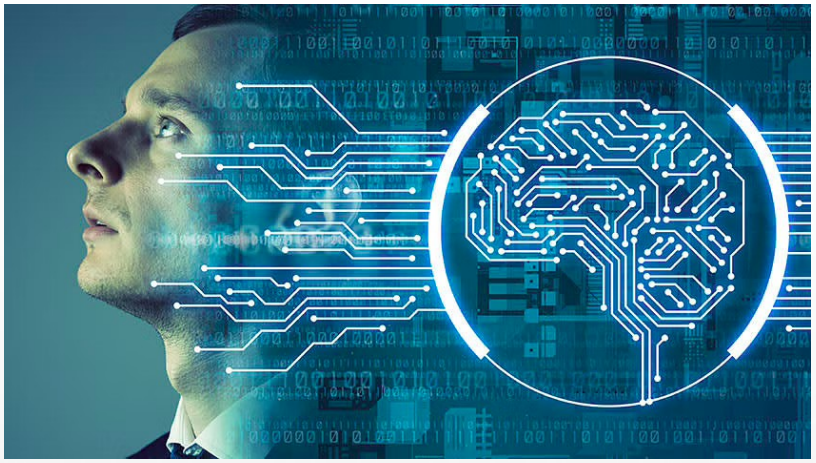
CTU
CZECH TECHNICAL UNIVERSITY IN PRAGUE

Uleam
UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ

Search Algorithms in AI

Master. Cesar Sinchiguano
Robotics and Cybernetics

Today's lecture



What is a search algorithm?

