USN | | | | | | | | | | |

CMRIT — CMR Institute of Technology, Bengaluru. Accredited with A+ Grade by NAAC

Internal Assessment Test 3 Scheme & Solutions– January 2022

Sub:	Artificial Intelligence & Machine Learning					Sub Code:	18CS71	Branch:	ISE
Date:	27/1/2022	Duration:	90 min's	Max Marks:	50	Sem/Sec:	VII A, B & C		OBE

	Scheme	MARKS	CO	RBT
1.	• Defining the problem characteristics - 2 M • Explanation - 8 M	10M	CO1	L1
2	• Defining the Production rules - 5 M • Solution - 5 M	10M	CO1	L3
3	• Algorithm for BFS and DFS - 8M • Comparison - 2 M	10M	CO1	L2
4	• Steepest ascent hill climbing -1 M • Drawbacks - 2M • Example - 7M	10M	CO1	L3
5	• AND_OR graph definition – 1M • AO* algorithm - 4M • Example - 5M	10M	CO1	L3
6	• Solution to the problem - 10M	10M	CO1	L3
7.a	• Defining Q-Learning with example - 5M	5M	CO1	L2
7.b	• Solution to the problem - 5M	5M	CO1	L3

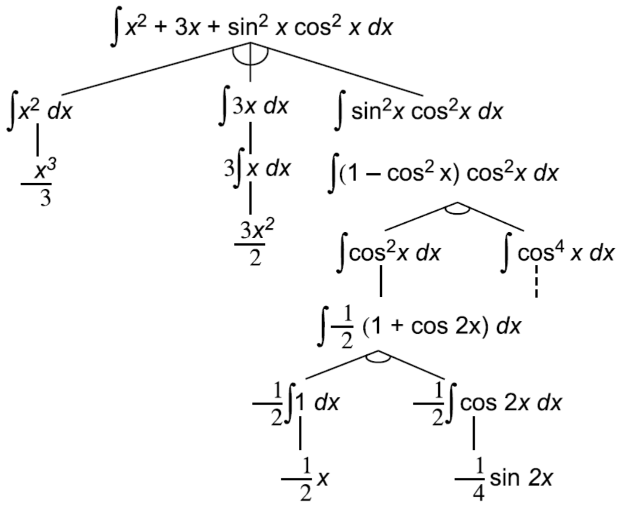
Solutions

1	List and explain the problems characteristics which must be analyzed before deciding on a proper heuristic search	10	CO1	L1

Problem characteristics

- Is the problem decomposable into a set of (nearly) independent smaller or easier sub problems?
- Can solution steps be ignored or at least undone if they prove unwise?
- Is the problem's universe predictable?
- Is a good solution to the problem obvious without comparison to all other possible solutions?
- Is the desired solution a state of the world or a path to a state?
- Is a large amount of knowledge absolutely required to solve the problem, or is knowledge important only to constrain the search?
- Can a computer that is simply given the problem return the solution, or will the solution of the problem require interaction between the computer and a person?

Is the problem decomposable?



Can solution steps be ignored or at least undone if they prove unwise?

Ignorable: Example theorem proving

Recoverable: Example: the 8 puzzle

Irrecoverable: Example chess

Is the universe predictable?

Do we clearly know the outcome of a move?

Is a good solution absolute or relative?

Using a set of rules to arrive at answering a question – solution can be absolute

Travelling sales person – Solution is relative

Is the solution a state or path?

In certain cases, final solution is sufficient. (answering question based on facts)

In the water jug problem, path to solution is important

What is the role of knowledge?

Knowledge may be required to limit the search space.

Does the task require interaction with a person?

2 A water jug problem states "You are given two jugs, a 4-gallon one and a 3-gallon one, a pump which has unlimited water which you can use to fill the jug, and the ground on which water may be poured. Neither jug has any measuring markings on it. How can you get exactly 2 gallon of water in the 4-gallon jug?"

