CSI 5137 AI enabled software verification and testing

Assignment 1 – Traveling salesman using heuristic search algorithms (Hill climbing and random search)

Divya Reddy – 300290332
Maheedhar Vundela – 300282606

Objective

The objective of this report is to summarize the implementation of a solver for the Traveling Salesman problem using Hill Climbing, which is a metaheuristic search algorithm. The report contains an overview of the traveling salesman issue, a hill climbing algorithm, an explanation of our method, and a comparison of our approach to the Random search algorithm.

TSP - Traveling Salesman Problem

The traveling salesman problem (TSP) is an algorithmic problem that seeks the shortest path between a set of points and locations that must be visited. The cities a salesman could visit are represented by the points in the problem statement. The salesman's objective is to minimize both travel costs and distance traveled. It is an NP hard problem. TSP can be symmetric or asymmetric. In this report, we only look at the symmetric traveling salesman problem, which means that the distance between two cities is the same in both directions.

Algorithm:

- **Random search algorithm:** Within the given budget, the algorithm repeatedly generates a random solution, compares its fitness to the known best, and keeps the best one. Random search is effective when the underlying problem does not give any guidance to begin with. First, random search should always be the default sanity check against our own search methodology to compare if our algorithm generates better results than the results generated by random search.

- **Hill climbing algorithm:** Hill climbing algorithm is a technique which is used for optimizing mathematical problems. Hill climbing is a heuristic search algorithm which continuously moves in an elevated direction to reach the peak of the hill(best solution). A node of hill climbing algorithm has two components which are state and value. It

evaluates the neighbor node state and selects the state which optimizes the current cost and sets it as the current state.

Features of hill climbing algorithm:

- greedy approach
- no backtracking
- feedback mechanism
- incremental change

We chose this approach because it is frequently utilized for optimization issues in the field of AI and has been widely used to solve the traveling salesman problem. Furthermore, this method is simpler and capable of producing good results.

Approach

1. current_solution = generate initial solution randomly

2. Repeat:

    2.1 generate neighbor solutions (differ from current solution by a single element)

    2.2 best_neighbour = get highest quality neighbor of current_solution

    2.3 If quality(best_neighbour) <= quality(current_solution)

        2.3.1 Return current_solution

    2.4 current_solution = best_neighbour

Representation,Fitness and Operator

1) **Representation:**
    - First, let's code an instantiation of the traveling salesman problem and instantiation should be a list of cities.

- Create a **random solution generator**. Let's create a list of identifiers of all cities, and from there iteratively pick a city from that list at random and add it to the solution.
- Create a function calculating the **route length.**
- Create a function which generates all **neighbors** of a solution.
- Create a function to find the **best neighbor.** Choose the best neighbor amongst all neighbors.
- Now Implement the hill climbing algorithm(core). We first create a random solution and determine the length of its route. The best option is then chosen after creating all neighboring solutions. We then repeat the same pattern, updating the existing solution each time with the best neighbor, as long as the best neighbor is superior to the current solution. We return the current solution after this step is finished (and its route length).

2) Operator:

- Conduct an assessment of the current state. Stop the process and indicate success if it is a goal state(better than the current state).
- The iterator i "visits" each city, thus when i is equal to 0, i-1 is "at" the city that was visited before or after (which is exactly what we want, since we want to end up at the first city again). As a result, solution[i] provides us the present city, whereas solution[i-1] gives us the past city. The distance between these cities is then easily calculated using the tsp, and the overall length of the route is then increased accordingly (routeLength).
- Iterate over all neighbors and when a neighbor has a shorter route length, both the best neighbor and best Route length are updated.
- We repeat the same pattern with the current solution being updated each time with the best neighbor as long as the best neighbor is superior to the existing solution. We return the current solution after this step is finished (and its route length).

3) Fitness:

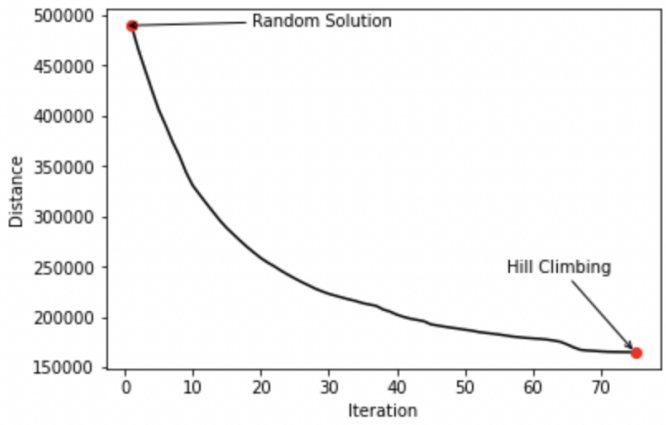
- The fitness function demonstrated the total distance that a salesperson had to travel given the current tour (solution), the aim of the algorithm was to minimize this number as far as possible.
- TSP prefers shorter routes. The fitness function is defined as the inverse of the total distance/route length. The greater the value, the better the algorithm.

