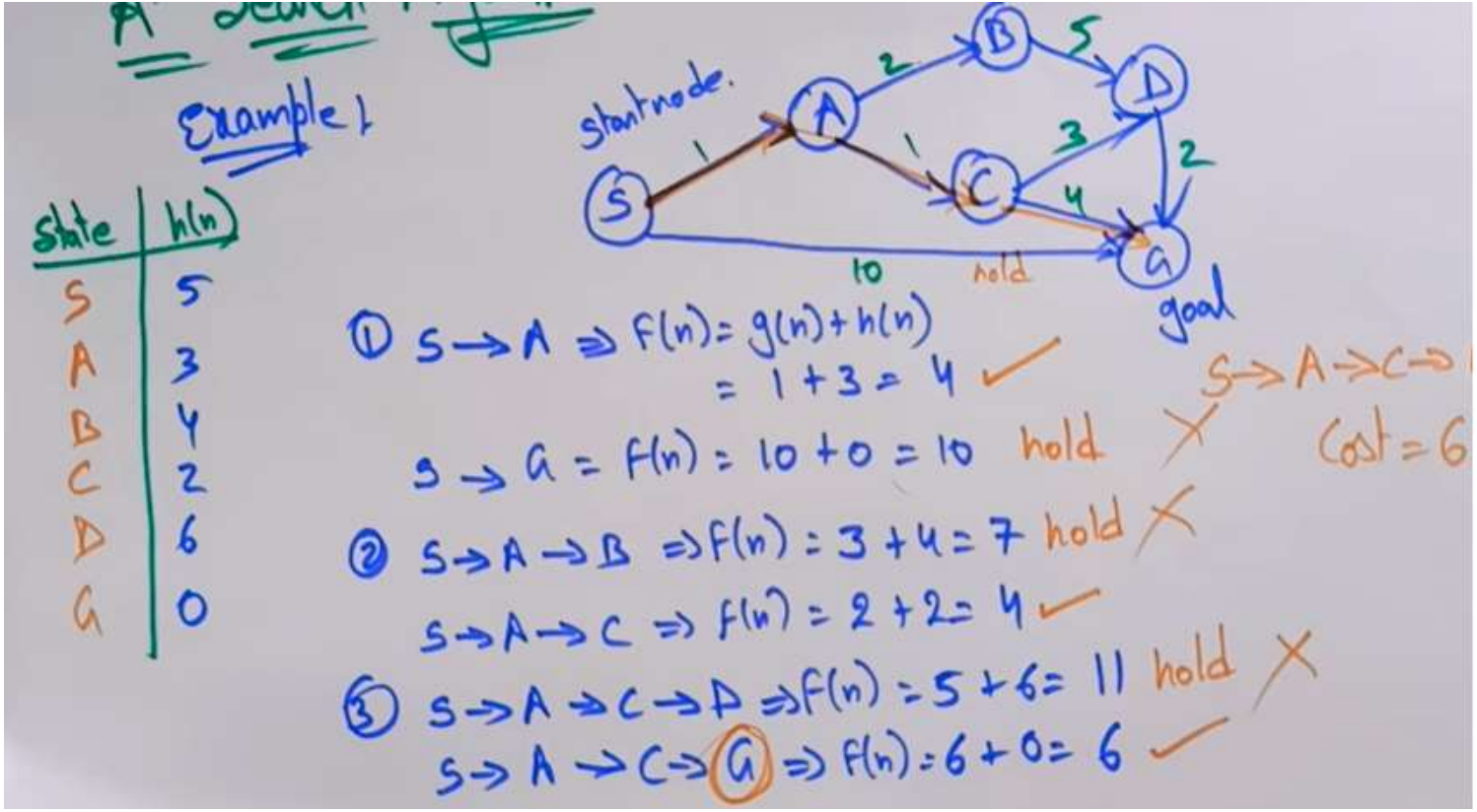
Step 5: Else, if node n' is already in OPEN and CLOSED, then it should be attached to the back pointer which reflects the lowest g(n') value.

Step 6: Return to Step 2.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

3)A* Search



Heuristic Search Strategies

4)Simulated Annealing

- Simulated annealing is a technique used in AI to find solutions to optimization problems.
- It is based on the idea of annealing in metallurgy, where a metal is heated and then cooled slowly in order to reduce its brittleness.
- In the same way, simulated annealing can be used to find solutions to optimization problems by slowly changing the values of the variables in the problem until a solution is found.
- The advantage of simulated annealing over other optimization methods is that it is less likely to get stuck in a local minimum, where the solution is not the best possible but is good enough.
- This is because simulated annealing allows for small changes to be made to the solution, which means that it can escape from local minima and find the global optimum.
- Simulated annealing is not a guaranteed method of finding the best solution to an optimization problem, but it is a powerful tool that can be used to find good solutions in many cases.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