10 CO1 L3

i)Write the production rules for the above problem

ii) Write any one solution for the above problem

- The state space for this problem can be described as a set of ordered pairs of integers (x,y) where,
  - x represents the number of gallons of water in the 4-gallon jug
  - y represents the number of gallons of water in the 4-gallon jug.
  - Values of x can be 0,1,2,3, or 4
  - Values of y can be 0,1,2, or 3.
- Start state is (0,0)
- Goal state is (2, n) for any of value of n

Production Rules:

No.	Rule		Result	Description
1	(x, y) if $x < 4$	$\rightarrow$	$(4, y)$	Fill the 4-gallon jug
2	(x, y) if $y < 3$	$\rightarrow$	$(x, 3)$	Fill the 3-gallon jug
3	(x, y) if $x > 0$	$\rightarrow$	$(x - d, y)$	Pour some water out of the 4-gallon jug
4	(x, y) if $y > 0$	$\rightarrow$	$(x, y - d)$	Pour some water out of the 3-gallon jug
5	$(x, y\}$ if $x > 0$	$\rightarrow$	$(0, y)$	Empty the 4-gallon jug on the ground
6	(x, y) if $y > 0$	$\rightarrow$	$(x, 0)$	Empty the 3-gallon jug on the ground
7	(x, y) if $x + y \geq 4$ and $y > 0$	$\rightarrow$	$(4, y - (4 - x))$	Pour water from the 3-gallon jug into the 4-gallon jug until the 4-gallon jug is full
8	(x, y) if $x + y \geq 3$ and $x > 0$	$\rightarrow$	$(x - (3 - y), 3)$	Pour water from the 4-gallon jug into the 3-gallon jug until the 3-gallon jug is full
9	(x, y) if $x + y \leq 4$ and $y > 0$	$\rightarrow$	$(x + y, 0)$	Pour all the water from the 3-gallon jug into the 4-gallon jug
10	(x, y) if $x + y \leq 3$ and $x > 0$	$\rightarrow$	$(0, x + y)$	Pour all the water from the 4-gallon jug into the 3-gallon jug
11	$(0, 2)$	$\rightarrow$	$(2, 0)$	Pour the 2 gallons from the 3-gallon jug into the 4-gallon jug
12	$(2, y)$	$\rightarrow$	$(0, y)$	Empty the 2 gallons in the 4-gallon jug on the ground

Solution

Gallons in the 4-Gallon Jug	Gallons in the 3-Gallon Jug	Rule Applied
0	0	
		2
0	3	
		9
3	0	
		2
3	3	
		7
4	2	
		5 or 12
0	2	
		9 or 11
2	0	

Q. No.	Question	Marks	CO	Level
3	Give the algorithms for DFS and BFS? Compare both with suitable example.	10	CO1	L2

Requirements of a good control strategy:

1. A good control strategy causes motion (Change of state from initial to final)
2. A good control strategy must be systematic.

BFS

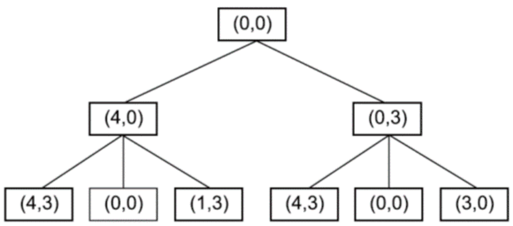
- 1. Create a variable called NODE-LIST and set it to initial state.
- 2. Until a goal state is found or NODE-LIST is empty

A. Remove the first element from the NODE-LIST and call it El.If NODE-LIST was empty quit.

B. For each way that each rule can match the state described in E do:

1. Apply the rule to generate a new state.
2. If the new state is goal state quit and return this state.
3. Otherwise, add the new state to the end of NODE-LIST.

Two Levels of a Breadth-First Search Tree



DFS:

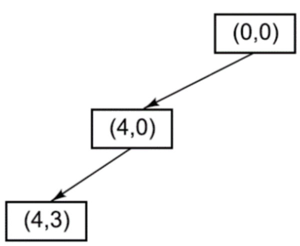
- 1. If the initial state is a goal state quit and return success.
- 2. Otherwise, do the following until success or failure is signaled

A. Generate a successor E of the initial state. If there are no more successor, signal failure.

B. Call DFS with E as the initial state.

C. If success is returned signal success, otherwise continue in this loop.

A Depth-First Search Tree



Advantages:

DFS:

- Requires less memory since only the nodes on the current path are stored.
- By chance, DFS may find a solution without examining much of the search space at all.

