Artificial Intelligence

BS (CS) _Spring_2025

Lab_04 Manual

Learning Objectives:

1. Uninformed searches (BFS,DFS,UCS)
2. Greedy Search

Lab Manual

Uninformed Search:

Uninformed search is a class of general-purpose search algorithms. Uninformed search algorithms do not have additional information about state or search space other than how to traverse the tree, so it is also called **blind** search. They operate in a brute force, meaning they try out every part of search space blindly.

Types:

1. Breadth-first Search
2. Depth-first Search
3. Depth-limited Search
4. Iterative deepening depth-first search
5. Uniform cost search

Uniform Cost Search

Uniform Cost Search (UCS) is a **variant of BFS** that considers **path cost** rather than depth. It always expands the **least-cost node** first, using a **priority queue**.

Features:

- Guarantees **optimal solution** if all costs are positive.
- Uses a **priority queue** to expand the lowest-cost node first.
- Works well in weighted graphs.

Step-by-Step Guide:

1. **Initialize** a priority queue with the **start node and cost = 0**.
2. Expand the **node with the least cost** from the priority queue.
3. If the **goal is found**, return the path.
4. Otherwise, **add neighbors to the queue with their updated costs**.
5. Repeat until the goal is found or the queue is empty.

Informed Search:

Informed search algorithms, also known as **heuristic search algorithms**, use problem-specific knowledge to find solutions **more efficiently** than uninformed search algorithms. These algorithms make use of a heuristic function to estimate the cost of reaching the goal from a given state.

Heuristics function:

Heuristic is a function which is used in Informed Search, and it finds the most promising path. It provides an **estimate** of the cost from a given node to the goal node. The heuristic method, however, might not always give the best solution, but it guaranteed to find a good solution in reasonable time. Heuristic function estimates how close a state is to the goal. It is represented by $\boldsymbol{h(n)}$, and it calculates the cost of an optimal path between the pair of states. The value of the **heuristic function is always positive.**

Admissibility of the heuristic function is given as:

$$\boldsymbol{h(n)} <= \boldsymbol{h*(n)}.$$

Here $\boldsymbol{h(n)}$ is heuristic cost, and $\boldsymbol{h*(n)}$ is the estimated cost. Hence heuristic cost should be less than or equal to the estimated cost. Two popular informed search algorithms are **Greedy Search** and **A* (A-star).**

1. Greedy Search:

Greedy Search is an informed search algorithm that always chooses the path that appears to be the best at the current moment. It evaluates each state based solely on the heuristic function, without considering the cost of reaching that state. While Greedy Search can find solutions quickly, it **may not** always find the **optimal solution.**

Step by step Guide:

1. **Initialize** the **open list** with the initial state and its **heuristic value**.
2. **Initialize** the **closed list** as empty.
3. **While the open list is not empty:**
    - Remove the state with the **lowest heuristic value** from the open list.
    - If the state is the **goal**, **return the path** to the goal.
    - Otherwise, **expand the state** and add its **neighbors** to the open list with their heuristic values.
    - Add the **current state** to the **closed list**.
4. If the open list is empty and the goal state **has not been found,** return failure**.**