Results:

We have executed our solution and plotted the distance graphs of the hill climbing approach for each of the datasets below.

1. **pr76.tsp**

```
Path for random solution: [19, 64, 48, 57, 59, 18, 63, 49, 12, 11, 46, 45, 68, 73, 47, 60, 55, 56, 52, 16, 17, 53, 20, 62, 70, 65, 31, 61, 67, 6, 51, 66, 40, 75, 2, 72,
5, 69, 34, 22, 54, 38, 1, 71, 21, 39, 0, 24, 33, 32, 4, 43, 27, 50, 35, 28, 23, 25, 74, 41, 7, 14, 58, 37, 8, 10, 13, 3, 42, 26, 29, 44, 36, 15, 9, 30]
Distance for random solution: 645656.0407144945
Path for hill climbing is [73, 7, 6, 11, 14, 15, 16, 17, 31, 32, 34, 35, 36, 30, 29, 28, 40, 59, 58, 57, 54, 50, 55, 56, 62, 61, 60, 38, 37, 33, 39, 51, 49, 48, 22, 0,
75, 74, 1, 2, 3, 18, 27, 41, 53, 52, 43, 47, 46, 68, 67, 66, 69, 70, 71, 72, 63, 64, 65, 42, 4, 5, 8, 9, 25, 26, 44, 45, 23, 21, 20, 24, 19, 10, 12, 13]
Distance for hill climbing is 165131.92625079548
```