4)Simulated Annealing

- The basic idea behind simulated annealing is to start with a random solution and then slowly change the values of the variables in the solution.
- The changes are made in such a way that the solution always remains close to the current optimum.
- The goal is to find the global optimum by making small changes to the solution.
- Simulated annealing has been used to solve a variety of optimization problems, including the travelling salesman problem, the knapsack problem, and the satisfiability problem.
- It has also been used in image recognition, machine learning, and other areas of AI.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

5)Constraint Search Problem(CSP)

- Consider a Sudoku game with some numbers filed initially in some squares.
- You are expected to fill the empty squares with numbers ranging from 1 to 9 in such a way that no row, column or a block has a number repeating itself.
- This is a very basic constraint satisfaction problem.
- You are supposed to solve a problem keeping in mind some constraints.
- The remaining squares that are to be filled are known as variables, and the range of numbers (1-9) that can fill them is known as a domain.
- Variables take on values from the domain.
- The conditions governing how a variable will choose its domain are known as constraints.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

5)Constraint Search Problem(CSP)

- A **constraint satisfaction problem (CSP)** is a problem that requires its solution within some limitations or conditions also known as constraints. It consists of the following:
- A finite set of **variables** which stores the solution (V = {V1, V2, V3,....., Vn})
- A set of **discrete** values known as **domain** from which the solution is picked (D = {D1, D2, D3,.....,Dn})
- A finite set of **constraints** (C = {C1, C2, C3,....., Cn}) where each can involve any number of variables C_k: power set(D)→{true, false}, k=1….n
- The elements in the domain can be both continuous and discrete but in AI, we generally only deal with discrete values. Also, note that all these sets should be finite except for the domain set.
- Each variable in the variable set can have different domains.
- For example, consider the Sudoku problem again. Suppose that a row, column and block already have 3, 5 and 7 filled in. Then the domain for all the variables in that row, column and block will be {1, 2, 4, 6, 8, 9}.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

5)Constraint Search Problem(CSP)

Popular Problems with CSP

The following problems are some of the popular problems that can be solved using CSP:

1. Crypt Arithmetic (Coding alphabets to numbers.)
2. n-Queen (In an n-queen problem, n queens should be placed in an nXn matrix such that no queen shares the same row, column or diagonal.)
3. Map Coloring (coloring different regions of map, ensuring no adjacent regions have the same color)
4. Crossword (everyday puzzles appearing in newspapers)
5. Sudoku (a number grid)
6. Latin Square Problem

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

CSP as search problem

- The domains and variables together determine a set of all possible assignments (solutions) that can be complete or partial.
- Finding the one(s) that satisfy all the constraints is a search problem like finding the shortest path between two nodes in a graph.
- The CSP search graph's nodes represent the assignments, and an edge from node to vertex corresponds to a change in a single variable.
- The start node is the empty solution, with all variables unassigned.
- Each time when we cross an edge, we change the value of exactly one variable.
- The search stops when we find the complete assignment that satisfies all the constraints.
- The constraint satisfaction check corresponds to the goal test in the general search

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Heuristic Search Strategies

CSP as search problem

- We can visualize the CSP and the structure of its solutions as a constraint graph**.**
- If all the constraints are binary, the nodes in the graph represent the CSP variables, and the edges denote the constraints acting upon them.
- However, if the constraints involve more than two variables (in which case, we call them global), we create constraint hyper-graphs.
- They contain two types of nodes: the regular nodes that represent variables and hyper-nodes that represent constraints.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Local Search and Optimization problem

The informed and uninformed search expands the nodes systematically in two ways:

- keeping different paths in the memory and
- selecting the best suitable path,

❖ Which leads to a solution state required to reach the goal node. But beyond these “classical search algorithms," we have some “local search algorithms” where the path cost does not matters, and only focus on solution-state needed to reach the goal node. A local search algorithm completes its task by traversing on a single current node rather than multiple paths and following the neighbors of that node generally.

❖ Although local search algorithms are not systematic, still they have the following two advantages:

- Local search algorithms use a very little or constant amount of memory as they operate only on a single path.
- Most often, they find a reasonable solution in large or infinite state spaces where the classical or systematic algorithms do not work.

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Local Search and Optimization problem

Does the local search algorithm work for a pure optimized problem?

- Yes, the local search algorithm works for pure optimized problems.
- A pure optimization problem is one where all the nodes can give a solution.
- But the target is to find the best state out of all according to the objective function.
- Unfortunately, the pure optimization problem fails to find high-quality solutions to reach the goal state from the current state.
- Note: An objective function is a function whose value is either minimized or maximized in different contexts of the optimization problems.
- In the case of search algorithms, an objective function can be the path cost for reaching the goal node, etc.

