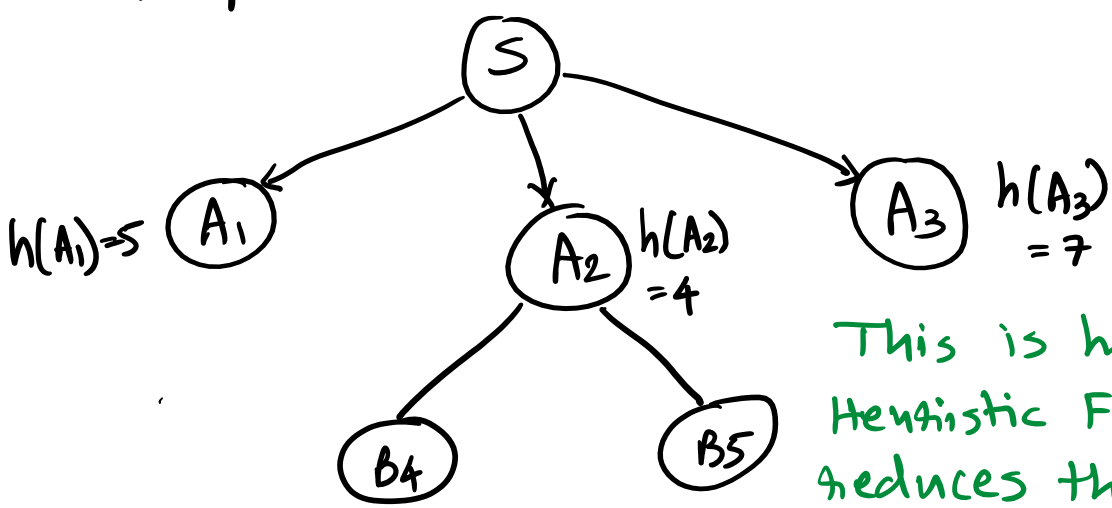
Informed Search - Heuristic Functions

Sunday, September 15, 2024 1:28 PM

☆ Heuristic meaning: "To Find Out / To explore"

☆ Goal of Heuristics in AI search: To reduce the state space by exploiting the 'knowledge' of the problem domain

☆ Heuristic Function: A Heuristic Function assigns an 'estimated cost' to each child node which is a 'guess' of cost to reach to the goal state from that particular node.



This is how a Heuristic Function reduces the state space.

☆ Problem Relaxation: Compromising the rules to solve the problem in a 'Relaxed' manner.

e.g. :- 8 puzzle game:

e.g. :- 8 puzzle game:

Start State				Goal State		
7	2	6			1	2
5		4		3	4	5
8	3	1		6	7	8

As per the rules, we can move 5→, 2↓, 6← or 3↑ only in the start state. But in a relaxed problem, we can think of moving any number anywhere.

Example of a Heuristic Function in 8-puzzle game:

1 Number of mismatched | misplaced tiles

How many moves will be required to reach from the start state to the goal state in a Relaxed 8-puzzle game in the following example?

Start State				Goal State		
7	2	6			1	2
5		4		3	4	5
8	3	1		6	7	8

Ans = 8 moves

7	2	6
5		4
8	3	1

7		6
5	2	4
8	3	1

cost = 8

7	2	6
	5	4
8	3	1

cost = 8

7	2	6
5	4	
8	3	1

cost = 7

7	2	6
5	3	4
8		1

cost = 8

This cost is our Heuristic Function!

$\boxed{2}$ Manhattan Distance

Suppose we assume another approach for problem relaxation. This time we will not allow the 'tiles' to float in any direction but they can still float in either horizontal or verticle direction only.

Start State

7	2	6
5		4
8	3	1

Goal State

	1	2
3	4	5
6	7	8

Now let's calculate cost of start state with this new method.

Tile to move: 1 2 3 4 5 6 7 8

cost: 3 + 1 + 2 + 1 + 2 + 4 + 3 + 2

Total cost = 18

7	2	6
5		4
8	3	1

7		6
5	2	4
8	3	1

7	2	6
	5	4
8	3	1

7	2	6
5	4	
8	3	1

7	2	6
5	3	4
8		1

Manhattan cost:

3 + 1 + 4 + 2 + 0 + 2 + 2 + 3 = 17

☆ Admissible Heuristic Function

→ A Heuristic Function that always under estimates the cost of any node to reach to goal position is an Admissible Heuristic Function.

