Artificial Intelligence and Expert Systems

Laboratory

Assessment

Practical – 50%
Oral Based on Practical – 30%
Mini Project – 20%
Lab Credit -1

Assignments

Implement problem solver using A* algorithm: 8 puzzle, pacman, : (PyPlot Game project)

Implement game play with adversarial search using minimax algorithm: eg. Tic tac toe, Chess

Write a program to solve Constraint Satisfaction problem (Map coloring problem and crypt-arithmetic problem solver, Sudoku)

Implement a Unification algorithm and test it for various input cases. Demonstrate reasoning/inferencing using logic programming: Prolog / Swi-Prolog / JESS (Java Expert System Shell)

Assignments

6. Implement a local search algorithm or genetic algorithm eg chess, n-queens, travelling salesman problem

7. Implement a chatbot using techniques of Natural Language Processing

8. Implement a Neural network for a real life application

9. Mini Project

State Space Search Notations

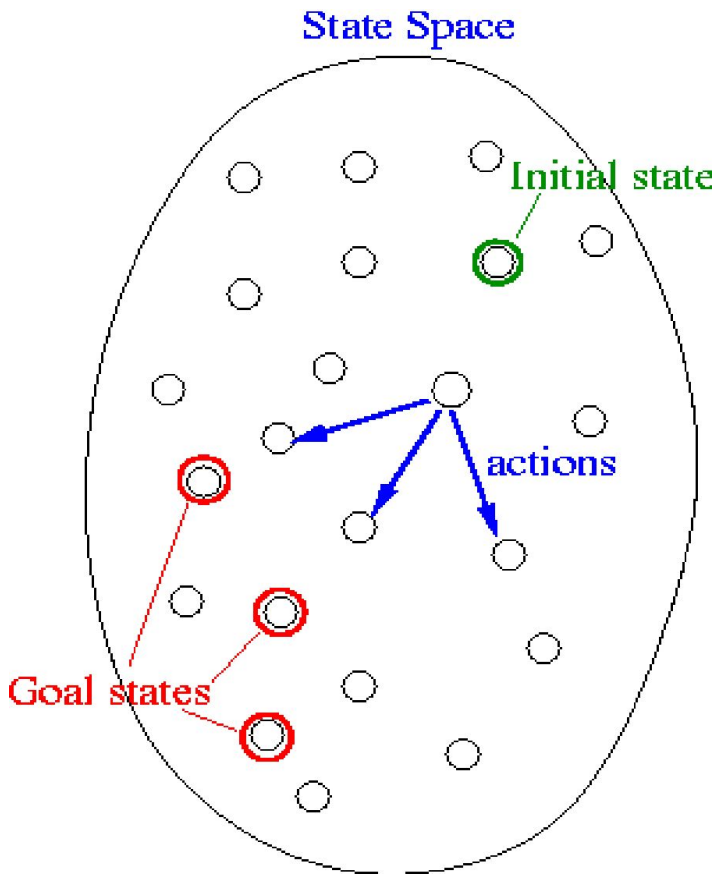
- Let us begin by introducing certain terms.
- An initial state is the description of the starting configuration of the agent
- An action or an operator takes the agent from one state to another state which is called a successor state. A state can have a number of successor states.
- A plan is a sequence of actions. The cost of a plan is referred to as the path cost. The path cost is a positive number, and a common path cost may be the sum of the costs of the steps in the path.
- *Search* is the process of considering various possible sequences of operators applied to the initial state and finding out a sequence which culminates in a goal state.

Search Problem

We are now ready to formally describe a search problem.

A search problem consists of the following:

- S: the full set of states
- s^0 : the initial state
- A:S→S is a set of operators
- G is the set of final states. Note that G ⊆ S



