CS561 - ARTIFICIAL INTELLIGENCE LAB

ASSIGNMENT-2: Hill Climbing and Simulated Annealing

Group Name: **1801cs31_1801cs32_1801cs33**

Students:

Names	Roll No.	Batch
M **Maheeth Reddy**	1801CS31	B.Tech.
M **Nitesh Reddy**	1801CS32	B.Tech.
Nischal A	1801CS33	B.Tech.

Question 2. Simulated Annealing

Subquestion e.

i. Check whether the heuristics are admissible.

Answer:

The heuristic h1(n) = Number of displaced tiles, **is always admissible**.

Reason: In the 8-puzzle problem, each displaced tile must be moved at least once to reach the goal state. So, the total number of moves to order the tiles correctly, or the cost to reach the goal state will be greater than or equal to the number of displaced tiles. Since, this heuristic is not overestimating the cost of reaching the goal state, it is admissible.

The heuristic h2(n) = Total Manhattan distance, **is always admissible**.

Reason: Admissible heuristics must not overestimate the number of moves to solve this problem. Since we can only move one block at a time and in only one of the four directions. The optimal scenario for each block is that it has a clear, unobstructed path to its goal state. This is a Manhattan Distance of 1. The rest of the states for a pair of blocks is sub-optimal, meaning it will take more moves than the Manhattan Distance to get the block in the right place. Thus, this heuristic does not overestimate the cost of reaching the goal state. Therefore it is admissible.

ii. What happens if we make a new heuristics h3 (n) = h1 (n) * h2 (n)

Answer:

The newly created heuristic h3(n) may or may not be admissible.

Reason: A heuristic h is admissible if h(n) <= h*(n) where h*(n) is the true cost to a nearest goal. We have checked that h1(n) and h2(n) are admissible in the previous answer. So, h1(n) <= h*(n) and h2(n) <= h*(n).

Now, h3(n) = h1(n) * h2(n) does not guarantee that h3(n) <= h*(n). So, the admissibility of the heuristic h3(n) cannot be deduced.

iii. What happens if you consider the blank tile as another tile?

Answer:

The value of the heuristics may increase as we will now consider the heuristic values of all tiles without excluding any. This may have an effect on the admissibility of the heuristic.

iv. What if the search algorithm got stuck into Local optimum? Is there any way to get out of this?

Answer:

In the case of Simulated Annealing, we can sometimes get out of the Local Optimum by accepting candidates with a higher cost to escape from the local optimum.

For Hill Climbing, we could use Random-walk hill-climbing, Random-restart hill-climbing to avoid getting stuck in the Local Optimum.