BFS:

- Will not get trapped exploring a blind alley.
- BFS is guaranteed to find a solution if one exists.
- Minimal solution is always found.

4	Explain steepest hill climbing search technique with an algorithm. Comment on its drawbacks and how to overcome these drawbacks. Also apply the same for 8-puzzle problem	10	CO1	L3

Steepest-Ascent Hill climbing : It first examines all the neighboring nodes and then selects the node closest to the solution state as next node.

Algorithm: Steepest-Ascent Hill climbing

Step 1: Evaluate the initial state. If it is goal state then exit else make the current state as initial state

Step 2: Repeat these steps until a solution is found or current state does not change

a. Let SUCC be a state such that any successor of the current state will be better than it;

b for each operator that applies to the current state do:

i. apply the new operator and create a new state

ii. Evaluate the new state

iii. if this state is goal state then quit else compare with SUCC

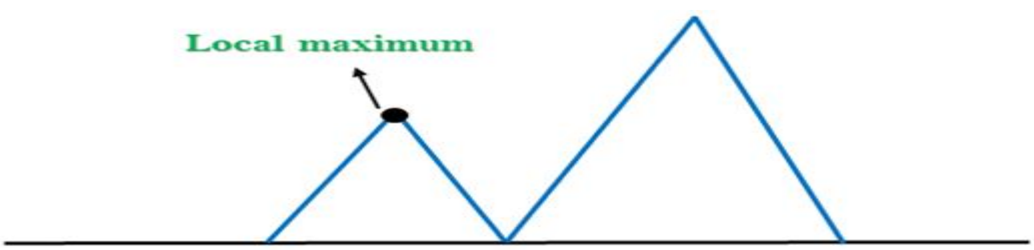
iv. if this state is better than SUCC, set this state as SUCC

c. if SUCC is better than current state set current state to SUCC

Drawbacks:

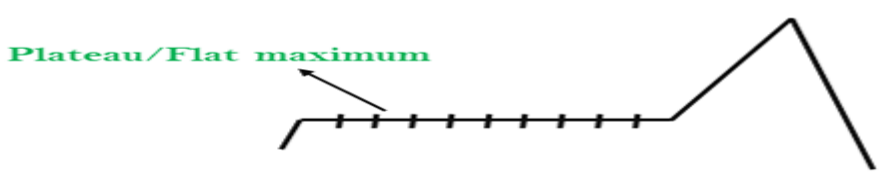
1. Local Maximum: A local maximum is a peak state in the landscape which is better than each of its neighboring states, but there is another state also present which is higher than the local maximum.

Solution: Backtracking technique can be a solution of the local maximum in state space landscape. Create a list of the promising path so that the algorithm can backtrack the search space and explore other paths as well.



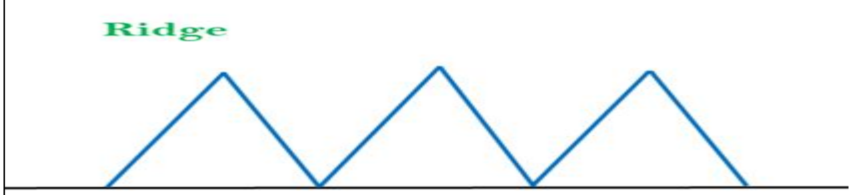
2. Plateau: A plateau is the flat area of the search space in which all the neighbor states of the current state contains the same value, because of this algorithm does not find any best direction to move. A hill-climbing search might be lost in the plateau area.

Solution: The solution for the plateau is to take big steps or very little steps while searching, to solve the problem. Randomly select a state which is far away from the current state so it is possible that the algorithm could find non-plateau region.