→ As a result, the cost predicted by an Admissible Heuristic Function will always less than or equal to the actual cost to reach to goal or equal to the actual cost to reach to goal state from that node.

$$h(n) \leq h^*(n)$$

$h(n)$ = cost predicted by the Heuristic Function

$h^*(n)$ = actual cost.

→ They are also called "Optimistic" H.F.

→ Disadvantage: They may slow down our search by assigning lower cost to a node.

☆ Inadmissible Heuristic Functions:

→ These H.F. never under estimates the cost (They over estimates the cost).

→ As a result cost predicted by an Inadmissible H.F. will be always greater than or equal to the actual cost.

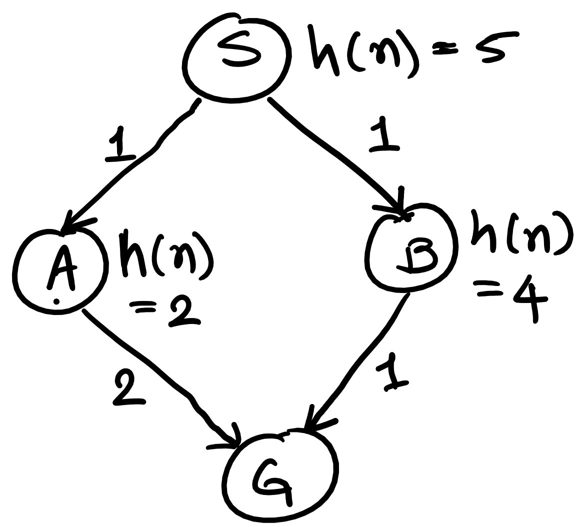
$$h(n) \geq h^*(n)$$

→ They are also known as "Pessimistic" H.F.

→ Disadvantage: They may block the optimal

→ Disadvantage: They may block the optimal path by assigning higher cost to a node.

Example: Suppose we are using an Inadmissible H.F. that estimates the following graph:



☆ Dominance:

Suppose we have two Admissible H.F. h_1 & h_2 and if $h_2(n) \geq h_1(n)$ for all n then h_2 is better than h_1.

This is also said as "h_2 dominates h_1"

∴ In our 8-puzzle example the Heuristic Function Manhattan distance is better (dominates) the Heuristic Function No. of misplaced tiles because

① Both of them are admissible &

② Manhattan distance estimates higher cost for all the nodes than no. of misplaced tiles.

☆ Composite Heuristic Function:

Suppose we have two H.F. and for some nodes h_1 gives higher cost and for others h_2 gives higher cost and both of them are admissible then $\max(h_1, h_2)$ is also admissible.

$\therefore \max(h_1, h_2, \ldots, h_n)$ is called a composite Heuristic Function.

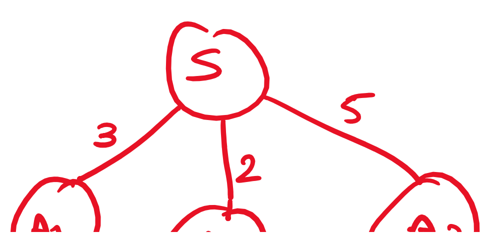
Composite Heuristics take more computation time but they are also more accurate

☆ An overview of g-value, h-value & f-value

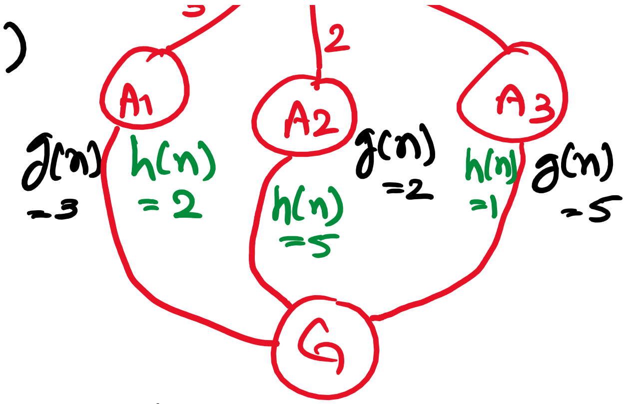
h-value : cost to reach to goal state from the given
(h(n)) node predicted by Heuristic function.

g-value : Actual cost to reach to node 'n' from
(g(n)) the start state.

f-value : $f(n) = g(n) + h(n)$



f-value : $f(n) = g(n) + h(n)$



☆ Consistent Heuristic Functions

→ An admissible H.F. in which f-value never decreases along any path towards goal state

→ Although not all the admissible H.F. are consistent but most of them are.