Importance of search algorithms

Properties of search algorithms

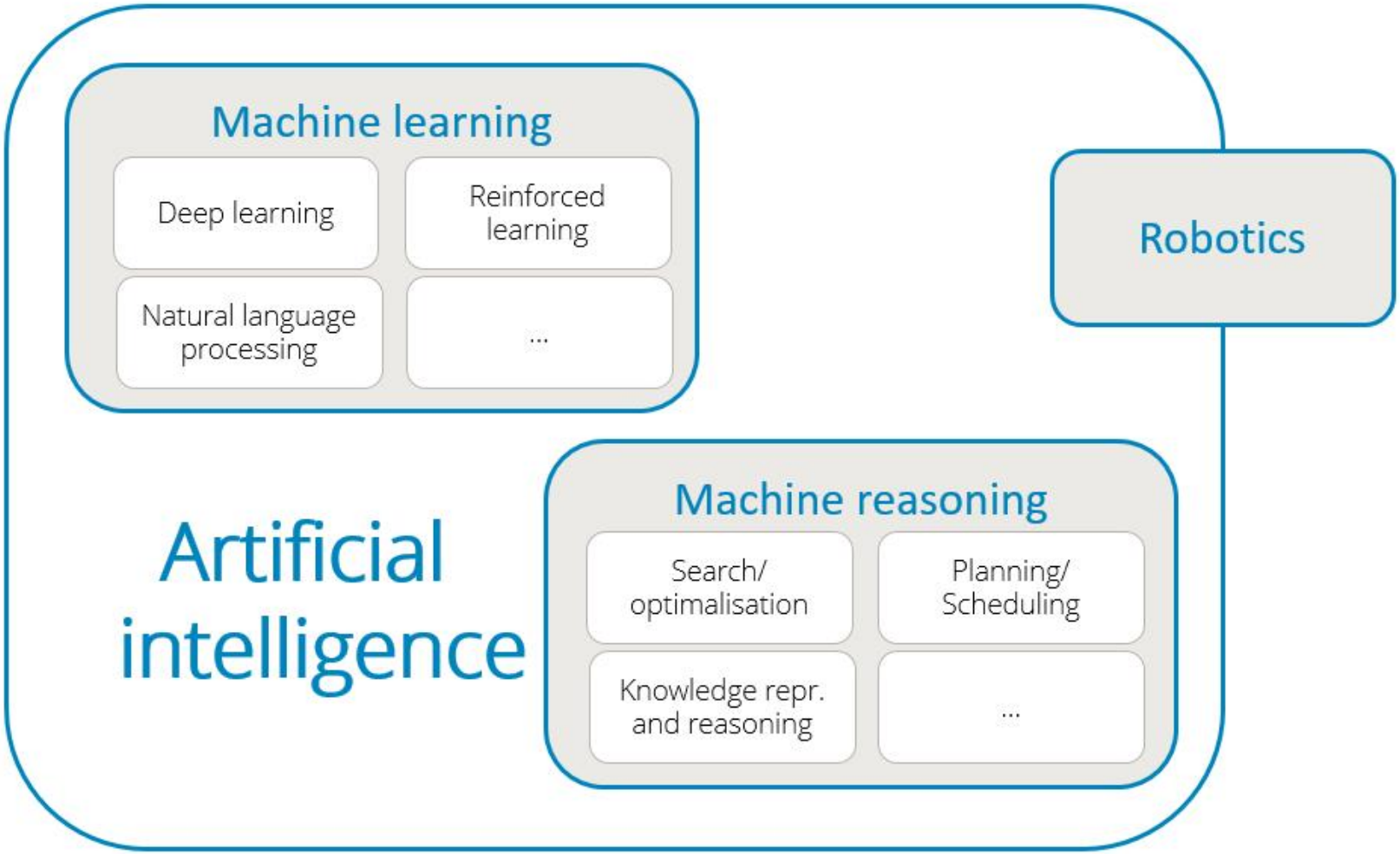
How search algorithms work

Types of search algorithms

* Uninformed search algorithms

* Informed search algorithms

Applications of search algorithms

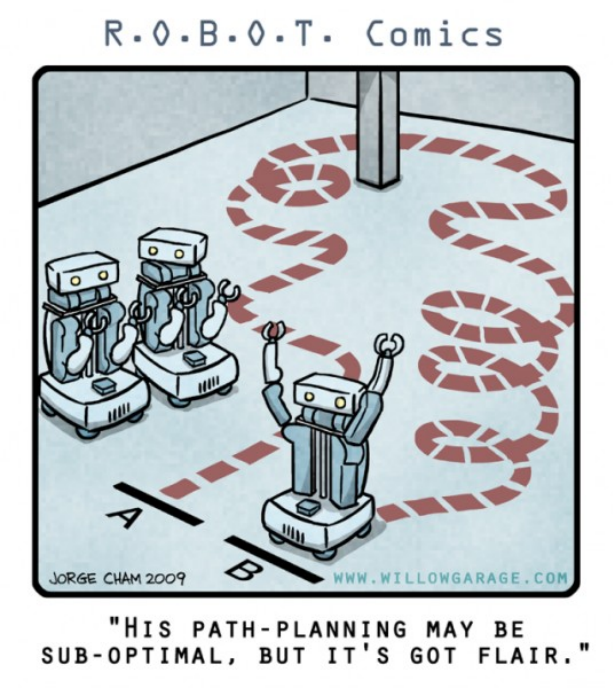

Machine learning
Deep learning
Reinforced learning
Natural language processing
...
Robotics
Artificial intelligence
Machine reasoning
Search/ optimalisation
Planning/ Scheduling
Knowledge repr. and reasoning
...

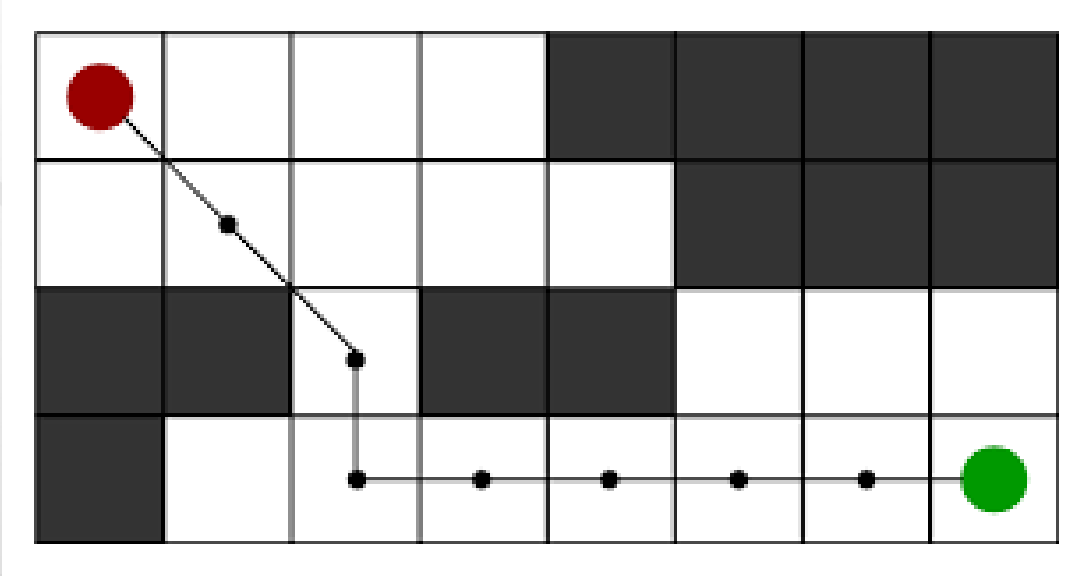
What is a search algorithm?

Search algorithms are algorithms that help in solving search problems.

A search problem consists of a search space, start state, and goal state.

Search algorithms help the AI agents to attain the goal state through the assessment of scenarios.



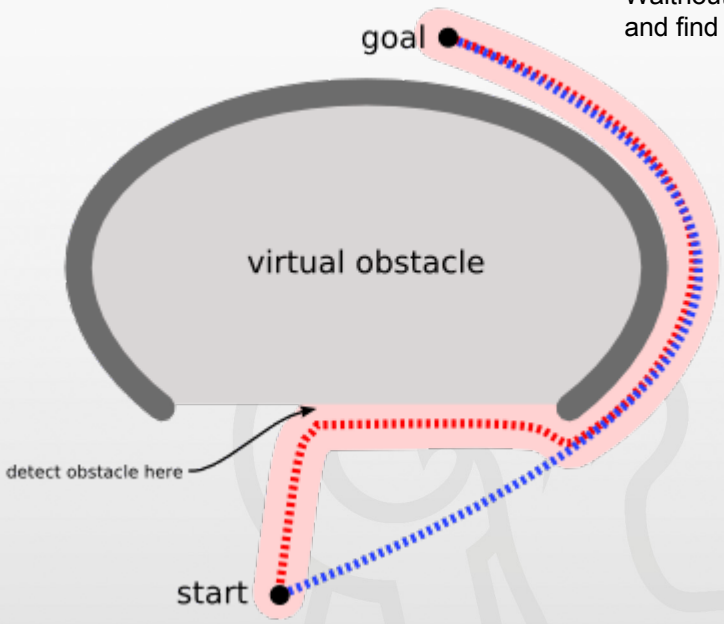


Cognitive: relating to the processes of thinking and reasoning

What is a search algorithm?

