Assignment 4

CS 776: Evolutionary Computing

(Fall 2016)

Submitted by,

Jose Thomas Painumkal

[Email: josepainumkal@gmail.com]

Assignment 4 - The Travelling Salesperson

Technique and deviations from the canonical genetic algorithm:

Fitness Function: Since TSP is a minimization problem, I defined the fitness function F(x) as 1/f(x) where f(x) is the objective function. In this problem, the objective function f(x) calculates the cost of the tour represented by an individual (i.e. the string of cities).

Selection: For selection, I used Truncation selection [1]. First I sorted the entire population in the decreasing order of their fitness levels. After the sorting, the first half (50%) of the population contains individuals with the best fitness levels. Then two random individuals were picked from the first half of the population to perform crossover/mutation to produce the offspring. This process is repeated to generate the entire offspring.

Crossover: I used Partially Mapped Crossover [2] to implement my GA. Partially Mapped Crossover was suggested by Goldberg and Lingle (1985). In PMX, a portion of one parent's string is mapped onto a portion of the other parent's string and the remaining information is exchanged. [3]

Mutation: Instead of normal mutation (swapping two random cities), I used Reverse Sequence Mutation (RSM) [4]. In RSM, two cities are selected at random and reversed the order of the cities in between them. I noticed that this mutation technique helps in bringing more variations to my generations and my GA converges faster than using normal mutation.

I analyzed the quality, reliability and speed of my GA. The results of the analysis and findings are given below in Table 1.

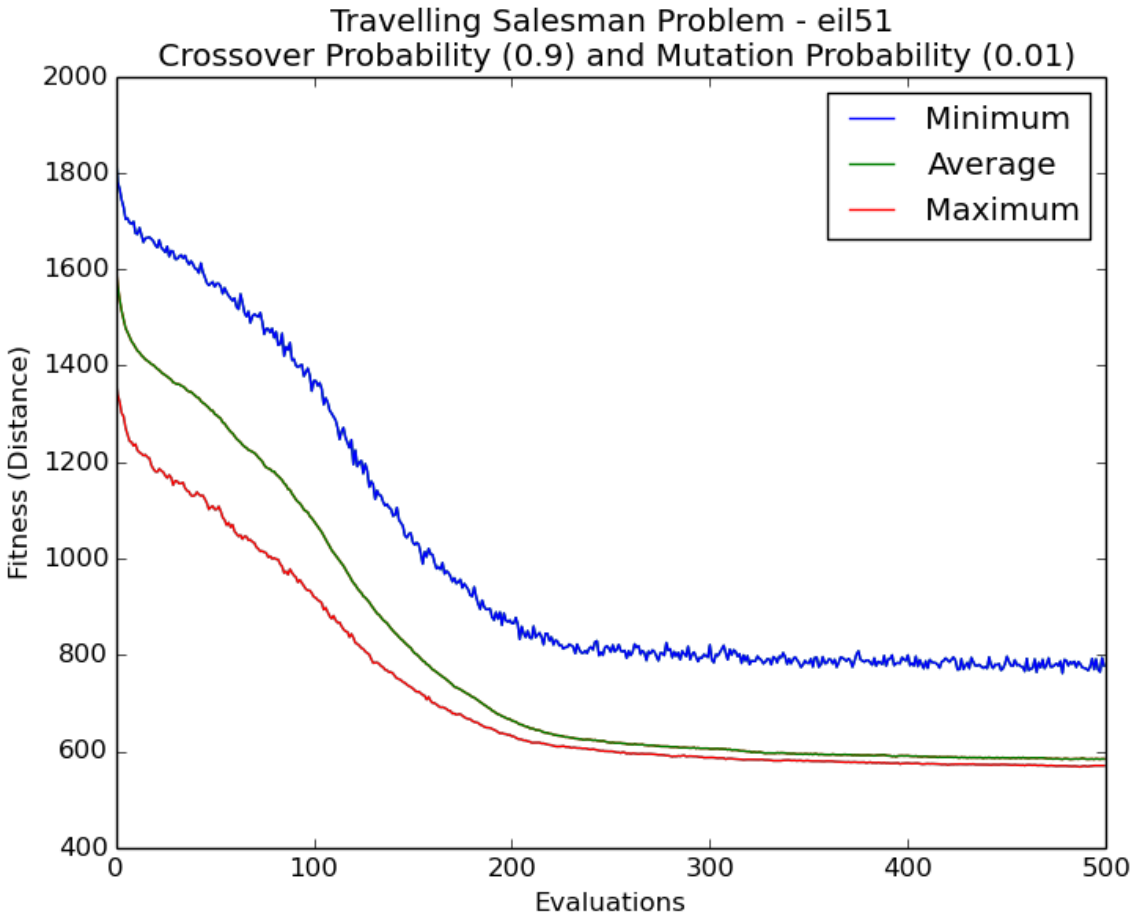
	Eil51	**Berlin52**
Optimum Distance	426	7542
Best Distance of my GA	555	9964
Average Best Distance	561.63	9984.28
Quality (% distance from Optimum)	1.35	24.42
Reliability (% of runs you get within Quality)	46.66	56.66
Speed (Average number of evaluations needed to get within Quality)	140.93	146.6

