# SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

## Department of AI &DS

### Course Name – 23ADT201 ARTIFICIAL INTELLIGENCE

### II Year / III Semester

## UNIT 2

## Topic:PROBLEM SOLVING

CASE STUDY:

**Problem Definition**

- **Objective**: Find the most efficient route for a delivery driver to take to deliver packages to multiple locations in a city.
- **Heuristic**: Minimize the total travel time or distance.

# HEURISTIC SEARCH STRATEGIES

1. Heuristic search strategies in AI are techniques used to solve problems more efficiently when traditional methods are impractical due to time or computational constraints.
2. These strategies rely on heuristics, which are rules of thumb or educated guesses, to guide the search process towards more promising paths in the solution space

# HEURISTIC SEARCH STRATEGIES

Uninformed Search Algorithms

- Uninformed search is a class of general-purpose search algorithms which operates in brute force-way.
- 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.

# HEURISTIC SEARCH STRATEGIES

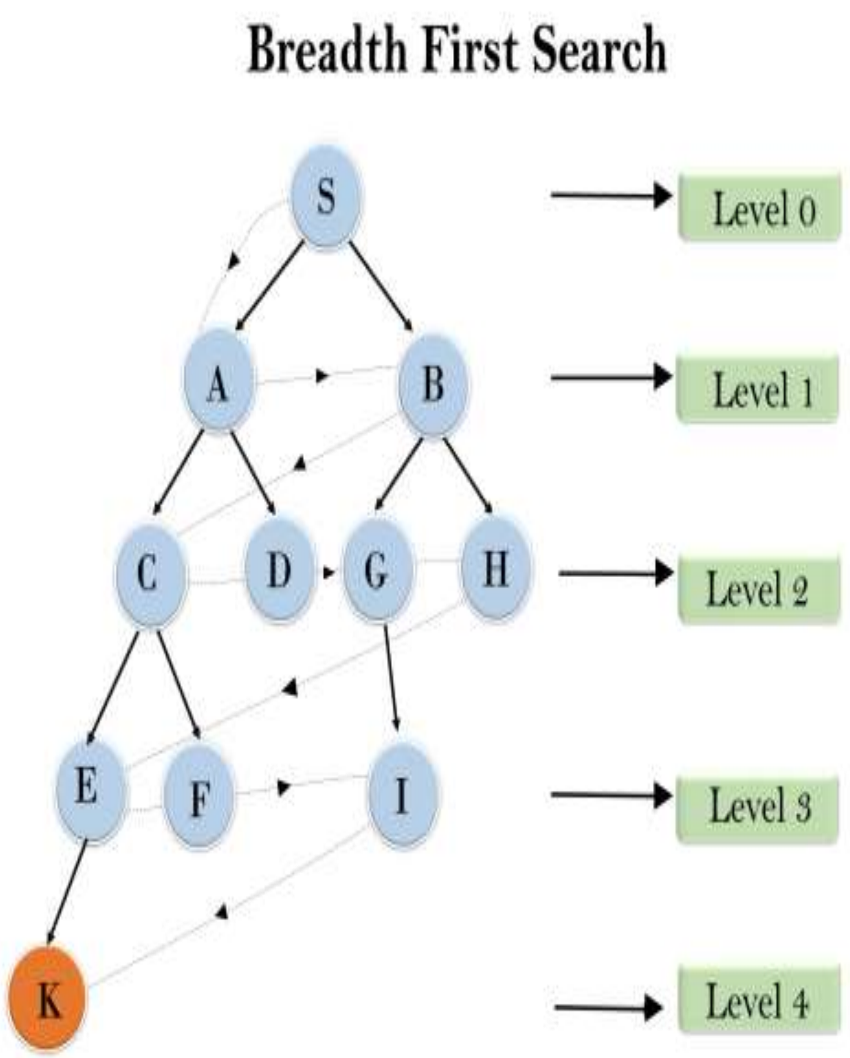
Following are the various types of uninformed search algorithms:

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

# HEURISTIC SEARCH STRATEGIES

1. Breadth-first Search:

- Breadth-first search is the most common search strategy for traversing a tree or graph. This algorithm searches breadthwise in a tree or graph, so it is called breadth-first search.
- BFS algorithm starts searching from the root node of the tree and expands all successor node at the current level before moving to nodes of next level.
- The breadth-first search algorithm is an example of a general-graph search algorithm.
- Breadth-first search implemented using FIFO queue data structure.