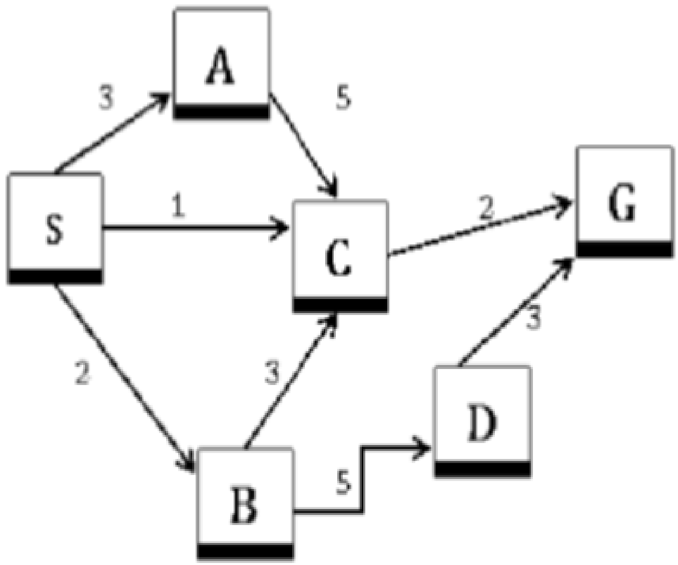
3. Ridges: A ridge is a special form of the local maximum. It has an area which is higher than its surrounding areas, but itself has a slope, and cannot be reached in a single move.

Solution: With the use of bidirectional search, or by moving in different directions, we can improve this problem.



5	What are AND-OR graphs, explain how problem reduction (AO*) algorithm uses AND-OR graphs for search procedure. Apply AO* algorithm for the below graph where S is the Initial State and G is the Goal state. The table below indicates h' values.	10	CO1	L3

State	h(n)
S	6
A	2
B	4
C	1
D	2
G	0



AND-OR Graphs:

- Useful for representing the solution of problems that can be solved by decomposing them into a set of smaller problems, all of which must be then solved.
- Some problems are best represented as achieving sub goals, some of which achieved simultaneously and independently (AND)

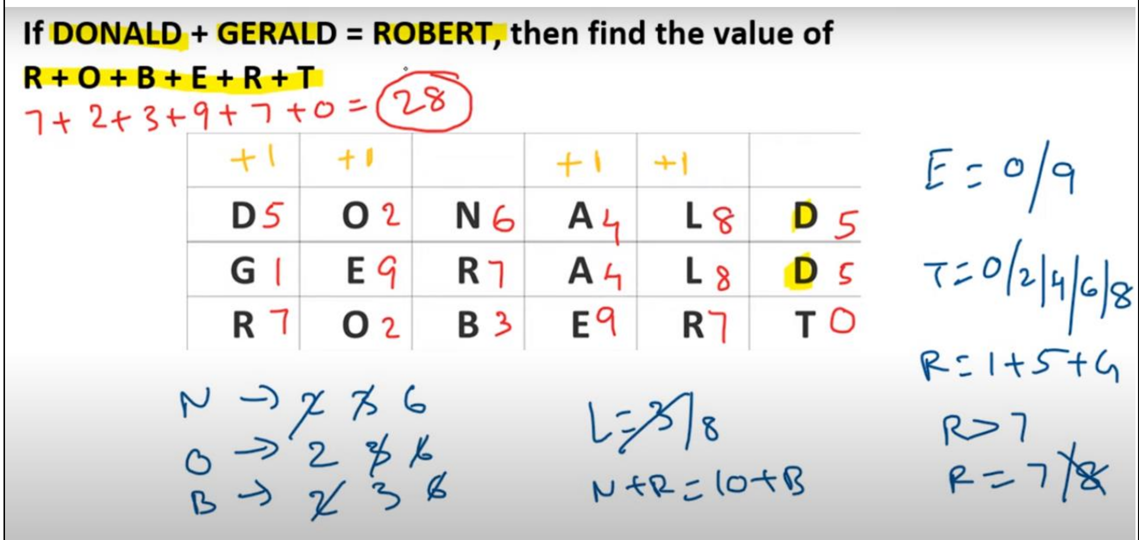
Up to now, only dealt with OR options

AO* Algorithm:

1. Initialise the graph to start node
2. Traverse the graph following the current path accumulating nodes that have not yet been expanded or solved
3. Pick any of these nodes and expand it and if it has no successors call this value *FUTILITY* otherwise calculate only *f'* for each of the successors.
4. If *f'* is 0 then mark the node as *SOLVED*
5. Change the value of *f'* for the newly created node and let f' reflect on its predecessors by back propagation.
6. Wherever possible use the most promising routes and if all descendants of a node is marked as *SOLVED* then mark the parent node as *SOLVED*.

