**Question #1**

Provide an example of boards in which beam search with two start points does not give the same result as hill climbing with a single restart.

The local beam search algorithm begins with *k* randomly generated states. At each step, all successors of the *k* states are generated. The *k* successors with the best score are passed to the next round and the process is repeated. Unlike with the Hill Climbing algorithm (both without and without restart), local beam search allows sharing of successor quality information between the *k* states.

**Example Overview**

**Heuristic Cost Function (*h*):** The number of pairs of attacking queens. In this case, we are trying to minimize the heuristic function (i.e. ). This is not hill climbing in the strictest sense of the interpretation. If we wanted to maximize a parameter, we could use heuristic, , where is defined as:

The summation:

is used because it represents the maximum number of possible collision combinations:

()

You would then maximize to 28. For simplicity, I will discuss below minimize to zero due to its greater simplicity. However, the concept is identical regardless of which of the two approaches are used.

**Queen Symbol:** ♠

**Board Overview**

**Board # 1:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** |
| **8** |  |  |  |  |  |  | ♠ |  |
| **7** |  |  |  |  | ♠ |  |  |  |
| **6** |  | ♠ |  |  |  |  |  |  |
| **5** |  |  |  | ♠ |  |  |  |  |
| **4** |  |  |  |  |  | ♠ |  |  |
| **3** |  |  |  |  |  |  |  | ♠ |
| **2** |  |  | ♠ |  |  |  |  |  |
| **1** | ♠ |  |  |  |  |  |  |  |

Board # – Initial Board for the Hill Climbing Algorithm and Initial State #1 for Local Beam Search.

**Description:** This board comes from the textbook *Artificial Intelligence: A Modern Approach* by Russell and Norvig (see page 123 in the third edition).

**Heuristic Value of this Board:**

**List of Pairs of Attacking Queens:**

(D5, G8)

**Importance of this Board:**

Per Russell and Norvig, this board is a local minima. Any movement of a queen results in a higher heuristic function. Hence, if Hill climbing was run on this board, it would immediately terminate since it is at a local minimum.

**:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** |
| **8** | 18 | 12 | 14 | 13 | 13 | 12 | 14 | 14 |
| **7** | 14 | 16 | 13 | 15 | 12 | 14 | 12 | 16 |
| **6** | 14 | 12 | 18 | 13 | 15 | 12 | 14 | 14 |
| **5** | 15 | 14 | 14 | ♠ | 13 | 16 | 13 | 16 |
| **4** | ♠ | 14 | 17 | 15 | ♠ | 14 | 16 | 16 |
| **3** | 17 | ♠ | 16 | 18 | 15 | ♠ | 15 | ♠ |
| **2** | 18 | 14 | ♠ | 15 | 15 | 14 | ♠ | 16 |
| **1** | 14 | 14 | 13 | 17 | 12 | 14 | 12 | 18 |

Board # – Restart Board for the Hill Climbing Algorithm and Initial State #2 for Local Beam Search.

**Description:** This board also comes from the textbook *Artificial Intelligence: A Modern Approach* by Russell and Norvig (see page 123 in the third edition). The eight queens are shown on the board. The cells not populated with queens have numbers describing the heuristic cost if a queen in that column was moved into that space. For example, if the queen in cell B4 was move to B8, the heuristic cost would be 12.

**Heuristic Value of this Board:**

**List of Pairs of Attacking Queens:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| (A4, B3) | (A4, C2) | (A4, E4) | (B3, C2) | (B3, D5) | (B3, F3) | (B3, H3) | (C2, E4) | (C2, G2) |
| (D5, E4) | (D5, F3) | (D5, G2) | (E4, F3) | (E4, G2) | (F3, G2) | (F3, H3) | (G2, H3) |  |

**Importance of this Board:**

Per Russell and Norvig, the minimum heuristic cost by moving one queen within its column is 12 (see cells: B8, B6, E7, E1, F8, F6, G7, and G1).

**Board Traversal Using Hill Climbing**

**Step #1:** The hill climbing algorithm examines all possible successors of . Per Russell and Norvig, no successors have lower heuristic costs. Hence, the algorithm terminates. Since it is not a goal state, it generates another random board (in this case ).

**Step #2:** The hill climbing algorithm examines and observes the eight successor states that have identical minimum value (i.e. B8, B6, E7, E1, F8, F6, G7, and G1). The algorithm chooses **only one** of the successors and then generates subsequent successors until a local minimum is found or the goal is reached.

**Conclusion:** The hill climbing algorithm did not investigate any possible solutions using even though it was much closer to the board than . Moreover, it only ever followed one path through the board at a time.

**Board Traversal Using Local Beam Search**

**Step #1:** The local beam algorithm generates all possible successors for and . Per Russell and Norvig, no successors have lower heuristic costs. Hence, the algorithm terminates. Since it is not a goal state, it generates another random board (in this case).

**Step #2:** The hill climbing algorithm examines and observes the eight successor states that have identical minimum value (i.e. B8, B6, E7, E1, F8, F6, G7, and G1). The algorithm chooses **only one** of the successors and then generates subsequent successors until a local minimum is found or the goal is reached.