In the below tree structure, we have shown the traversing of the tree using BFS algorithm from the root node S to goal node K. BFS search algorithm traverse in layers, so it will follow the path which is shown by the dotted arrow, and the traversed path will be:

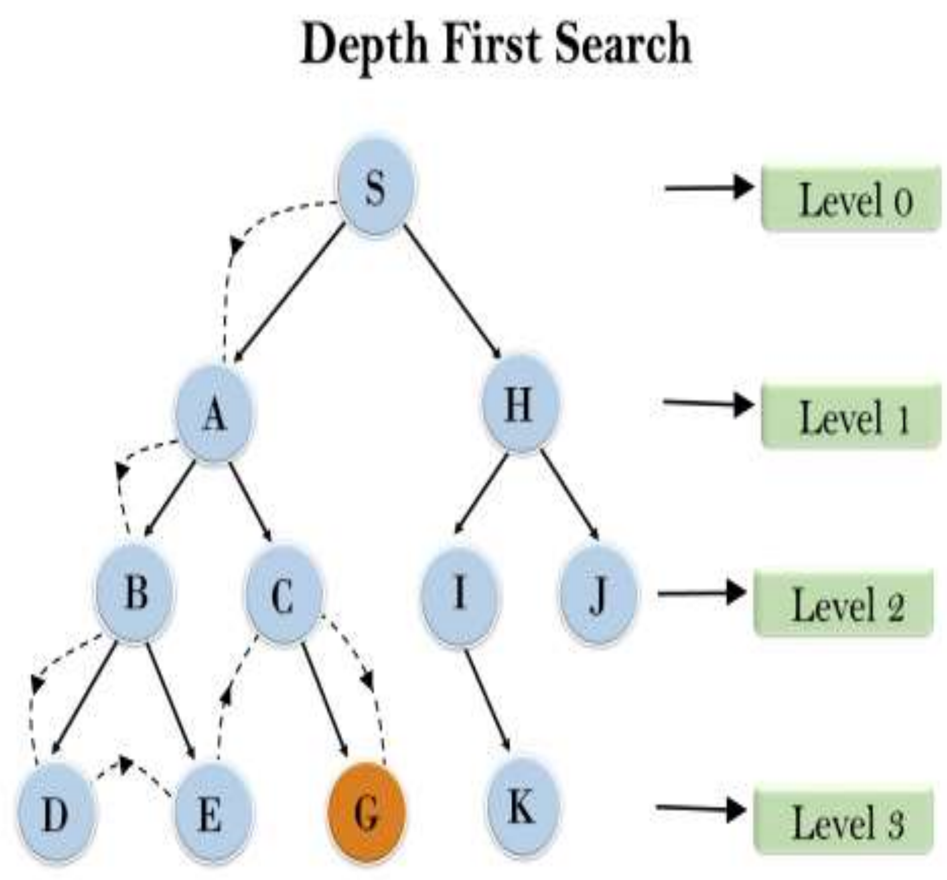
1. S---> A--->B---->C--->D---->G--->H--->E---->F---->I---->K

# HEURISTIC SEARCH STRATEGIES

2. Depth-first Search

- Depth-first search isa recursive algorithm for traversing a tree or graph data structure.
- It is called the depth-first search because it starts from the root node and follows each path to its greatest depth node before moving to the next path.
- DFS uses a stack data structure for its implementation.
- The process of the DFS algorithm is similar to the BFS algorithm.