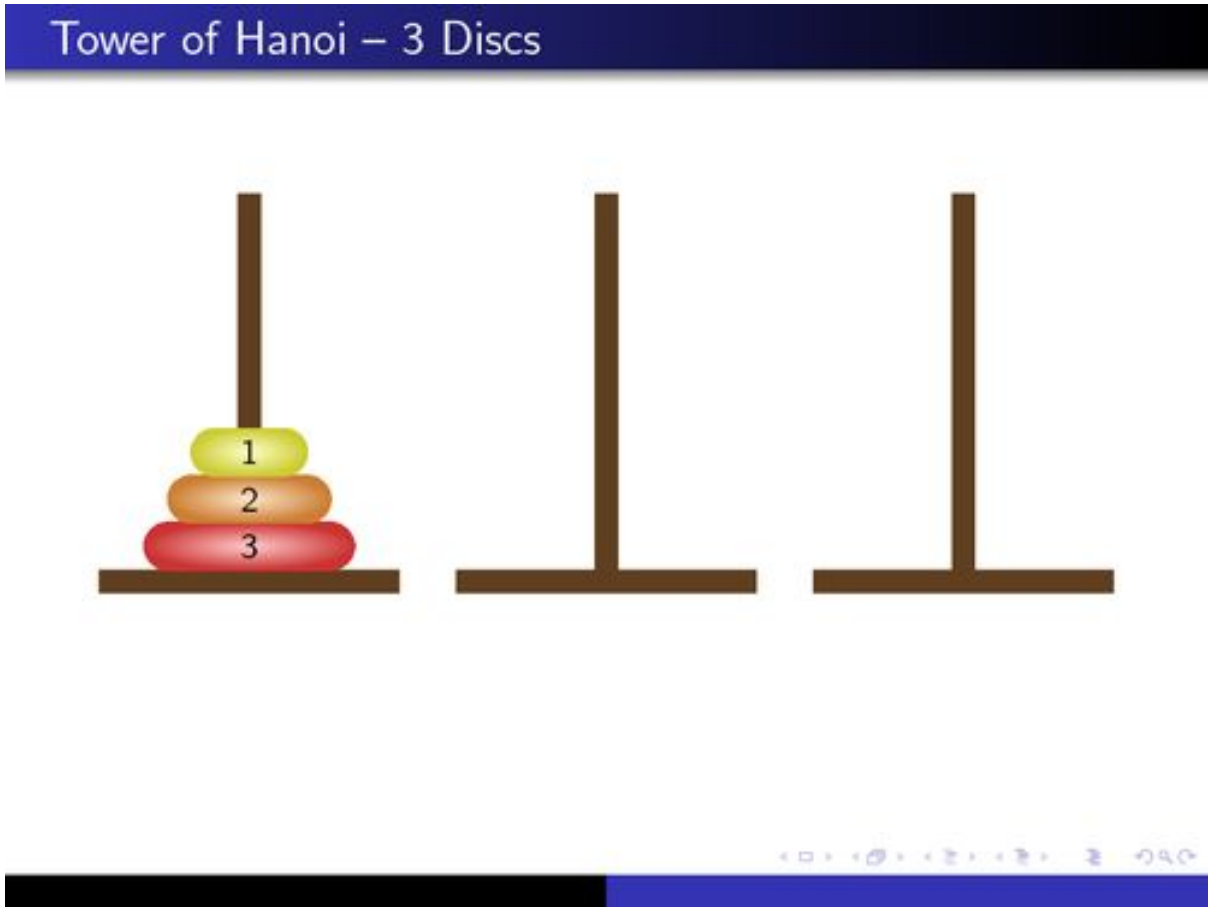
These are schematically depicted in above Figure

Tower of Hanoi Problem

Puzzle game where the objective is to move an entire stack of disks from the source position to another position. Three simple rules are

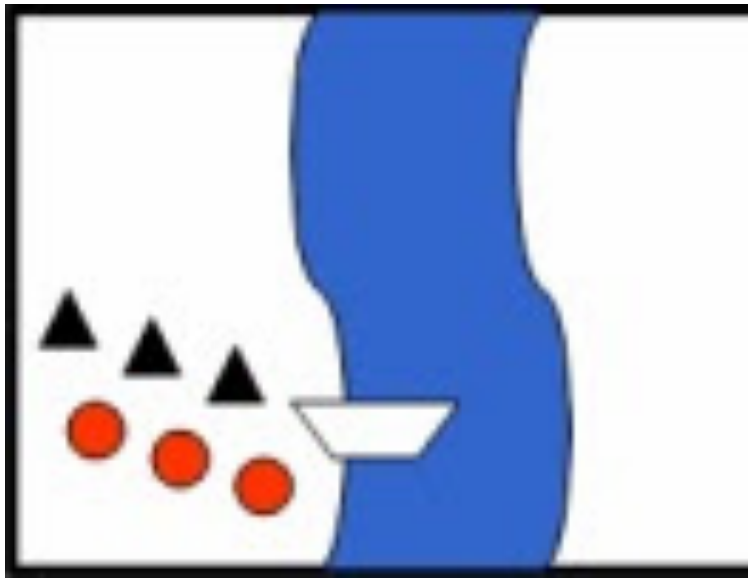
1. Only one disk can be moved at a time.
2. Each move consists of taking the upper disk from one of the stacks and placing it on top of another stack. In other words, a disk can only be moved if it is the uppermost disk on a stack.
3. No larger disk may be placed on top of a smaller disk.

- Suppose we have a stack of three disks. Our job is to move this stack from **source A** to **destination C**. How do we do this?