The algorithms provide search solutions through a sequence of actions that transform the initial state to the goal state.

Waithout these algorithms, AI machines and applications cannot implement search functions and find viable solutions.



Agents		Environments
Robot	→	Room
Chatbot	→	Chatting
Vehicle	→	Road
Program	→	Data & Rules
Machine	→	Working Field

Cognitive: relating to the processes of thinking and reasoning

Importance of search algorithms

Solving problems:

logical search mechanisms such as problem definition, actions, and search space.

Search programming:

Many AI tasks can be programmed in terms of search

Goal-based agents:

These agents solve problems by searching for the most ideal series of actions that can provide the best solution to a problem.

Importance of search algorithms

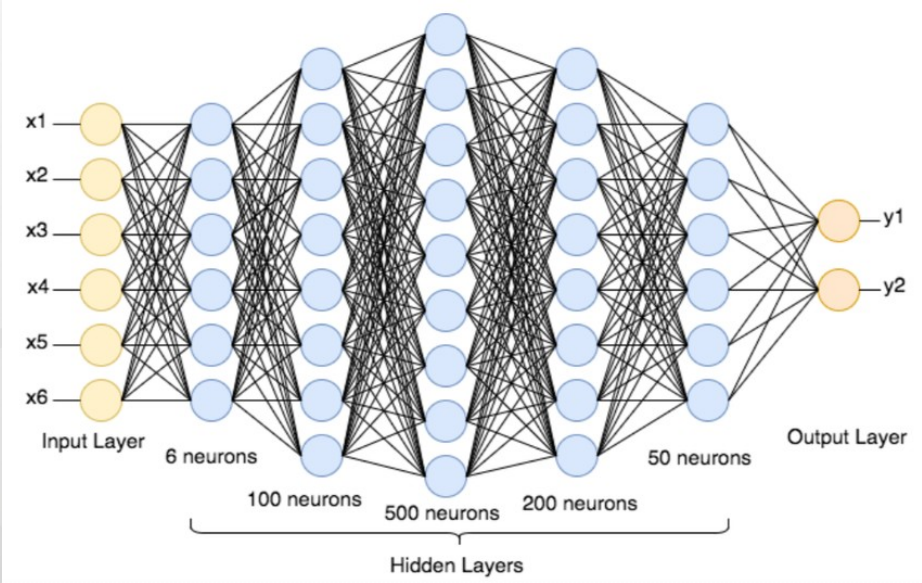
Support production systems:

Production systems use search algorithms to search for the sequence of rules that can result in the desired action.

Neural network systems:

These are computing systems that consist of interconnected nodes, a hidden layer, an input layer, and an output layer.

Neural networks are used to perform various tasks in artificial intelligence. Search algorithms enhance the searching of connection weights that will lead to the desired input-output mapping.



Properties of search algorithms

Completeness:

A search algorithm can be said to be complete if it provides a solution for a given inputÂ

Optimality:

These are the best solutions given by the search algorithms at the lowest path cost.

Time Complexity:

These algorithms have a maximum time needed to accomplish a task or provide a solution.

Space Complexity:

They have a maximum memory or storage space needed when conducting a search operation.

This memory is also based on the complexity of the task.

How search algorithms work

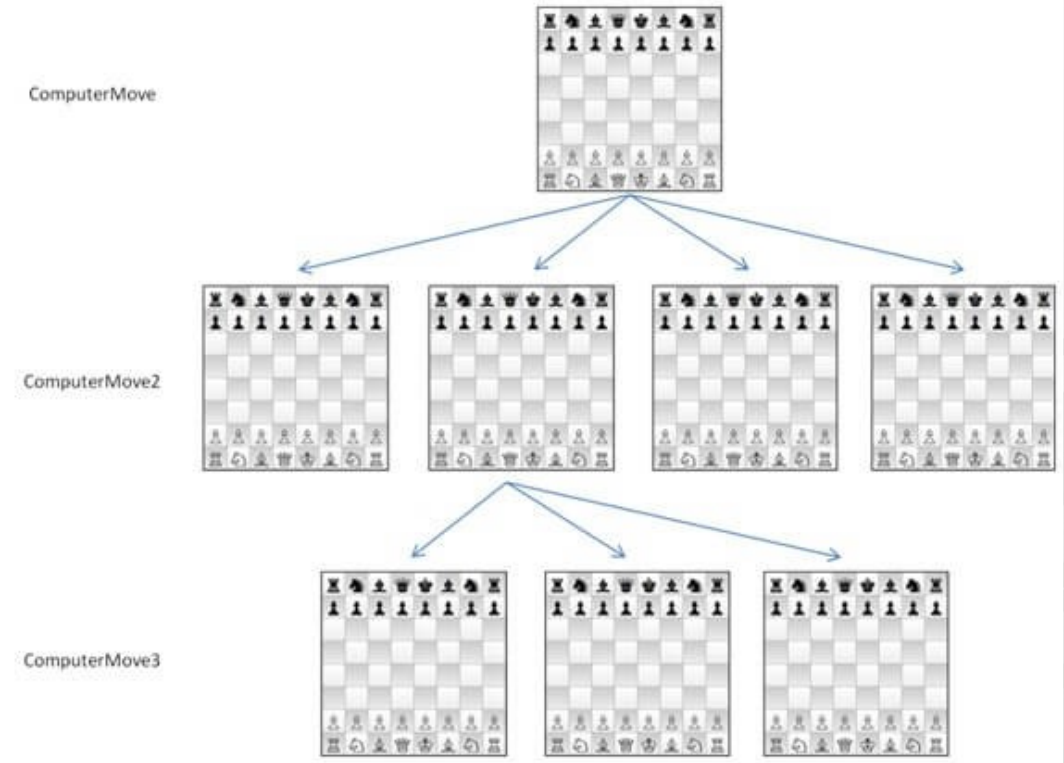
The AI agents find the best solution for the problem by searching for all the possible solutions.

