**** **Bansilal Ramnath Agarwal Charitable Trust’s**

**Vishwakarma Institute of Information Technology, Pune-48**

**(An Autonomous Institute affiliated to Savitribai Phule Pune University)**

**Department of Computer Science and Engineering (Artificial Intelligence)**

**LAB SUBMISSION**

**Artificial Intelligence**

**CAUA31201**

*Submitted by:*

*Avdhoot Nilesh Pimparkar,*

*PRN: 22210178,*

*Roll Number: 381054*

*Third Year*

*Semester I Academic Year 2024-25*

**Assignment: 2**

Aim:

1. To understand the basics of Constraint Satisfaction Problems (CSP) and how they apply to real-world problems.
2. To learn how to represent a problem in terms of variables, domains and constraints.
3. To understand how to use backtracking algorithms to search for solutions that satisfy all constraints.
4. To learn about optimization techniques like forward checking and constraint propagation to improve search efficiency.
5. To understand how CSPs can be applied to problems like scheduling, map colouring and Sudoku, among others.

Theory:

1. **Constraint Satisfaction Problem (CSP) Overview:**

A Constraint Satisfaction Problem (CSP) is a mathematical problem defined by a set of variables, each having a domain of possible values, and a set of constraints that specify allowable combinations of values. The objective is to assign values to all variables in such a way that all constraints are satisfied.

1. **Components of a CSP:**

* **Variables:** These represent the unknowns in the problem that need to be assigned values. For example, in a scheduling problem, variables could represent time slots.
* **Domains:** Each variable has a domain, which is the set of possible values it can take. For instance, in a map-colouring problem, the domain might be a set of colours (e.g., red, blue, green).
* **Constraints:** These are rules that restrict the values that variables can take. Constraints typically involve relationships between two or more variables. For example, in map colouring, two adjacent regions cannot have the same colour.

1. **Problem Overview:**

The Missionaries and Cannibals problem is a classic river-crossing puzzle that illustrates state space search and constraint satisfaction. The scenario involves three missionaries and three cannibals who must cross a river using a boat that can carry at most two people at a time. The challenge is to transport everyone across without ever leaving a group of missionaries outnumbered by cannibals, as this would lead to the missionaries being eaten.

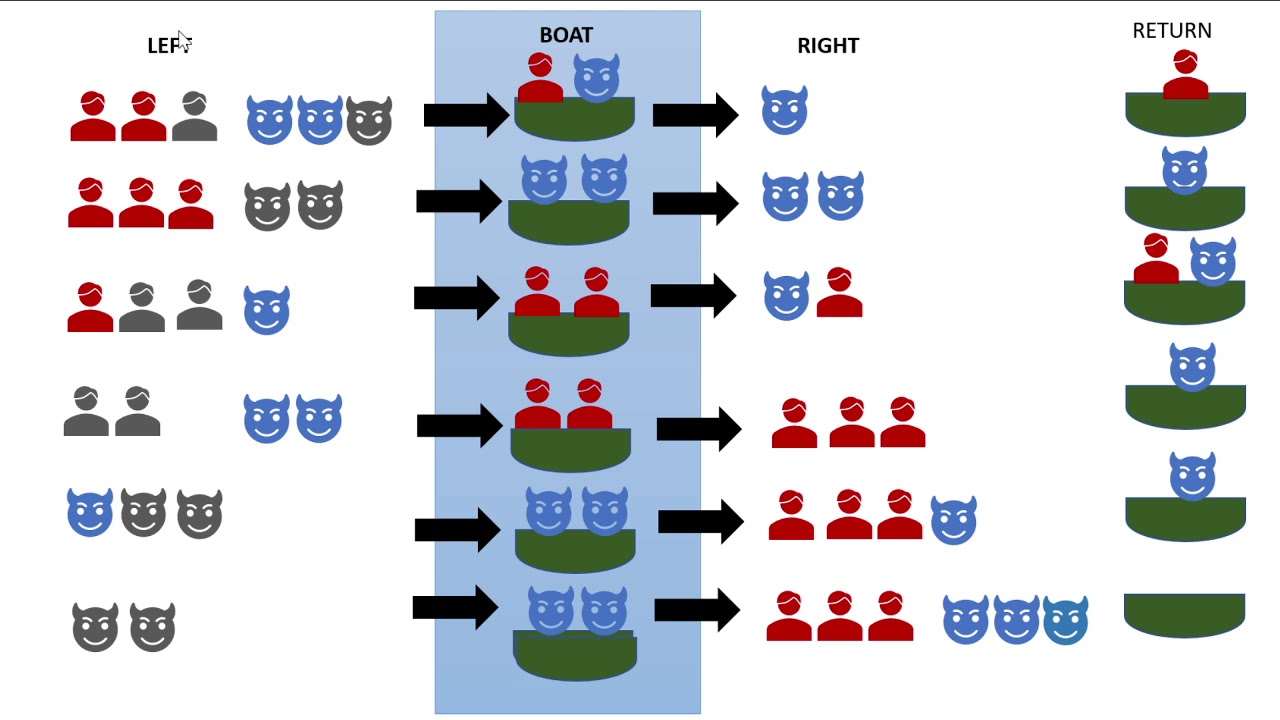


Figure . 3 Missionary 3 Cannibals Problem

1. **Components of the Problem:**

* **State Representation:**

Each state is represented as (M\_left, C\_left, Boat\_position), where:

* **M\_left:** Number of missionaries on the starting side.
* **C\_left:** Number of cannibals on the starting side.
* **Boat\_position:** Indicates if the boat is on the left or right side of the river.
* **Initial State:** (3, 3, 'left'), with all missionaries and cannibals on the left side.
* **Goal State:** (0, 0, 'right'), where everyone has successfully crossed to the right side.

1. **Constraints:**

The number of cannibals must never exceed the number of missionaries on either side of the river unless there are no missionaries present.

1. **Possible Actions (Moves):**

**Actions involve moving one or two people across the river:**

* **Move 1 missionary.**
* **Move 2 missionaries.**
* **Move 1 cannibal.\**
* **Move 2 cannibals.**
* **Move 1 missionary and 1 cannibal.**

1. **Search Approach:**

The problem can be solved using search algorithms like Depth-First Search (DFS) or Breadth-First Search (BFS). Each move generates a new state, which is checked against the constraints to ensure validity.

1. **Example Solution Sequence:**

A possible sequence of moves to solve the problem is:

1. Two cannibals cross the river.
2. One cannibal returns.
3. Two cannibals cross again.
4. One cannibal returns.
5. Two missionaries cross the river.
6. One cannibal and one missionary return.
7. Two missionaries cross the river.
8. One cannibal returns.
9. Two cannibals cross the river.
10. One cannibal returns.
11. Two cannibals cross the river.

This sequence ensures that at no point are the missionaries outnumbered by cannibals.

Conclusion:

The Missionaries and Cannibals problem serves as an excellent illustration of constraint satisfaction and search strategies in problem-solving. By effectively modelling the problem, we can apply search algorithms like Depth-First Search (DFS) or Breadth-First Search (BFS) to explore possible moves and ensure that all constraints are respected. The solution requires careful planning to avoid leaving missionaries outnumbered by cannibals. This problem not only highlights the challenges of constraint satisfaction but also provides a clear example of how structured approaches can lead to successful outcomes in complex scenarios.