Table 1. Results of my GA for scenarios, eil51 and berlin52

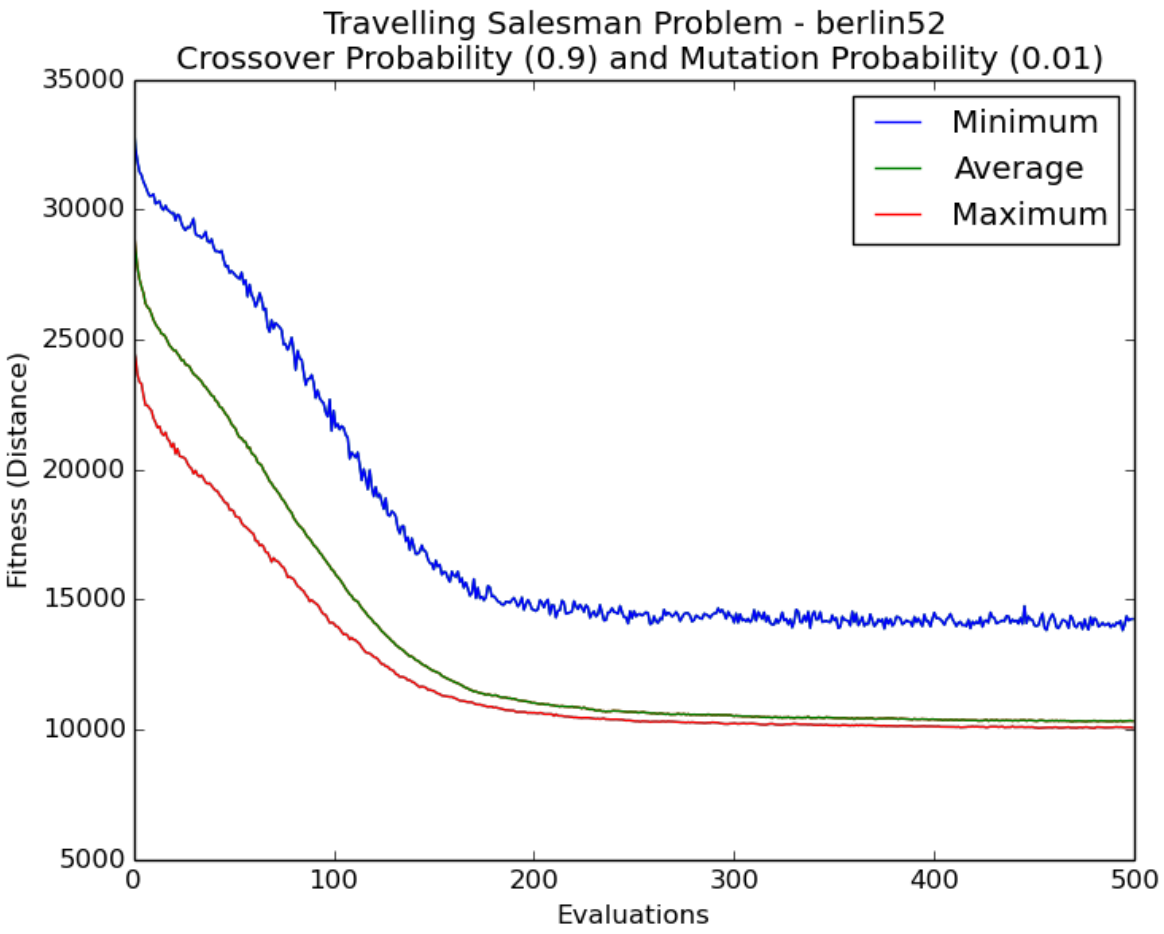
The population size used was 200 and number of generations were 500. The crossover probability ratio was 0.9 and mutation rate was 0.01. The GA was executed up to 30 runs. For eil51, my GA came within 1.35% of the optimal distance, whereas for berlin52, it was 24.42% of the optimal distance. I noticed that the number of generations determined the success of the genetic reproduction. The population become more and more fit with each generation due to effective selection and crossover techniques implemented in the GA. The results show that it is possible to achieve optimal distance by running over more generations and with increased computational power.

Graphs

The graph of average minimum, average-average and average maximum fitness for eil51 is given below.



The graph of average minimum, average-average and average maximum fitness for berlin52 is given below.



References:

[1] Evolutionary Algorithms 3 Selection", Geatbx.com, 2016. [Online]. Available: http://www.geatbx.com/docu/algindex-02.html#P350_17595. [Accessed: 12- Oct- 2016].
[2]"PMX Crossover Operator Tutorial", Rubicite.com, 2016. [Online]. Available: http://www.rubicite.com/Tutorials/GeneticAlgorithms/CrossoverOperators/PMXCrossoverOperator.aspx. [Accessed: 12- Oct- 2016].
[3] P. Larranaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, S. Dizdarevic, Genetic algorithms for the Travelling Salesman Problem: A Review of Representations and Operators, Artificial Intelligence Review 13: 129 – 170, 1999
[4] Abdoun, O., Tajani, C., Abouchabaka, J., 2012. Analyzing the performance of mutation operators to solve the traveling salesman problem. Int. J. Emerg. Sci. 2 (1), 61–77, 2002