The process of searching is done using search algorithms.

Search algorithms work in two main phases:

Defining the problem

Searching in the search space.



Defining the problem

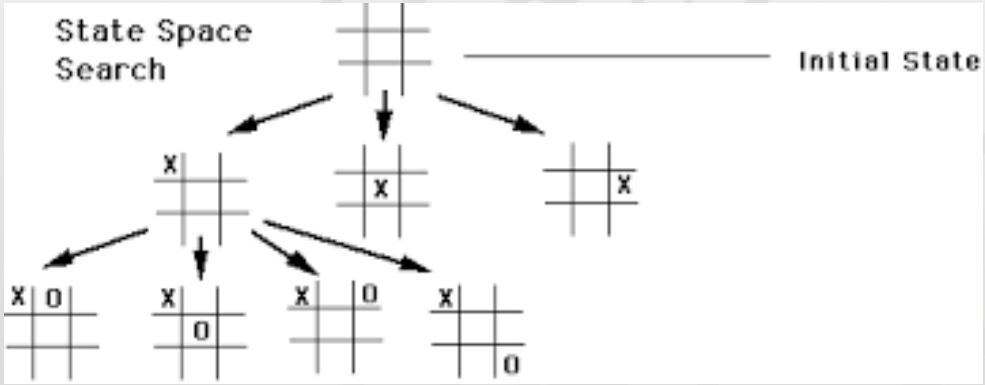
Initial state: This is the start state in which the search starts.

State space: These are all the possible states that can be attained from the initial state through a series of actions.

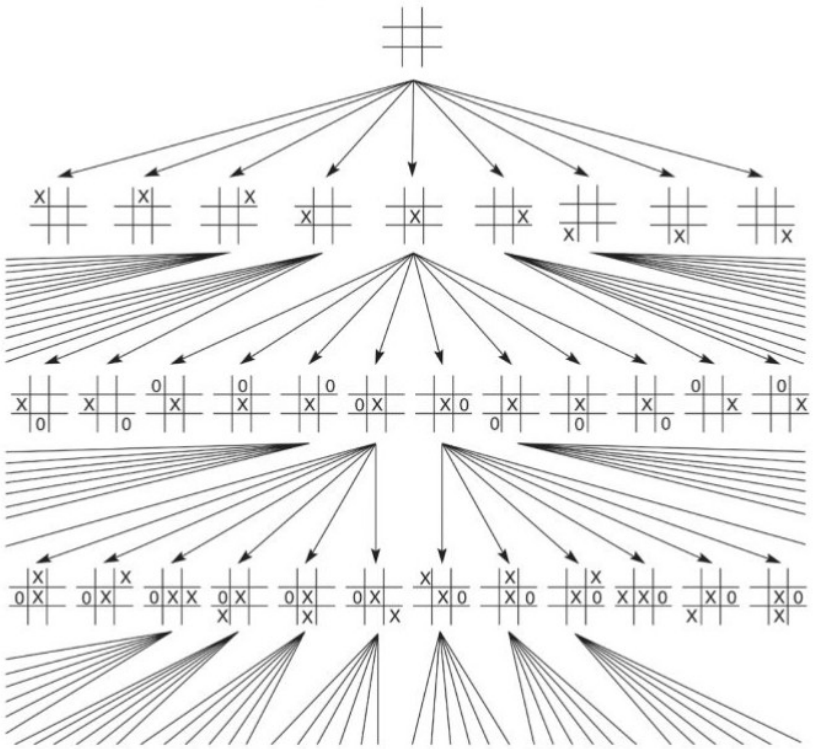
Actions: These are the steps, activities, or operations undertaken by AI agents in a particular state.

Goal state: This is the endpoint or the desired state.

Goal test: This is a test conducted to establish whether a particular state is a goal state.



Portion of the state space for tic-tac-toe.



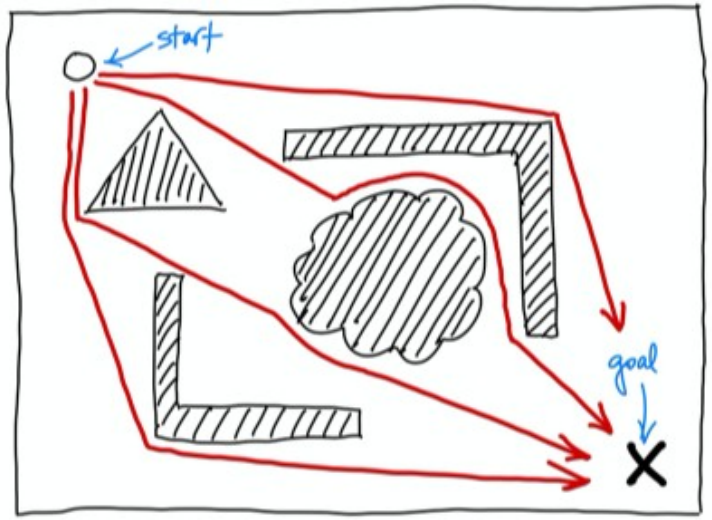
Path cost:Â This is the cost associated with a given path taken by the agents.