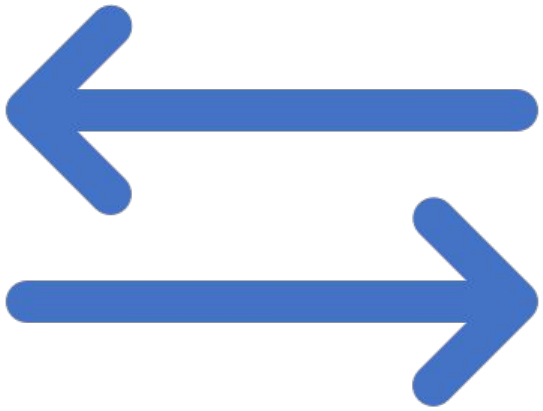
1. Move the first disk from A to C
2. Move the first disk from A to B
3. Move the first disk from C to B
4. Move the first disk from A to C
5. Move the first disk from B to A
6. Move the first disk from B to C
7. Move the first disk from A to C

The Missionaries and Cannibals: A classic AI problem



On one bank of a river are three missionaries and three cannibals. There is one boat available that can hold up to two people and that they would like to use to cross the river. If the cannibals ever outnumber the missionaries on either of the river's banks, the missionaries will get eaten.
How can the boat be used to safely carry all the missionaries and cannibals across the river?

State Space representation



An initial state is: (3, 3, LEFT, 0, 0)

Possible moves are:

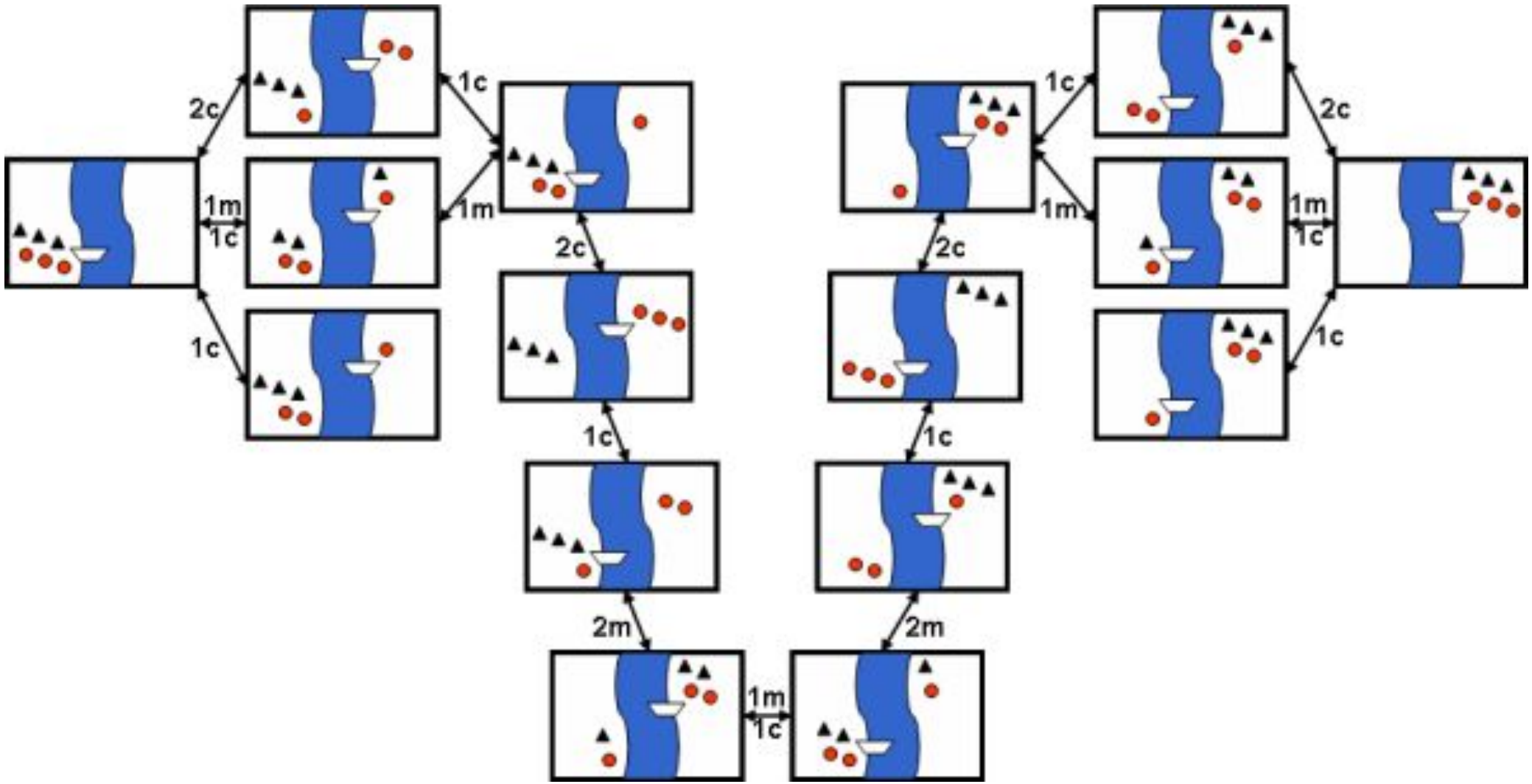
from (3, 3, LEFT, 0, 0) to (2, 2, RIGHT, 1, 1) ·

from (2, 2, RIGHT, 1, 1) to (2, 3, LEFT, 1, 0)

