Artificial intelligence is the science of making programs that can act rationally in an environment. Agents are programs that sense the world and map those to a set of actions. Humans are agents in the sense that they form actions based on their current understanding of the world. While a machine uses sensor to understand its environment and actuators to perform actions. It acts rationally in the sense that it is able to optimize the way, set of actions, towards a goal. [[1]](#footnote-1)

Two types of agents are reflex and goal based. Reflex agents act on the current state of the environment while ignoring the history of what it’s done. Goal based agents act based on information about the ending state. Such as which option brings you closer to the goal.

Search problems utilize goal based agents. A goal is a set of world conditions in which the agent has achieved success. In travelling from Portland to Seattle the start state would be Portland and the goal state would be Seattle. The program will ask after each action “is this Seattle? “ The program will terminate once the goal is reached. This problem can be represented with a graph where each node is a location and arcs represent which cities can be travelled between. Additional information such as the length between cities and the direct distance to the goal can be considered. An uninformed search does not take this information into account while a heuristic search does. A well-formed problem has the following properties. The state space consists of the world state and search state. The world state is every object in the world even the irrelevant ones. The search state is all of the information needed for planning. In order to get from the current state to the next state a successor function is used. It is the description of the possible actions and costs of what can be done. Each Path has a corresponding cost function, whether it is based on distance or moves.

A common problem is pathing, moving from one place to another while avoiding obstacles. The state space is the location of the agent, the actions available are moving in a direction. The successor function gives a new location next to the current one. The goal state is a pre-specified end.

The algorithm spends time expanding nodes and looking for the goal state. Once the goal state has been taken out of the fringe the path from start to goal must be shown. This path will follow the same search strategy as the algorithm and will be chosen from the states expanded.

A modern Approach to Artificial intelligence Chapter 2 by Stuart Russel and Peter Norvig.

In travelling from Portland to Seattle the state space is the five cities, the successor function is driving from one to the other. The start state is Portland and the goal state is Seattle. Each state is a node and the successor function is the arcs between them which may have costs. It is useful to think of a search tree that works over the state space. From the current node the nodes available to travel to are grouped together. The algorithm used decides the search strategy to move through this group.[[2]](#footnote-2)

Depth first search chooses based on which node has the largest depth, farthest from the root. Breadth first search expands all the nodes from the start node first. Then all the nodes from those. It moves through the tree by levels. For these examples ties will be broken by alphabetical order. [[3]](#footnote-3)

Depth first search will go:

Expanded: [Portland] fringe: [Astoria, Tacoma, St Helens]

Expanded: [Portland, Astoria] , fringe: [Seattle, Tacoma, St Helens]

Expanded: [Portland, Astoria, Seattle], fringe [Tacoma, St Helens]

Path returned: [ Portland, Astoria, Seattle],

Breadth first search will go:

2 https://www.cs.uoregon.edu/Classes/16F/cis471/slides/

3 A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig.

Expanded: [Portland] fringe: [Astoria, Tacoma, St Helens]

Expanded: [Portland, Astoria] , fringe: [Tacoma, St Helens]

Expanded:[Portland, Astoria, St Helens], fringe [Tacoma, Seattle]

Expanded:[Portland, Astoria, St Helens, Tacoma], fringe [Seattle]

Expanded:[Portland, Astoria, St Helens, Tacoma, Seattle], fringe:[]

Path returned: [Portland, Astoria, St Helens, Tacoma ,Seattle],

A variant of these search algorithms which includes the weights of the edges is uniform cost search. This search strategy works by choosing the path with the least cumulative cost. Here is the example again but with path weights which are rough distances. [[4]](#footnote-4)

140

150

50

150

30

100

The order of the states expanded under Uniform cost search are:

Expanded: [Portland] fringe: [Astoria, Tacoma, St Helens]

Expanded:[Portland, St Helens] , fringe: [Tacoma, Astoria, Seattle]

4 A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig.

Expanded:[Portland, Astoria, St Helens], fringe [Tacoma, Seattle]

Expanded:[Portland ,Astoria, St Helens, Tacoma], fringe [Seattle]

Expanded:[Portland, Astoria, St Helens, Tacoma, Seattle], fringe:[]

Path returned : [Portland , St Helens, Seattle]

Searches can also include a heuristic which is information about the node related to the goal. Distances from the goal are common heuristics. For the current example driving time will be the heuristic. Here is the graph again with the heuristic included. [[5]](#footnote-5)

140

30

150

100

150

50

Greedy search will consider only the path which has the smallest heuristic[[6]](#footnote-6). Here the order is:

Expanded: [Portland] fringe: [Astoria, Tacoma, St Helens]

Expanded: [Portland, Tacoma] fringe: St Helens, Astoria, Seattle]

5A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig.

6 A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig.

Expanded: [Portland, Tacoma, Seattle], fringe [Astoria, St Helens,.]

Path returned: [Portland Tacoma, Seattle]

A more well -rounded search ,A\*, chooses the sum of the least cumulative path cost and the heuristic. To the right of the fringe states are the calculations for each state. [[7]](#footnote-7)

Expanded: [Portland] fringe: [Astoria, Tacoma, St Helens] Astoria= 100+180, Tacoma = 150+40, St Helens= 150+30.

Expanded:[Portland, St Helens] , fringe:[ Astoria, Tacoma, Seattle] Astoria= 100+180, Tacoma = 150+40, Seattle=170+0

Expanded:[Portland, St Helens, Seattle], fringe [Astoria, Tacoma,.]

Path Returned: [Portland, St Helens, Seattle]

7 A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig.

1. A modern Approach to Artificial intelligence Chapter 2 by Stuart Russel and Peter Norvig. [↑](#footnote-ref-1)
2. https://www.cs.uoregon.edu/Classes/16F/cis471/slides/ [↑](#footnote-ref-2)
3. A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig. [↑](#footnote-ref-3)
4. A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig. [↑](#footnote-ref-4)
5. A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig. [↑](#footnote-ref-5)
6. A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig. [↑](#footnote-ref-6)
7. A modern Approach to Artificial intelligence Chapter 3 by Stuart Russel and Peter Norvig. [↑](#footnote-ref-7)