Sharon Shin

December 2nd, 2021

ICS 340

**Deliverable D Report**

*Effectiveness of Genetic Algorithm on Traveling Salesman Problem:*

Due to the random nature of a genetic algorithm, without creating an algorithm in which many iterations are considered, we cannot guarantee an effective and efficient solution to the TSP. The reason I chose this algorithm is because it is interesting to me, and I wanted to learn more about it. With a deeper understanding of machine learning and AI (Artificial Intelligence), I wanted to take this approach would help me understand those topics a little more.

*How will you pick a starting tour?*  
The starting tour is chosen by the default input graph and how it is read in. From this array list of nodes, we will make randomized tours using a randomized number generator to randomly pick numbers from the original list of nodes to populate the first generation. From there, the initial population will breed with each other by splicing the first parent’s array list of nodes and adding the remaining nodes from the second parent in the order the second parent’s tour is for a randomized number of iterations. The total amount of generations can range from 6-11. There is also a small chance of mutations occurring based on a counter that increments and a modulo function.

*What will you do to improve the tour?*

Because this is a random algorithm, there is no guarantee that the tours will improve. However, I have implemented a couple of ways to make sure that the tours will be improved for the most part. First, the population of tours will be sorted from shortest distance to highest distance and the top 25% of the tours with the shortest distances will secure a spot in the next generation. Also the ith + (i+1)th tours will breed, ensuring that the best tours will breed with each other to potentially have an improved child. To avoid the offspring from becoming too close to each other, the mutations will happen at random times to switch up the gene pool and allow for more potential tours. Finally, a master array list will keep track of every tour we have tried so far to avoid repeated tours and increase our chances for better tours.

*What are the criteria for stopping?*

Because this algorithm will not find the best tour, or may not even come close, a random number generator will decide how many iterations to do from 6-11.