# HEURISTIC SEARCH STRATEGIES




1. In the below search tree, we have shown the flow of depth-first search, and it will follow the order as:
2. Root node--->Left node ----> right node.
3. It will start searching from root node S, and traverse A, then B, then D and E, after traversing E, it will backtrack the tree as E has no other successor and still goal node is not found. After backtracking it will traverse node C and then G, and here it will terminate as it found goal node.

# HEURISTIC SEARCH STRATEGIES

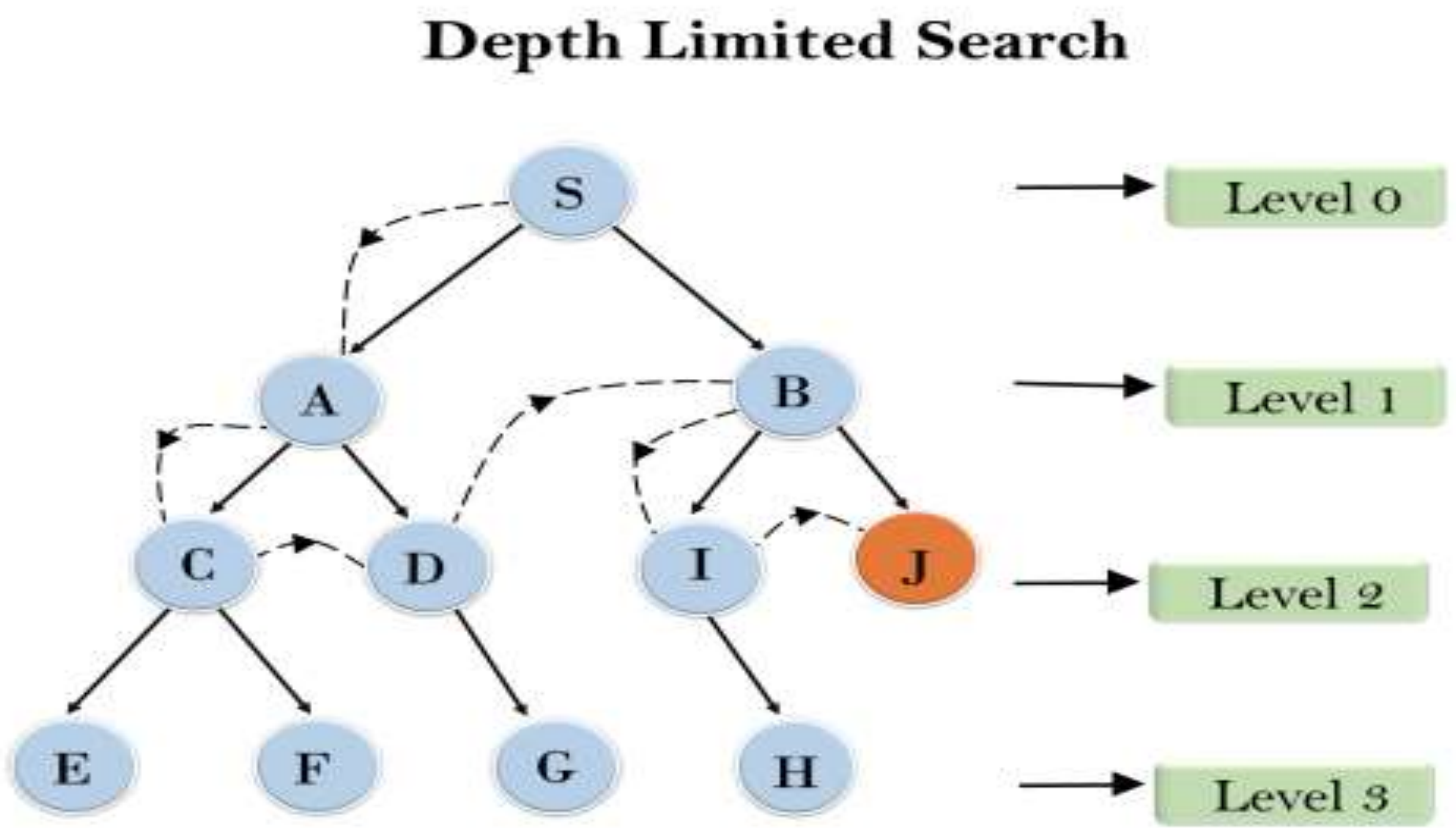
3. Depth-Limited Search Algorithm:

A depth-limited search algorithm is similar to depth-first search with a predetermined limit. Depth-limited search can solve the drawback of the infinite path in the Depth-first search. In this algorithm, the node at the depth limit will treat as it has no successor nodes further.

Depth-limited search can be terminated with two Conditions of failure:

- Standard failure value: It indicates that problem does not have any solution.
- Cutoff failure value: It defines no solution for the problem within a given depth limit.

# HEURISTIC SEARCH STRATEGIES



# HEURISTIC SEARCH STRATEGIES

4. Uniform-cost Search Algorithm:

Uniform-cost search is a searching algorithm used for traversing a weighted tree or graph.

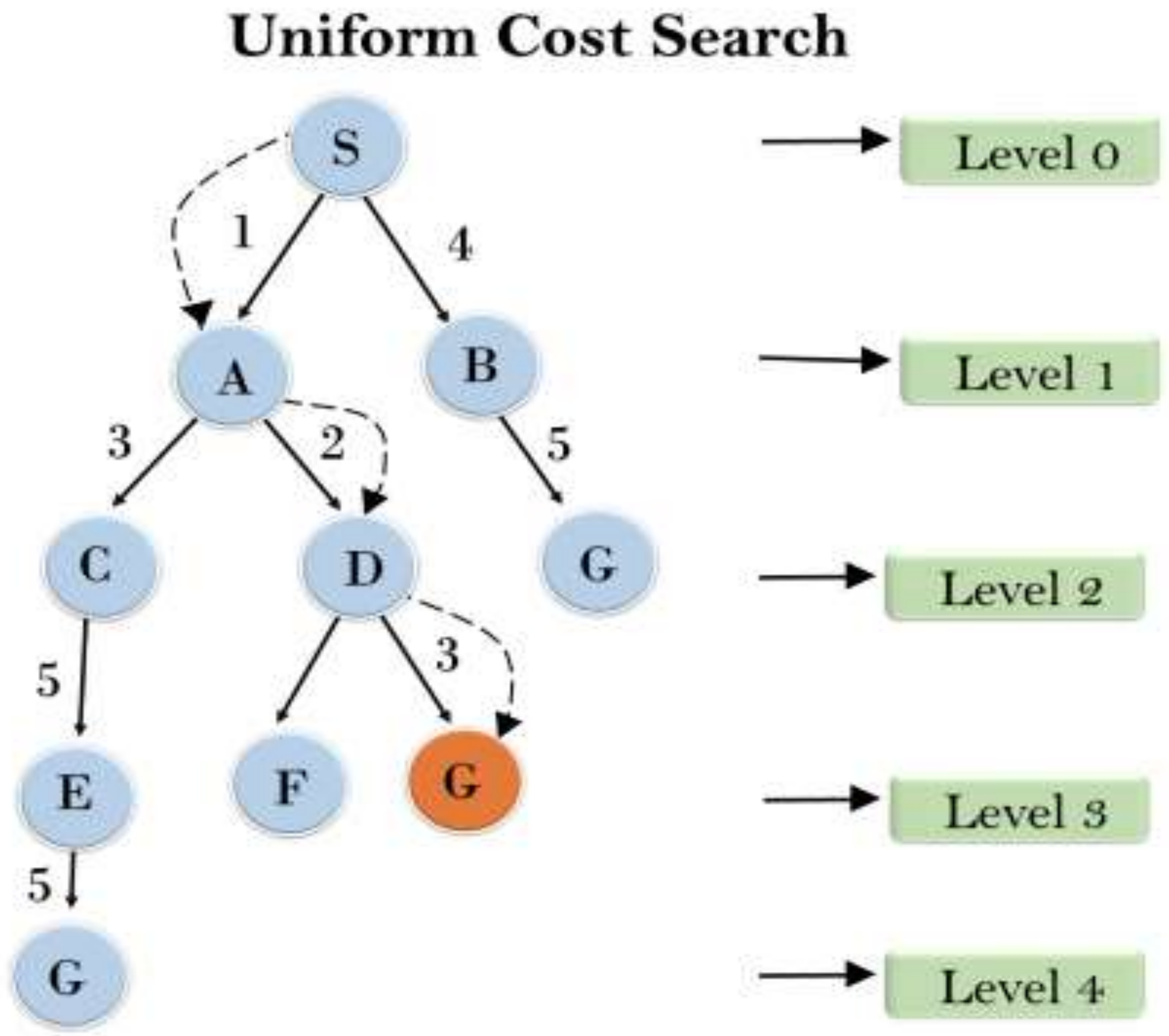
This algorithm comes into play when a different cost is available for each edge.