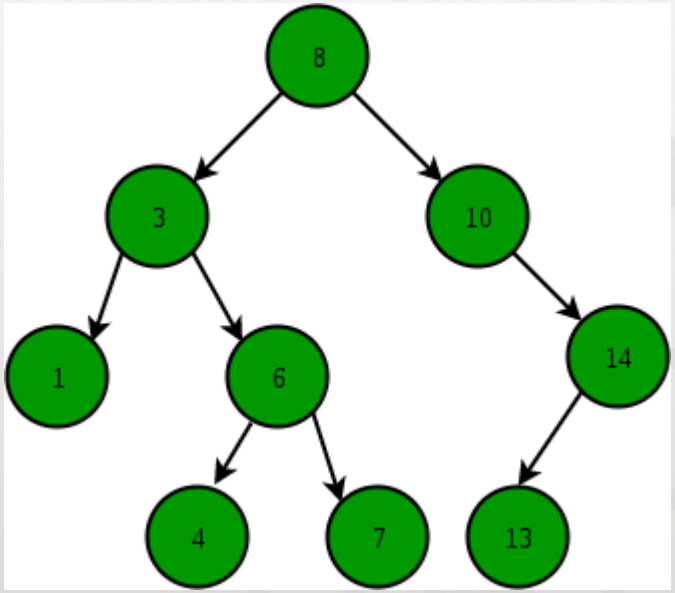
Searching in the search space

The agents use the search algorithms to perform a search in the search space.

A search space is an abstract configuration that consists of a search tree of possible solutions.

A search tree is used to configure the series of actions. The initial state is configured as the root of the search tree. The branches are the actions while the nodes are the outcomes of the actions.

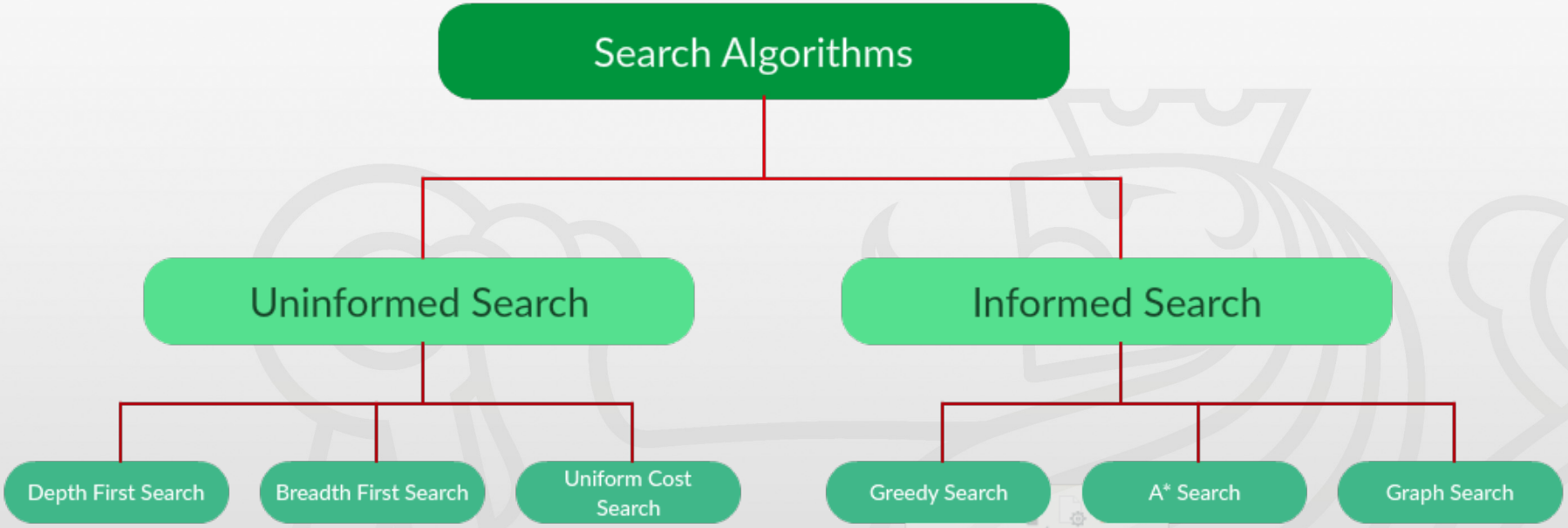




When we have a given problem in AI, the search algorithm will identify the initial state, state space, actions, goal state, and the path cost. From the initial state, a series of actions will be performed as the search algorithms search for the goal state.

Types of search algorithms

Search algorithms can be divided into two broad categories: uninformed search algorithms and informed search algorithms.



Uninformed search algorithms

These algorithms are also called blind algorithms.

