Assignment 3.1: GA on TSP problem

Aleksandra Ristic
arc19002@student.mdh.se

1. Important operations

The genetic algorithm contains selection, crossover, mutation, replacement. With the *selection*, the parents who will participate in the creation of the child are selected. In solution is implemented choosing the first parent who belongs in the half best of population and the second parent is chosen randomly. From two parents we are making one child. With *crossover* we decide how we will mix parents, in solution: from first parent we are getting middle and other free spaces we are filling with order how is in second parent. *Mutation* serves to change the child, to be different from the parent, in solution is implemented inversion mutation- choosing part of child and flipping that part. By *replication*, we determine what we will preserve and whether we will preserve some individuals from the previous generation or not it is called elitism. Is solution elitism rate is 10, that mean that we will save 10% of best individuals from previous generation.

2. Representation of the individuals

For solving this problem we are using real representation, that mean real numbers ie id number of city and all individuals has the same size. Order of number represents order how the cities will be visited. There is no recurring number, because our goal is to visit all cities once, and all individuals are starting with same id of city, mutation and crossover are implemented on the way to do not change that first number.

3. Fitness function

Fitness function represents evaluation function, it is used for ordering individuals from best to worst in that way will be shown how that solution is optimal for the problem.

$fit(x_i) = \frac{1}{p(x_i)}$ - fitness function
$p(x_i) = \sum_{j=1}^{n} d(c_{j-1}, c_j)$
$d(c_1, c_2) = \sqrt{(c_2.x - c_1.x)^2 + (c_2.y - c_1.y)^2}$ - Distance between two cities
x_i -individual of generation
$p(x_i)$ - distance between all of cities in that individual

n - number of cities
$d(c_{j-1}, c_j)$ - distance between two cities

4. Parameters used in solution

$population\_size = 40$ - in one generation we will create 40 individuals
$crossover\_rate = 90$ - so the $elitism = 10$, with this it is decided to save 10% of previous best individuals, in our case with population size= 40 mean that in new generation 4 of them will be from previous generation.
$mutation\_rate = 0.40$ - will we do mutation on some individual depends will random function will chose number between 0 and 1 which is smaller than 0.40, in this way some of individuals will not do mutation.

5. Evolving performance through the generations

In figure 1 we can see that in the first 500 generations improving the solution is going really fast, and this part is also dangerous because it can lead us to get stuck at the local maximum, because of that we have mutation and crossover which will avoid that. When we pass 500 generation it seems that more important role play mutation then crossover. But it is obvious that with every new generation we are closer to the optimal solution. Between 1000 and 2000 generation we can see that generations were stack near one point, and after 2000 they found a better solution which can be improved more and more.

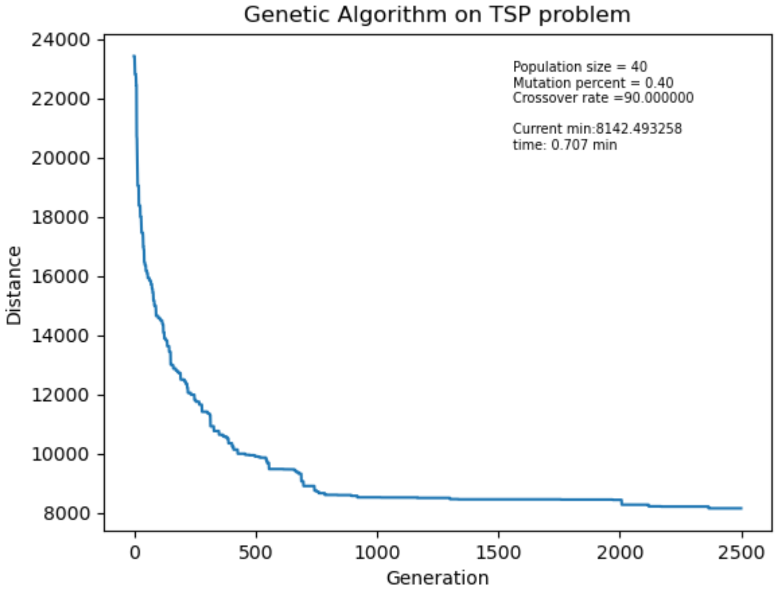

Figure 1: Performance of the population evolves with generations