The primary goal of the uniform-cost search is to find a path to the goal node which has the lowest cumulative cost. Uniform-cost search expands nodes according to their path costs form the root node. It can be used to solve any graph/tree where the optimal cost is in demand.

A uniform-cost search algorithm is implemented by the priority queue.

It gives maximum priority to the lowest cumulative cost. Uniform cost search is equivalent to BFS algorithm if the path cost of all edges is the same.

# HEURISTIC SEARCH STRATEGIES



# HEURISTIC SEARCH STRATEGIES

Iterative deepening depth-first Search:

The iterative deepening algorithm is a combination of DFS and BFS algorithms. This search algorithm finds out the best depth limit and does it by gradually increasing the limit until a goal is found.

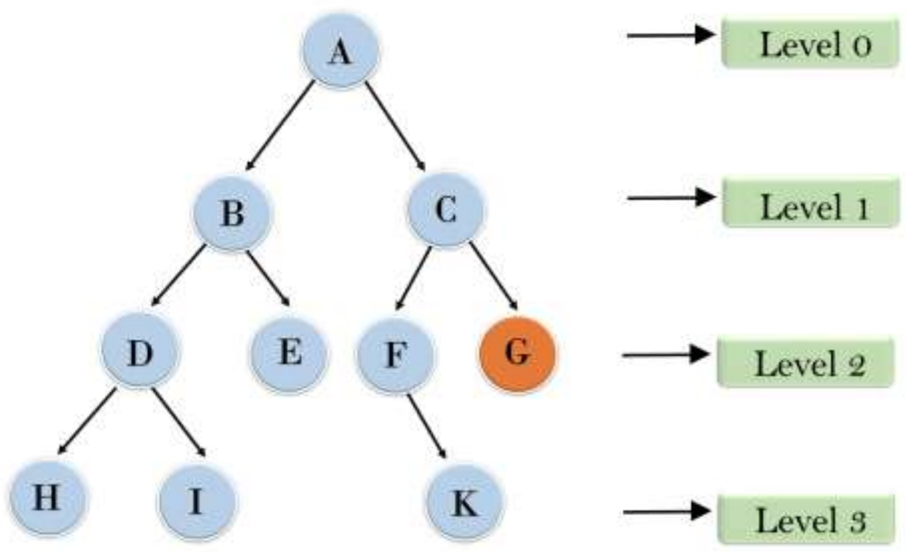
This algorithm performs depth-first search up to a certain "depth limit", and it keeps increasing the depth limit after each iteration until the goal node is found.

This Search algorithm combines the benefits of Breadth-first search's fast search and depth-first search's memory efficiency.