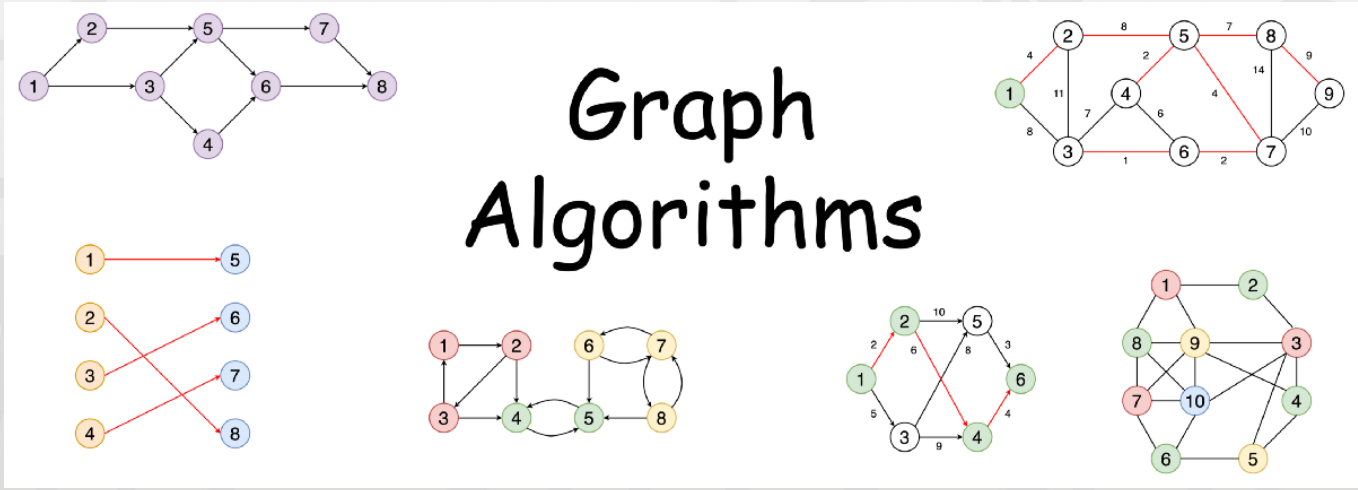
This is because they do not have supplementary information that can assist them to attain the end goal other than the information given in the problem definition

Operates in brute force-way

breadth-first search

depth-first search

uniform cost search



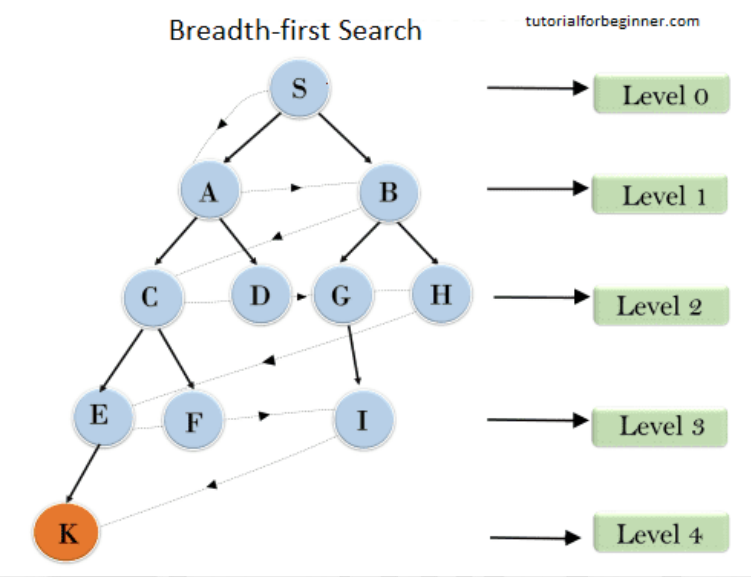
Breadth-first search

Breadth-first search is the most common search strategy for traversing a tree .

It begins at the tree root and traverses all the neighbor nodes in the current depth level before progressing to the nodes existing in the next depth level.

The breadth-first search algorithm is an example of a general-graph search algorithm.

Breadth-first search implemented using FIFO queue data structure.



Advantages:

BFS will provide a solution if any solution exists.

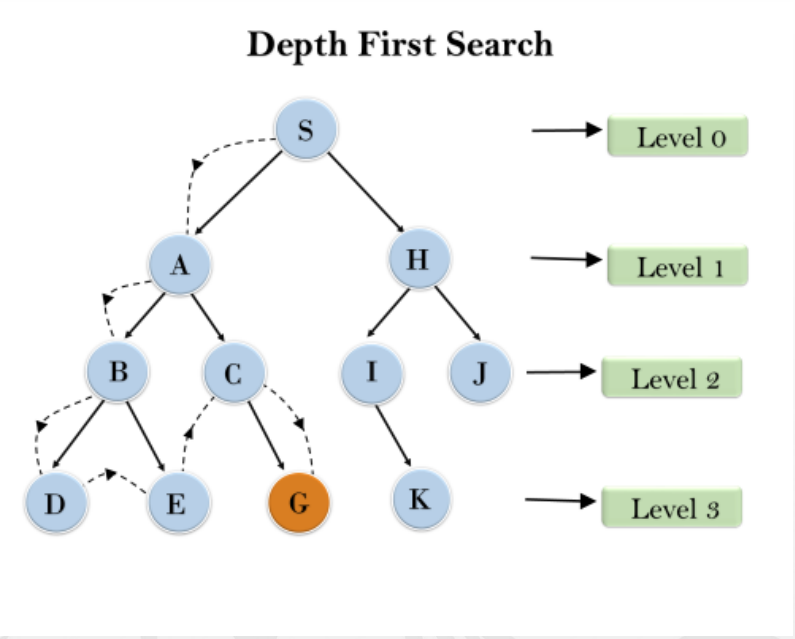
Disadvantages:

It requires lots of memory since each level of the tree must be saved into memory to expand the next level.