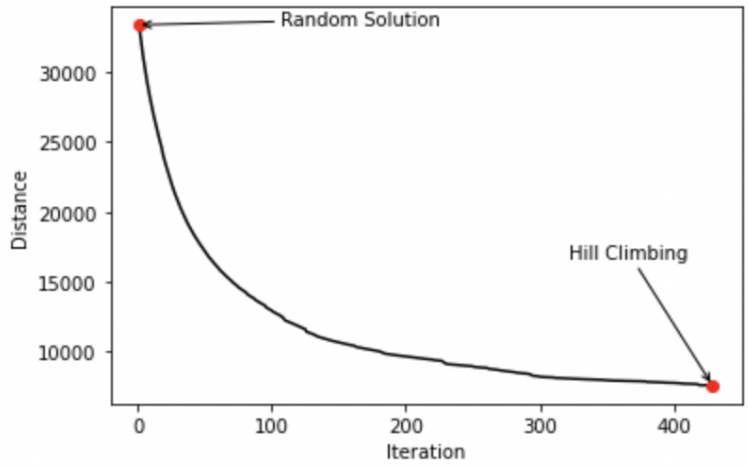


Distance for random solution: 645656.0407144945

Distance for hill climbing is: 165131.92625079548

2. a280.tsp

```
Path for random solution: [234, 178, 179, 4, 5, 66, 130, 268, 82, 90, 36, 201, 115, 87, 30, 167, 219, 21, 86, 169, 38, 111, 118, 189, 61, 163, 182, 153, 223, 37, 240,
28, 193, 203, 75, 198, 184, 249, 144, 79, 253, 266, 17, 258, 0, 62, 96, 250, 228, 136, 256, 13, 139, 126, 73, 231, 275, 261, 188, 229, 233, 141, 114, 224, 194, 101, 209,
197, 125, 47, 49, 123, 83, 24, 210, 238, 2, 230, 272, 222, 109, 277, 124, 215, 175, 207, 242, 71, 269, 104, 176, 56, 235, 20, 68, 166, 22, 108, 157, 190, 95, 142, 227,
159, 34, 43, 105, 192, 63, 217, 168, 92, 16, 19, 74, 50, 225, 89, 133, 88, 183, 138, 98, 152, 181, 102, 25, 172, 134, 59, 185, 72, 162, 145, 150, 237, 121, 41, 248, 81,
279, 260, 158, 1, 26, 211, 173, 177, 264, 220, 112, 99, 42, 148, 239, 93, 271, 100, 84, 259, 246, 156, 57, 106, 143, 202, 55, 78, 94, 23, 270, 252, 221, 132, 164, 32,
12, 161, 33, 216, 14, 199, 140, 8, 213, 180, 244, 70, 146, 18, 85, 206, 10, 107, 265, 91, 39, 151, 160, 7, 245, 251, 51, 76, 155, 131, 67, 165, 77, 103, 263, 241, 48,
191, 119, 186, 254, 129, 69, 137, 97, 117, 154, 218, 120, 31, 35, 187, 147, 171, 276, 127, 9, 3, 174, 52, 122, 247, 65, 278, 226, 128, 208, 214, 116, 64, 243, 53, 204,
80, 205, 274, 232, 46, 255, 27, 195, 110, 40, 200, 196, 236, 6, 267, 54, 29, 262, 170, 257, 149, 212, 60, 15, 58, 113, 11, 273, 135, 44, 45]
Distance for random solution: 35812.372517551354
Path for hill climbing is [193, 191, 190, 186, 184, 180, 151, 128, 153, 154, 152, 118, 116, 115, 114, 206, 228, 231, 232, 226, 225, 212, 203, 143, 142, 145, 146, 141,
147, 120, 40, 46, 53, 54, 61, 160, 161, 171, 170, 169, 168, 102, 103, 108, 111, 110, 162, 163, 165, 166, 167, 100, 75, 73, 65, 63, 39, 38, 37, 35, 224, 223, 222, 221,
60, 43, 56, 57, 64, 84, 82, 87, 107, 104, 106, 159, 158, 175, 176, 157, 156, 150, 177, 149, 148, 138, 248, 247, 0, 4, 5, 10, 11, 12, 14, 273, 246, 239, 238, 237, 236,
230, 229, 112, 86, 83, 72, 71, 70, 69, 52, 51, 50, 49, 31, 27, 24, 22, 23, 13, 271, 272, 260, 249, 235, 234, 233, 227, 267, 125, 29, 124, 123, 181, 182, 183, 144, 198,
197, 200, 199, 201, 202, 215, 214, 213, 210, 211, 91, 95, 94, 77, 76, 74, 85, 117, 122, 126, 127, 19, 132, 269, 268, 133, 130, 36, 48, 47, 62, 81, 80, 88, 109, 179, 178,
208, 250, 245, 244, 243, 240, 242, 241, 1, 279, 2, 261, 58, 67, 68, 66, 113, 216, 218, 217, 209, 251, 6, 7, 9, 8, 274, 259, 258, 257, 255, 254, 256, 262, 18, 28, 30, 32,
33, 34, 42, 59, 119, 155, 137, 264, 263, 277, 278, 3, 276, 275, 270, 15, 45, 55, 44, 41, 121, 136, 266, 265, 139, 140, 204, 205, 207, 252, 253, 21, 25, 26, 20, 129, 185,
189, 188, 187, 164, 101, 90, 79, 78, 89, 105, 172, 173, 174, 135, 134, 16, 17, 131, 92, 93, 96, 97, 98, 99, 192, 196, 195, 219, 220, 194]
Distance for hill climbing is 7572.199803597675
```