If starting node is *SOLVED* or value greater than *FUTILITY*, stop, else repeat from 2.

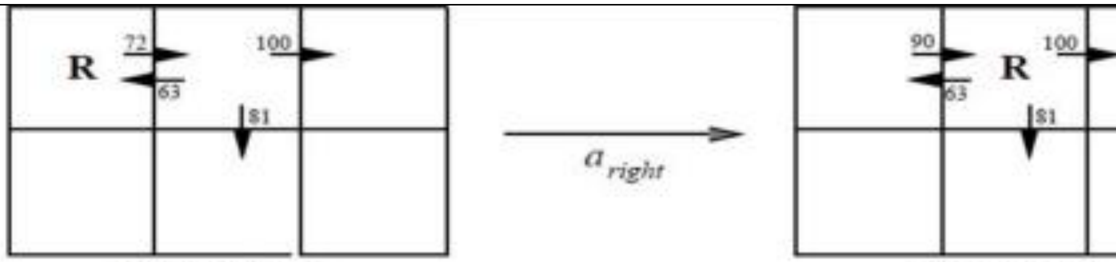
<table>
<tr><td></td><td>1. Current state is S
f(A)= 3+2 =5, f(B)= 2+4=6, f(C)=1+1=2
Since, C is having smaller distance compared to other nodes, we have chosen current state as C.
2.Current State is C
f(A)= 1+3=4, f(B) =1+4=5, f(G) = 2+1=3
Path = S -> C -> G</td><td></td><td></td><td></td></tr>
<tr><td>6</td><td>Solve the following crytparithmetic problem DONALD+GERALD=ROBERT


</td><td>10</td><td>CO1</td><td>L3</td></tr>
<tr><td>7a)</td><td>Explain the Q function and Q Learning Algorithm assuming deterministic rewards and actions with example.

Q-learning algorithm

For each s, a initialize the table entry $\hat{Q}(s,a)$ to zero
Oberserve the current state s
Do forever:
• Select an action a and execute it
• Receive immediate reward r
• Observe new state s'
• Update each table entry for $\hat{Q}(s,a)$ as follows
$$\hat{Q}(s,a) \leftarrow r + \gamma max_{a'} \hat{Q}(s',a')$$
• $s \leftarrow s'$

⇒ using this algorithm the agent's estimate $\hat{Q}$ converges to the actual Q, provided the system can be modeled as a deterministic Markov decision process, r is bounded, and actions are chosen so that every state-action pair is visited infinitely often</td><td>5</td><td>CO1</td><td>L2</td></tr>
</table>

	 Initial state: s_1 Next state: s_2 $\hat{Q}(s_1, a_{right}) \leftarrow r + \gamma \cdot \max_{a'} \hat{Q}(s_2, a')$ $\leftarrow 0 + 0.9 \cdot \max\{66, 81, 100\}$ $\leftarrow 90$ • each time the agent moves, Q Learning propagates $\hat{Q}$ estin *backwards* from the new state to the old			
7b)	Consider the following sentences: i)John likes all kinds of food ii) Apples are food iii) Chicken is food iv)Anything anyone eats and isn't killed by is food v)Bill eats peanuts and is still alive vi)She eats everything Bill eats. Translate these sentences into formulas in predicate logic. (a) Predicate Logic: 1. $\forall x: food(x) \rightarrow like(John)$ 2. $Food(Apples)$ 3. $Food(Chicken)$ 4. $\forall x, \forall y: Eat(x,y) \wedge \neg Killed(x) \rightarrow Food(y)$ 5. $Eats(Bill, Peanuts) \wedge Alive(Bill)$ 6. $\forall x: Eats(Bill, x) \rightarrow Eats(Swe, x)$	5	CO1	L3