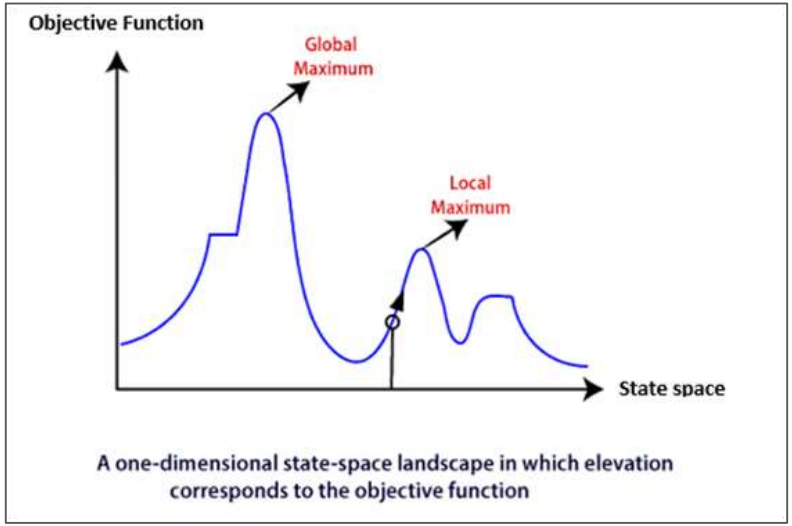
D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Local Search and Optimization problem

Working of a Local search algorithm:

Let's understand the working of a local search algorithm with the help of an example:Consider the below state-space landscape having both:

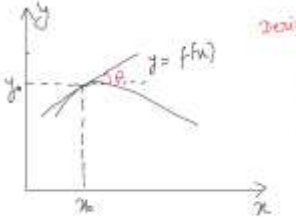
- Location: It is defined by the state.
- Elevation: It is defined by the value of the objective function or heuristic cost function
- In mathematics, the global maximum is the largest value in the whole function, while the local maximum is the largest value in a subset of the function.



A one-dimensional state-space landscape in which elevation corresponds to the objective function

Local Search and Optimization problem

D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE



Derivative of y w.r.t x at x_0

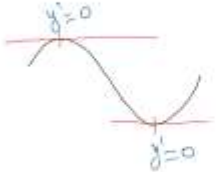
$$\left.\frac{dy}{dx}\right|_{x=x_0} = \tan\theta$$

The derivative of a function at a point x_0 is equal to the slope of the tangent to the curve at that point.

Critical Points

$\tan\theta = 0$

so $\frac{dy}{dx} = 0$ ⇒ The tangent is parallel to the horizontal axis.

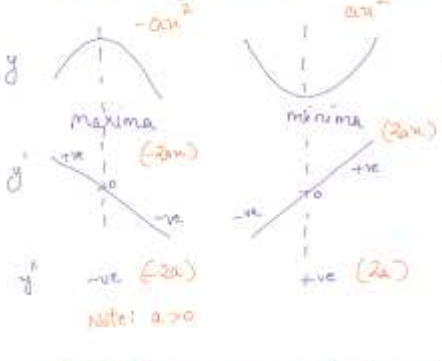


Regardless of whether the point is a local or global maximum or minimum $\left.\frac{dy}{dx}\right|_{x=x_0} = 0$ at these points.

And such points are called critical points.

[y' is another notation for $\frac{dy}{dx}$]

Differentiating maxima V/s minima



Steps:
1) Find critical points
2) First derivative test
if derivative changes from +ve to -ve at critical point ⇒ maxima
if derivative changes from -ve to +ve ⇒ minima

or 2nd derivative test
if $y'' < 0$ ⇒ maxima
$y'' > 0$ ⇒ minima

Tip: If there are multiple minima, calculate y values and check which is lowest that point is global minimum and rest are local minima. 1st test for maximum.

Example

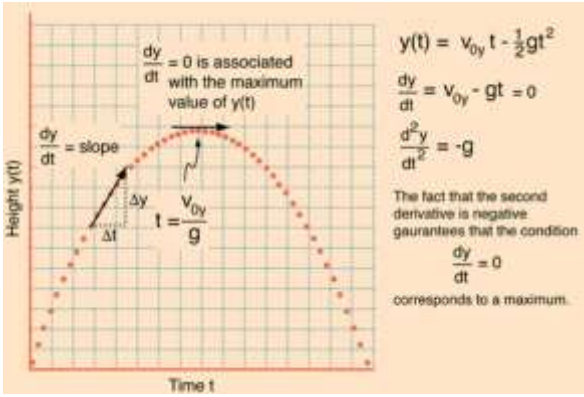
$y = x^4 + x^3 - 6x^2$

$y' = 4x^3 + 3x^2 - 12x$ (1st derivative)

