**Traveling Salesperson Problem –An Evolutionary Algorithm Approach with Wisdom of Crowds**

Will B. Crawford

CECS

Speed School of Engineering

University of Louisville, USA

[wbcraw02@louisville.edu](mailto:wbcraw02@louisville.edu)

1. **Introduction**

Genetic Algorithms, or GA’s, allow for the solution of TSP problems to be computed in significantly less time than another approach. This approach encodes cities as chromosomes and applies evolutionary genetic principles to evolve the chromosomes to a predefined minimum that is produced through a fitness function. Applying this to a population of individuals, one can evolve a population to create the shortest path to solve a TSP problem. However, depending on the implementation of the crossover and mutation methods, the population may take too long to come to a solution. So, in order to solve this issue, a Wisdom of Crowds or WOC approach which can speed up the time needed to compute a solution using an evolutionary genetic algorithm was used. This allows for the aggregation of the top solutions to calculate a theoretical better solution.

1. **Approach**

The genetic algorithm uses a Darwinian, or survival of the fittest. approach to bring population members from one generation to another. The mutation method uses a solution like 2-opt which will help the issue of the final path having a cross inside the path. In order to implement Wisdom of Crowds, an aggregation method was used to determine the final solution. The method chosen will create an *n* × *n* array where n is the total number of cities. Then, while iterating over the top 50% of population, which are the experts of the problem, each pair of cities is increased by one. This is called the agreement matrix, then the agreement matrix is used to compute the best path of the cities. To do this, the pair of cities that has the highest agreement is used to start the path and then built from that initial pair of cities, the next city is found by finding the highest agreed upon city that has not already been added to the path. This is done at each generation so that the genetic algorithm can be stopped quicker which will result in very quick computation times relative to a genetic algorithm.

1. **Results**
   1. **Data** (Describe the data you used.)

The data used was generated by a tool that randomizes Travelling Salesperson Problem nodes, the tool can be found in the references. In the following section, there are the outputs to all the given TSP files. The graphs were generated by inserting the relevant data into a csv file and then analysis was performed. The files are included for further analysis. The routes of the best individual as well as the individual who used WOC to find the path are included in the attached csv files.

* 1. **Results** (Numerical results and any figures or tables.)

*Figure 1: Output of GA Using Random11.tsp as input*

*Figure 2: Output of GA Using Random22.tsp as input*

*Figure 3: Output of GA Using Random44.tsp as input*

*Figure 4: Output of GA Using Random77.tsp as input*

*Figure 5: Output of GA Using Random97.tsp as input*

*Figure 6: Output of GA Using Random222.tsp as input*

1. **Discussion**

During all of the tests, the genetic algorithm had very good results especially due to the 2-Opt mutation method that was used. This method greatly tightened the fitness curves, so that the population does have diversity, but it also means that the population as a whole is improving towards the result. But, the most surprising result is the fitness of the route determined from the wisdom of crowds. This final result represents the best possible solution from every expert in the population. In all of the tests except for the last one, the final WOC result was the most optimal. In figure 4, the WOC member was stagnating, if however, the experiment was left to run longer, the WOC population would eventually come to the optimal. Using the aggregation method described above along with the 2-opt mutation method resulted in very good results and a close to optimal path in a very short time. Therefore, for larger problems beyond 222 cities a hybrid algorithm of 2-opt can be used to speed up the tests and allow the algorithm to be run for longer. Therefore, due to the nature of 2-Opt, the size of the problem mattered greatly for the speed at which the solution was found. However, if the algorithm were to be run on a computer with more resources, the output of the results would be found at a quicker rate. In general, in all of tests ran, the GA with WOC performed significantly better because of the speed at which the solution was found. The GA needs to evolve the population to the solution whereas the WOC implementation will take many solutions and take the average of all of them into account. This means that WOC should be implemented whenever possible with a GA in order to reduce the time needed to come up with a result.

1. **References**

**-** <https://www.tsp.gatech.edu/concorde/index.html>