The iterative search algorithm is useful uninformed search when search space is large, and depth of goal node is unknown.

# HEURISTIC SEARCH STRATEGIES

Iterative deepening depth first search




1'st Iteration-----> A
2'nd Iteration----> A, B, C
3'rd Iteration------>A, B, D, E, C, F, G
4'th Iteration------>A, B, D, H, I, E, C, F, K, G
In the fourth iteration, the algorithm will find the goal node.

# HEURISTIC SEARCH STRATEGIES

6. Bidirectional Search Algorithm:

- Bidirectional search algorithm runs two simultaneous searches, one form initial state called as forward-search and other from goal node called as backward-search, to find the goal node.
- Bidirectional search replaces one single search graph with two small subgraphs in which one starts the search from an initial vertex and other starts from goal vertex.
- The search stops when these two graphs intersect each other.
- Bidirectional search can use search techniques such as BFS, DFS, DLS, etc.

# HEURISTIC SEARCH STRATEGIES

Informed Heuristic Functions and Advanced Search Algorithms

Introduction to Informed Heuristic Functions

In AI, heuristic functions play a critical role in guiding search algorithms towards efficient solutions. An informed heuristic function provides an estimate of the cost from a current state to the goal, aiding algorithms in making informed decisions about which paths to explore.

Properties of Heuristic Functions

- Admissibility: A heuristic function h(n)h(n)h(n) is admissible if it never overestimates the true cost to reach the goal from state nnn.
- Consistency (Monotonicity): For every state nnn and its successor mmm, the estimated cost h(n)h(n)h(n) should be less than or equal to the cost of reaching mmm plus the estimated cost from mmm to the goal, ensuring consistency in estimates.

Greedy Best-First Search

- Concept: Greedy Best-First Search (GBFS) is an informed search algorithm that selects the node which appears closest to the goal based solely on the heuristic function h(n)h(n)h(n).
- Strategy: At each step, GBFS expands the node with the lowest h(n)h(n)h(n) value, making it greedy in nature as it prioritizes immediate gains based on the heuristic.
- Limitations: GBFS does not guarantee optimality since it may get stuck in local optima based on the heuristic estimates alone, without considering the actual path cost.

# HEURISTIC SEARCH STRATEGIES

A* Search Algorithm

- Overview: A* is a widely-used informed search algorithm that combines the actual path cost g(n)g(n)g(n) from the start node to node nnn with the heuristic function h(n)h(n)h(n).
- Optimality: A* is both complete (finds a solution if one exists) and optimal (finds the shortest path) when the heuristic function h(n)h(n)h(n) is admissible.
- F-function: The total cost function f(n)f(n)f(n) in A* is defined as f(n)=g(n)+h(n)f(n) = g(n) + h(n)f(n)=g(n)+h(n), guiding the algorithm to expand nodes based on their combined path cost and heuristic estimate.

# LEARNING TO SEARCH BETTER

Memory-Bounded Search

- Challenge: In scenarios with limited memory resources, traditional search algorithms like A* may face challenges due to the large number of nodes stored in memory.
- Memory-Bounded Heuristic Search: Techniques such as Memory-Bounded A* (MA*) address this by limiting the number of nodes stored in memory, focusing resources on nodes most likely to lead to the goal.
- Trade-offs: MA* trades off optimality for memory efficiency, ensuring that it operates within predefined memory constraints while attempting to find a satisfactory solution.

# LEARNING TO SEARCH BETTER

Learning to Search Better

- Machine Learning Integration: Recent advancements in AI have explored integrating machine learning techniques to enhance heuristic functions and search algorithms.
- Benefits: Learning-based approaches can adaptively improve heuristic estimates based on experience or training data, potentially leading to more accurate and efficient search strategies.
- Applications: Learning to search better can be applied in domains where heuristic functions are challenging to define manually, such as complex decision-making processes or dynamic environments.

# THANK YOU