BFS needs lots of time if the solution is far away from the root node.

https://youtu.be/x-VTfcmrLEQ

Depth-first search



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.

Advantage:

DFS requires very less memory as it only needs to store a stack of the nodes on the path from root node to the current node.

It takes less time to reach to the goal node than BFS algorithm.

Disadvantage:

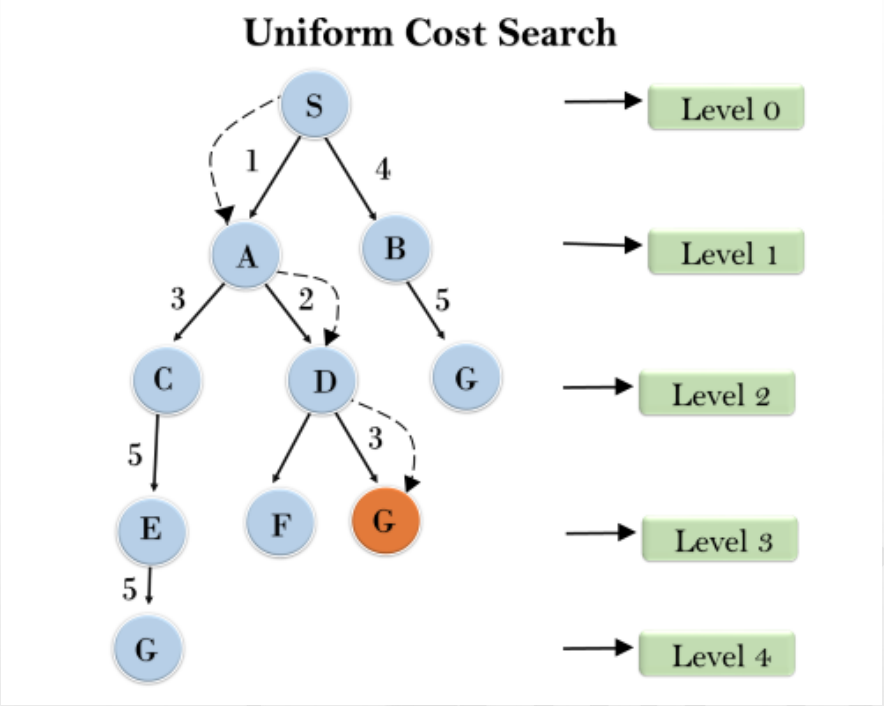
DFS algorithm goes for deep down searching and sometime it may go to the infinite loop.

https://youtu.be/NUgMa5coCoE

Uniform cost search

These algorithms differ from the breadth-first and depth-first algorithms in that they consider the cost.

When there are different paths for attaining the desired goal, the optimal solution of uniform cost algorithms is the solution that is associated with the least cost.



Advantages:

Uniform cost search is optimal because at every state the path with the least cost is chosen.

Disadvantages:

It does not care about the number of steps involve in searching and only concerned about path cost. Due to which this algorithm may be stuck in an infinite loop.

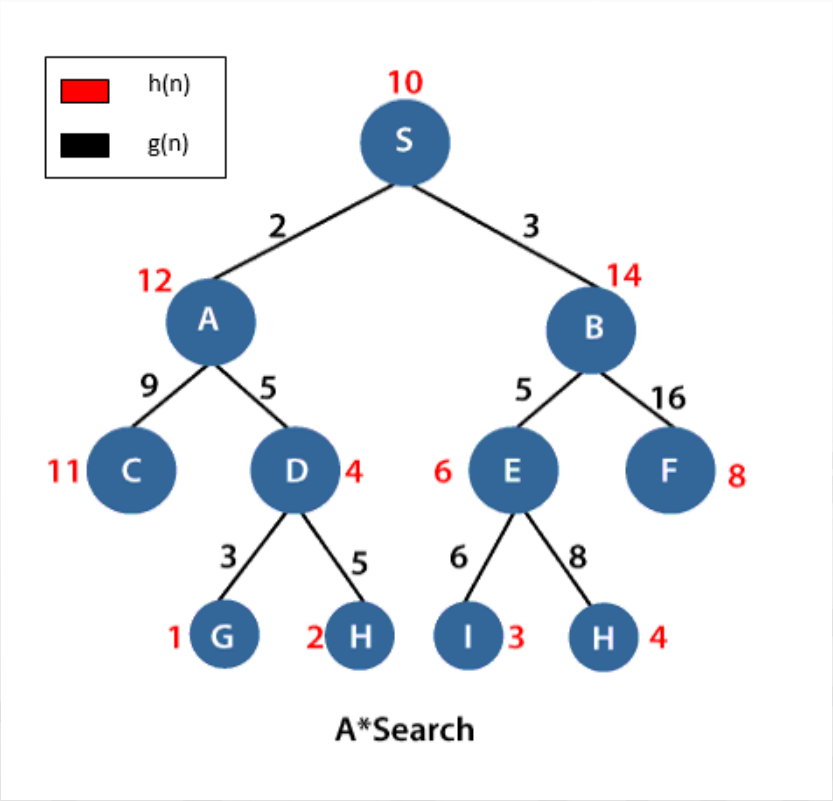
Informed search algorithms

These are heuristic algorithms that consist of the problem definition and supplementary information that assists in achieving the desired goal state.