Goal state is: (0, 0, RIGHT, 3, 3)

Solution

Move-1m1c-lr
Move-1m1c-rl
Move-2c-lr
Move-2c-rl
Move-2m-lr
Move-2m-rl
Move-1c-lr
Move-1c-rl
Move-1m-lr
Move-1m-rl


2c
1c
1m
1c
1c
1m
2c
1c
2m
1m
1c
1c
1m
2c
2c
1c
2m
1m
1c
2c
1m
1c
1c

Water Jug Problem in Artificial Intelligence – State Space Search

you are given two jugs, a 4-liter one and a 3-liter one. Neither has any measuring markers on it. There is a pump that can be used to fill the jugs with water. How can you get exactly 2 liters of water into a 4-liter jug

Representation of water Jug Problem in terms of state-space search

- State: (x, y)
- where x represents the quantity of water in a 4-liter jug and y represents the quantity of water in a 3-liter jug.
- That is, x = 0, 1, 2, 3, or 4 y = 0, 1, 2, 3
- Start state: (0, 0).
- Goal state: (2, n) for any n.

Here need to start from the current state and end up in a goal state.

Production Rules for Water Jug Problem in Artificial Intelligence

1	(x, y) is X<4 ->(4, Y)	Fill the 4-liter jug
2	(x, y) if Y<3 -> (x, 3)	Fill the 3-liter jug
3	(x, y) if x>0 -> (x-d, d)	Pour some water out of the 4-liter jug.
4	(x, y) if Y>0 -> (d, y-d)	Pour some water out of the 3-liter jug.
5	(x, y) if x>0 -> (0, y)	Empty the 4-liter jug on the ground
6	(x, y) if y>0 -> (x,0)	Empty the 3-liter jug on the ground
7	(x, y) if X+Y >= 4 and y>0 -> (4, y-(4-x))	Pour water from the 3-liter jug into the 4-liter jug until the 4-liter jug is full
8	(x, y) if X+Y>=3 and x>0 -> (x-(3-y), 3))	Pour water from the 4-liter jug into the 3-liter jug until the 3-liter jug is full.
9	(x, y) if X+Y <=4 and y>0 -> (x+y, 0)	Pour all the water from the 3-liter jug into the 4-liter jug.
10	(x, y) if X+Y<=3 and x>0 -> (0, x+y)	Pour all the water from the 4-liter jug into the 3-liter jug.
11	(0, 2) -> (2, 0)	Pour the 2-liter water from the 3-liter jug into the 4-liter jug.
12	(2, Y) -> (0, y)	Empty the 2-liter in the 4-liter jug on the ground.

The solution to Water Jug Problem in Artificial Intelligence

- 1. Current state = (0, 0)
- 2. Loop until the goal state (2, 0) reached
- – Apply a rule whose left side matches the current state
- – Set the new current state to be the resulting state
- (0, 0) – Start State
- (0, 3) – Rule 2, Fill the 3-liter jug
- (3, 0) – Rule 9, Pour all the water from the 3-liter jug into the 4-liter jug.
- (3, 3) – Rule 2, Fill the 3-liter jug
- (4, 2) – Rule 7, Pour water from the 3-liter jug into the 4-liter jug until the 4-liter jug is full.
- (0, 2) – Rule 5, Empty the 4-liter jug on the ground
- (2, 0) – Rule 9, Pour all the water from the 3-liter jug into the 4-liter jug.
- **Goal State reached**

Another solution to Water Jug Problem in Artificial Intelligence

- **Another solution to Water Jug Problem in Artificial Intelligence**
- (0, 0) – Start State
- (4, 0) – Rule 1, Fill the 4-liter jug
- (1, 3) – Rule 8, Pour water from the 4-liter jug into the 3-liter jug until the 3-liter jug is full.
- (1, 0) – Rule 6, Empty the 3-liter jug on the ground
- (0, 1) – Rule 10, Pour all the water from the 4-liter jug into the 3-liter jug.
- (4, 1) – Rule 1, Fill the 4-liter jug
- (2, 3) – Rule 8, Pour water from the 4-liter jug into the 3-liter jug until the 3-liter jug is full.
- **Goal State reached**