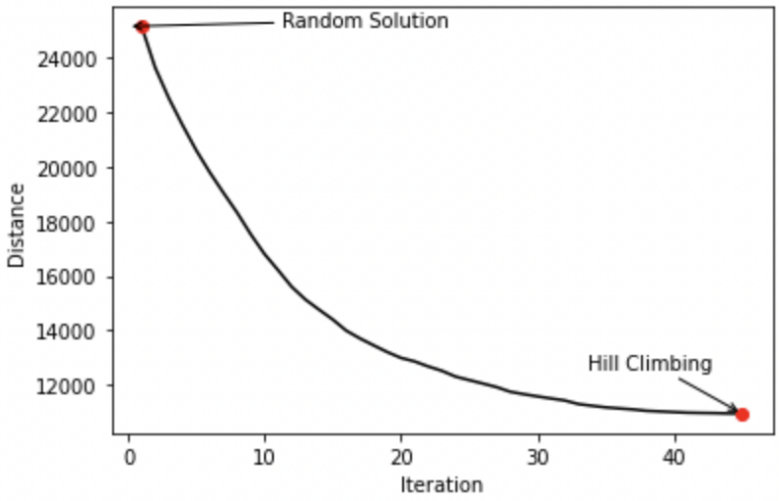


Distance for random solution: 35812.372517551354

Distance for hill climbing is: 7572.199803597675

3. berlin52.tsp

```
Path for random solution: [44, 5, 13, 9, 40, 38, 18, 45, 27, 34, 0, 6, 23, 3, 39, 49, 31, 50, 14, 41, 1, 2, 20, 29, 12, 10, 24, 15, 21, 22, 32, 30, 33, 16, 51, 37, 43,
28, 26, 48, 36, 17, 8, 4, 35, 7, 19, 25, 11, 42, 46, 47]
Distance for random solution: 26943.034337446366
Path for hill climbing is [19, 22, 20, 41, 6, 1, 29, 28, 43, 33, 34, 0, 21, 44, 31, 48, 25, 51, 10, 50, 32, 7, 40, 18, 2, 16, 17, 30, 46, 13, 12, 26, 27, 11, 42, 9, 8,
5, 3, 24, 47, 36, 35, 38, 39, 37, 14, 4, 23, 45, 15, 49]
Distance for hill climbing is 10934.579144069854
```

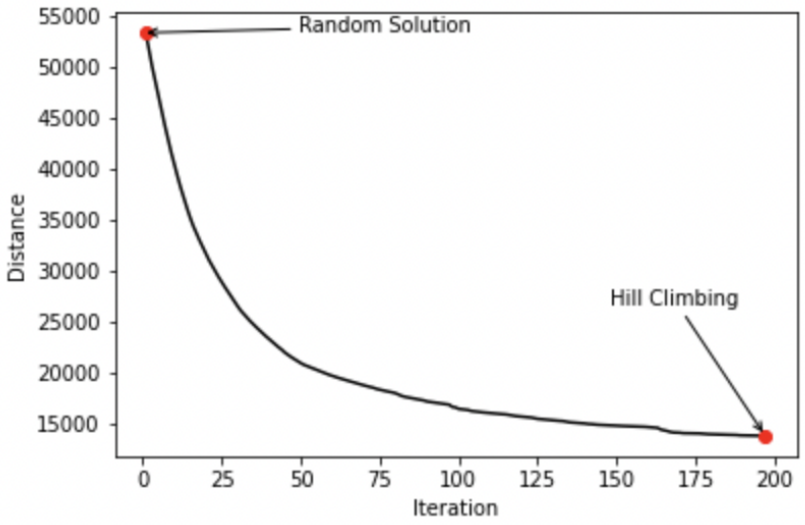


Distance for random solution: 26943.034337446366

Distance for hill climbing is: 10934.579144069854

4. ch150.tsp

Path for random solution: [137, 33, 147, 67, 36, 71, 49, 9, 97, 107, 134, 5, 10, 132, 117, 119, 26, 23, 82, 81, 140, 28, 35, 141, 69, 89, 53, 80, 142, 42, 29, 59, 84, 123, 126, 19, 118, 86, 7, 87, 122, 72, 62, 43, 68, 85, 79, 145, 149, 112, 64, 44, 111, 22, 93, 16, 148, 6, 56, 18, 76, 75, 40, 127, 66, 2, 21, 39, 113, 13, 17, 91, 120, 47, 136, 20, 94, 12, 46, 1, 48, 37, 11, 139, 131, 52, 31, 65, 78, 25, 96, 14, 121, 114, 34, 83, 45, 4, 95, 51, 98, 92, 0, 60, 63, 61, 128, 105, 57, 143, 50, 73, 135, 106, 24, 102, 130, 125, 129, 116, 101, 144, 124, 100, 99, 15, 55, 32, 133, 3, 104, 146, 74, 110, 58, 41, 108, 115, 70, 54, 38, 77, 88, 138, 90, 30, 103, 27, 8, 109]
Distance for random solution: 55314.96696764247
Path for hill climbing is [73, 122, 139, 59, 38, 11, 91, 34, 88, 83, 29, 62, 9, 112, 47, 72, 75, 86, 33, 95, 22, 40, 56, 128, 79, 77, 78, 2, 61, 148, 116, 65, 16, 147, 41, 8, 27, 5, 113, 101, 136, 123, 92, 125, 32, 130, 31, 37, 115, 126, 68, 117, 39, 138, 52, 23, 26, 30, 144, 143, 100, 35, 60, 10, 129, 12, 105, 90, 118, 67, 127, 114, 149, 20, 58, 15, 55, 82, 85, 98, 18, 1, 36, 50, 42, 66, 108, 137, 45, 89, 24, 109, 119, 46, 19, 53, 104, 110, 132, 121, 76, 133, 57, 49, 131, 84, 141, 142, 99, 4, 7, 93, 87, 120, 51, 145, 25, 74, 64, 54, 140, 43, 70, 44, 3, 103, 21, 124, 80, 28, 107, 69, 134, 17, 96, 97, 0, 81, 94, 106, 102, 6, 14, 13, 48, 146, 71, 63, 111, 135]
Distance for hill climbing is 13862.630103842079