Informed search algorithms can be further categorized into the following algorithms:

greedy search

A* tree search



A*Search

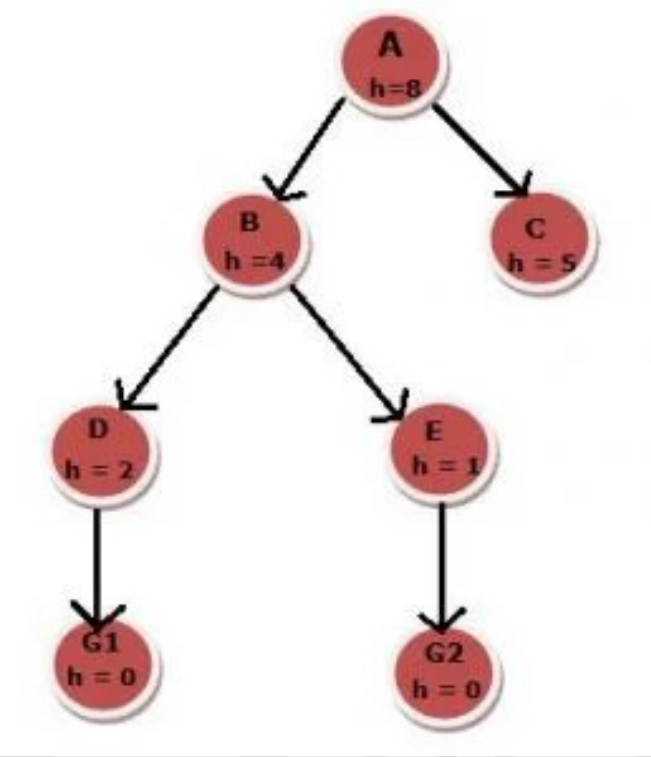
Greedy search

In this algorithm, we expand the closest node to the goal node.

The closeness factor is calculated using a heuristic function h (x). h (x) is an estimate of the distance between one node and the end or goal node.

Example: If we need to find the path from root node A to any goal state having minimum cost using greedy search, then the solution would be A-B-E-H. It will start with B because it has less cost than C, then E because it has less cost than D and then G2.

Â The lower the value of h (x), the closer the node is to the endpoint. When the greedy search is searching for the best path to the goal node, it will choose nodes with the lowest possible values.



A* tree search

This algorithm combines the attributes of the uniform cost algorithm and the greedy algorithm.

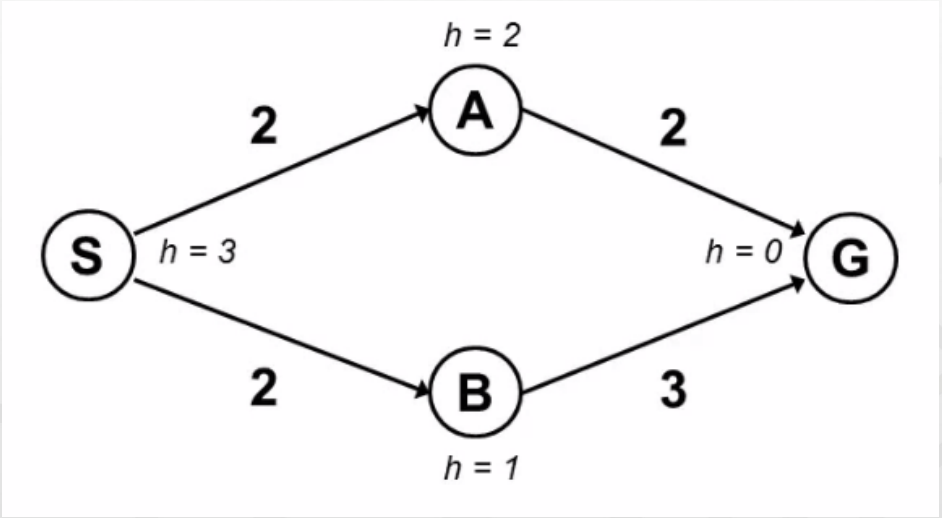
In this algorithm, the total cost (heuristic) which is denoted by f(x) is a sum of the cost in uniform cost search denoted by g(x) and cost of greedy search denoted by h(x).

$$f(x) = g(x) + h(x)$$

g(n) the cost (so far) to reach the node

h(n) estimated cost to get from the node to the goal

f(n) estimated total cost of path through n to goal. It is implemented using priority queue by increasing f(n).



A* tree search

This algorithm combines the attributes of the uniform cost algorithm and the greedy algorithm.

In this algorithm, the total cost (heuristic) which is denoted by f(x) is a sum of the cost in uniform cost search denoted by g(x) and cost of greedy search denoted by h(x).

$$f(x) = g(x) + h(x)$$

g(n) the cost (so far) to reach the node

h(n) estimated cost to get from the node to the goal

f(n) estimated total cost of path through n to goal. It is implemented using priority queue by increasing f(n).


CTU
CZECH TECHNICAL
UNIVERSITY
IN PRAGUE

Thanks

cesar.sinchiguano@uleam.edu.ec

https://inteligenciaartificialgrupo33.blogspot.com/p/metodos-de-busqueda-y-ejemplos.html

https://www.section.io/engineering-education/understanding-search-algorithms-in-ai/

cesar.sinchiguano@uleam.edu.ec