# TSP - Genetic Algorithm

Mohammad Poul Doust

December 2018

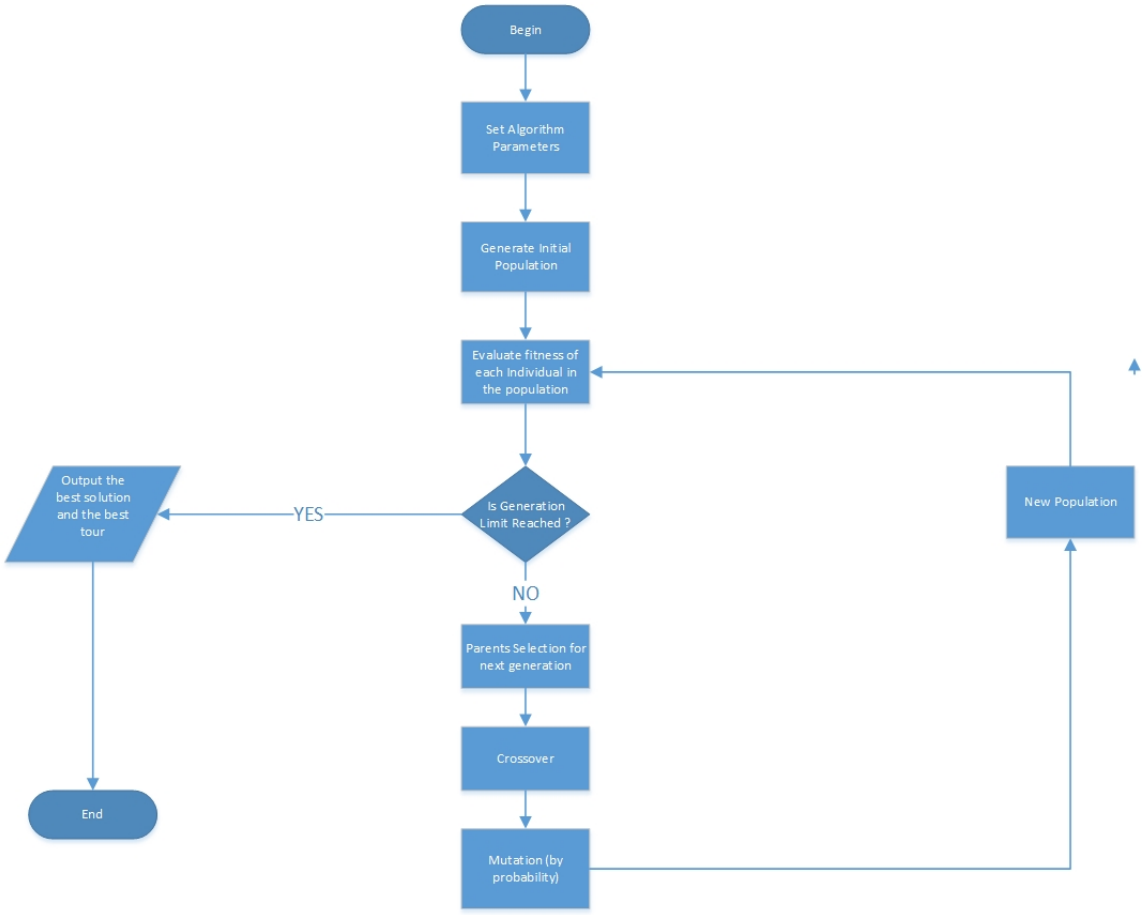
## 1 Introduction

We present in this report the main description for the attached source code to solve Travelling salesman Problem.

As illustrated in the figure below, the main algorithm pipeline contains several main components:

1. Initializing Algorithm Parameters
2. Generate Initial Population
3. Fitness Evaluation
4. Parents Selection
5. Crossover
6. Mutation
7. Generate New Population

The following sections will explain briefly each components.




## 2 Initializing Algorithm Parameters:

In this step, several parameters should be initialized by the user. Specifically,

1. Max Population Size: to specify the maximum number of individuals in any population
2. Max Generation Numbers: usually the termination condition
3. Mutation Rate: a real number in [0,1] specify the likelihood of mutation to happen

## 3 Generate Initial Population:

In this step an initial generation is created randomly by creating a first random individual (Tour), and use shuffling on this individual until we reach the max population size.

## 4 Fitness Evaluation for Elitism:

in this step, the fitness function for each individual should be calculated. For the TSP problem. the fitness function could be define as the total sum of distances for each tour. This fitness function is used to evaluate how good each individual (Solution). Consequently, the fittest individual is elited to the next generation directly.

## 5 Parents Selection:

In this step, parents are chosen to mate together (crossover) and have a new individual. the selection process could be done in several ways. we applied Roulette Selection (with flexibility in the code to add new other selection algorithms without changing the code). the basic of Roulette Selection is to pick a parent randomly according to weighted probability for each individual. the weights here represented by the fitness of the individual. In other words, individuals with high fitness function, have higher probabilities to be selected.

## 6 Crossover:

The crossover is the process of merging two individuals to have a new ones. there is several ways to implement the crossover. in our case, we implemented the Single Point Crossover. where a random point is generated bounded by the length of the solution. Afterwards, a new individuals are generated by taking the first part (before the picked point) from the first parent and the rest from the other taking into account the satisfaction of TSP constraints (no duplication)

## 7 Mutation:

Each individual might undergo a mutation with a probability (specified in the parameters). the mutation basically is done by scanning the individual genes and picking a random number (bounded by the length of tour) and swap the current gene with the randomly picked one.

## 8 Generate New Population:

We keep repeating steps 5,6,7 until we have a full new population (enhanced one). and we go back to step 4 while we have not reached the generation limit.