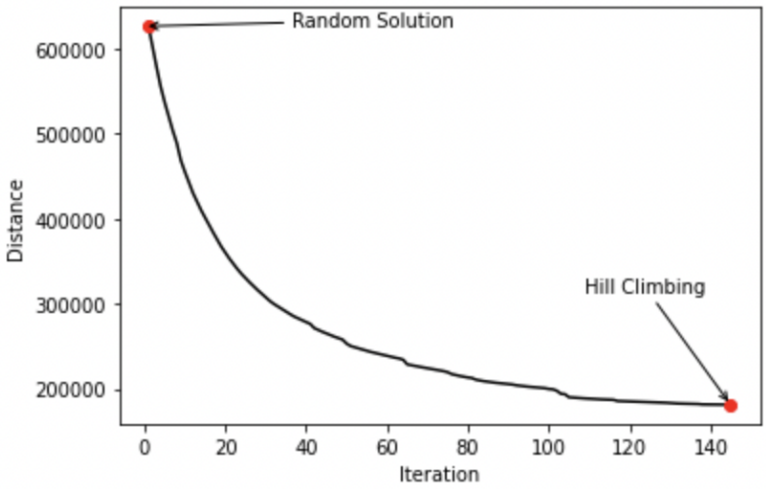


Distance for random solution: 55314.96696764247

Distance for hill climbing is: 13862.630103842079

5. bier127.tsp

```
Path for random solution: [23, 49, 80, 105, 109, 2, 111, 60, 43, 54, 99, 87, 50, 41, 8, 22, 28, 56, 75, 11, 91, 16, 98, 1, 21, 17, 125, 108, 53, 83, 64, 30, 81, 55, 114,
39, 9, 84, 94, 32, 36, 89, 112, 101, 85, 120, 44, 37, 5, 116, 4, 29, 122, 113, 59, 73, 6, 74, 70, 103, 52, 118, 12, 69, 58, 62, 46, 20, 86, 26, 34, 45, 65, 117, 97, 40,
33, 106, 57, 7, 19, 79, 110, 0, 68, 24, 61, 66, 15, 27, 38, 126, 63, 78, 107, 35, 25, 48, 31, 88, 72, 67, 10, 51, 124, 14, 92, 119, 121, 104, 115, 95, 82, 96, 47, 102,
76, 90, 77, 18, 71, 3, 13, 123, 42, 100, 93]
Distance for random solution: 600460.669402203
Path for hill climbing is [82, 100, 101, 62, 118, 95, 108, 87, 86, 109, 70, 58, 66, 7, 71, 20, 16, 21, 22, 23, 8, 115, 89, 57, 63, 99, 51, 55, 4, 119, 113, 5, 105, 56,
53, 52, 117, 48, 46, 49, 104, 14, 36, 35, 33, 37, 27, 121, 29, 13, 11, 19, 107, 3, 18, 17, 76, 78, 30, 40, 34, 39, 42, 26, 75, 67, 69, 68, 74, 77, 79, 24, 28, 31, 32,
25, 10, 124, 88, 91, 98, 64, 54, 45, 93, 92, 126, 94, 102, 44, 43, 2, 59, 61, 60, 90, 9, 114, 12, 6, 0, 15, 1, 50, 120, 123, 65, 112, 103, 85, 84, 72, 73, 38, 41, 122,
96, 97, 106, 110, 111, 47, 116, 83, 80, 125, 81]
Distance for hill climbing is 181040.64722284582
```



Distance for random solution: 600460.669402203

Distance for hill climbing is: 181040.64722284582

Inference:

On the aforementioned datasets, we also evaluated the overall distance for hill climbing vs random search. For each dataset, hill climbing produced a better solution (a shorter overall route length) to the traveling salesman problem.

Conclusion:

We have used a meta-heuristic approach(ie; Hill Climbing) for solving symmetric TSP. Based on our results and analysis, Hill Climbing has emerged out to be a better solution when compared to random search. This is due to its ability to exploit and explore the search space and the less number of parameters required.

References:

1. https://towardsdatascience.com/how-to-implement-the-hill-climbing-algorithm-in-python-1c65c29469de
2. https://www.javatpoint.com/hill-climbing-algorithm-in-ai
3. https://classes.engr.oregonstate.edu/mime/fall2017/rob537/hw_samples/hw2_sample2.pdf
4. https://www.section.io/engineering-education/understanding-hill-climbing-in-ai/