$y'' = 12x^2 + 6x - 12$ (2nd derivative)

$y' = 0 \Rightarrow x = 0, -2.15, 1.4$ (Critical points)

x	x_0 = 0	x_1 = -2.15	x_2 = 1.4
y''	-12	30.57	19.92
y	0	-16.3	-5.17
	Local Maximum	Global minima	local minima



D Y PATIL INTERNATIONAL UNIVERSITY
AKURDI PUNE

Local Search and Optimization problem

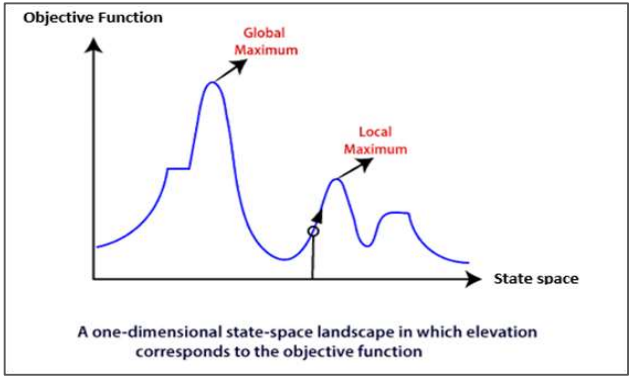
Working of a Local search algorithm:

The local search algorithm explores the above landscape by finding the following two points:

- Global Minimum: If the elevation corresponds to the cost, then the task is to find the lowest valley, which is known as Global Minimum.
- Global Maxima: If the elevation corresponds to an objective function, then it finds the highest peak which is called as Global Maxima. It is the highest point in the valley.

Below are some different types of local searches:

- Hill-climbing Search
- Simulated Annealing
- Local Beam Search



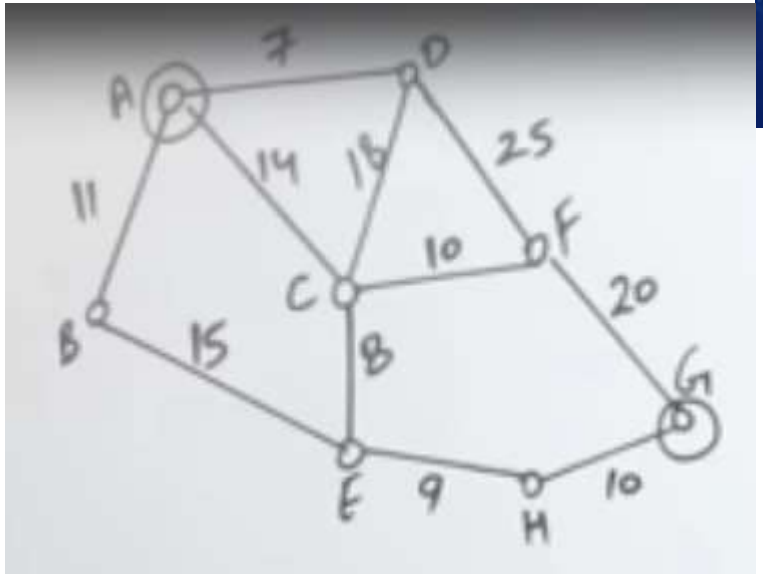
A one-dimensional state-space landscape in which elevation corresponds to the objective function

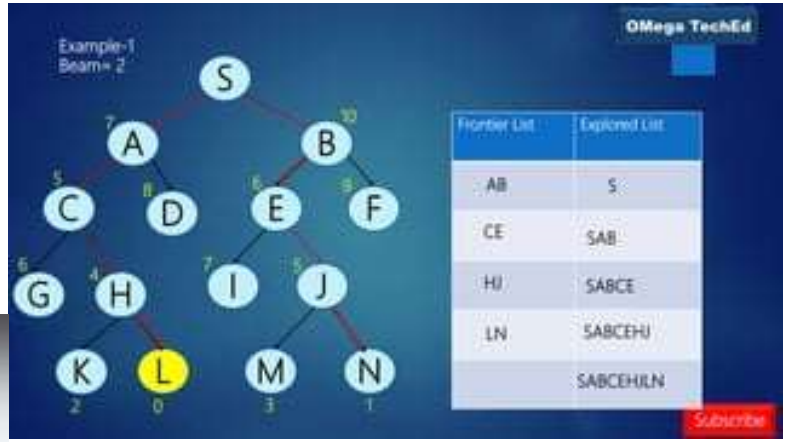
Local Search and Optimization problem

- **Local Beam Search**

It take care of space complexity (Constant)

Beam width=n





Thank you!