Laboratory of Evolutionary Computing

Genetic Algorithm for Traveling Salesman Problem

Silesian University of Technology

Author: Robert Czabanski, D.Sc.

Gliwice, 2019

1 Introduction

Genetic algorithms (GAs) formulate a research area within computer science which draws inspiration from the process of natural evolution. They are derivative-free stochastic approach based on biological evolutionary processes [5]. GAs mimic the mechanism of natural selection ("Survival of the fittest") [2] to solve optimization problems. A population of candidate solutions (individuals) characterized with a set of properties (chromosome, genotype) is evolved toward better results by using biologically inspired operators such as mutation, crossover and selection. There are a number of applications for GAs [4] which work with population of chromosomes represented by some underlying parameters set codes. This exercise shows an example of the GA approach [7] to solve the traveling salesman problem (TSP).

The TSP is one of the most intensively studied benchmarks for algorithms to deal with hard combinatorial optimization problems. The TSP considers a list of cities, usually with a full grid of connections, to be visited. Starting from a given city (home city), a so called salesman needs to visit all cities, each city exactly once, and return home in such a way that a total distance traveled is minimal.

The traveling salesman decision is not trivial NP-hard problem. When considering the symmetric TSP (where the distance d_{ij} from the i-th city to the j-th city is the same as the distance d_{ji} from the city j to the city i), the number of possible solutions is $(N-1)!/2$, which for only $N = 15$ cities equals $4.3589 \cdot 10^{+10}$.

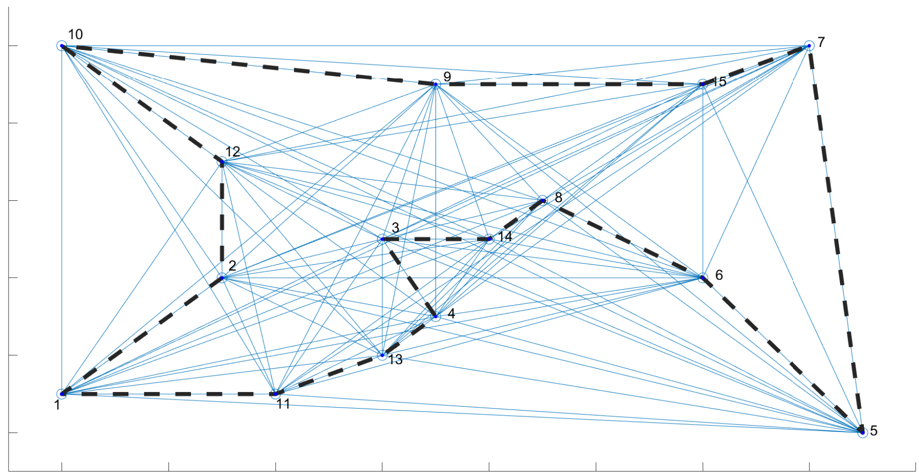

Figure 1: An example of a city network ($N = 15$) with a TSP solution

The TSP received considerable attention and various solutions were proposed [1]. Among them, the GAs based approaches can be distinguished as the most successful ones [8].

2 Algorithm

A structure of a simple genetic algorithm to solve the traveling salesman problem may be written as follows:

1. Create an initial population of P chromosomes (parents population).
2. Evaluate the cost (the total distance to be traveled) of each individual.
3. Choose $n{\cdot}P$ parents from the current population via proportional selection, where $n \in (0, 1]$.
4. Select randomly two parents to create offspring using crossover operator.
5. Repeat the Step 4 until $n \cdot P$ offspring are generated.
6. Apply mutation operators for changes in randomly selected offspring.
7. Replace old parent population with the best (of minimum cost) P individuals selected from the combined population of parents and offspring.
8. If the maximum number of generations ($T_{\max}$) were performed, then stop the algorithm; otherwise go to the Step 2.

The selection criteria, crossover, and mutation are the major operators, but crossover plays the most important role [6]. In the presented solution the Cycle Crossover Operator (CX) [7] is applied.

2.1 Chromosome

The most natural way to present a tour is by using a path representation. For example, an i-th tour π_i between $N = 5$ cities $2 \to 4 \to 5 \to 1 \to 3$ can be represented simply as a chromosome consisting of five alleles [2 4 5 1 3], where the allele position corresponds to the position in the order of visits.

2.2 Initial population

As an initial population of parents we can use a random sample of P permutations of the set $\{1, 2, \cdots, N\}$, where N is a number of cities to visit.

2.3 Cost value

The evaluation of chromosome is straightforward and defined as a the distance of travel between all cities for a given tour π_i (represented by the i-th chromosome):

$$f_i = d_{\pi_i(1),\pi_i(N)} + \sum_{k=1}^{N-1} d_{\pi_i(k),\pi_i(k+1)}, \tag{1}$$

where $d_{i,j}$ is the element of a distance matrix $\mathbf{D} = [d_{i,j}] \in \mathbb{R}^{N\times N}$.

2.4 Selection operator

In the presented method proportional selection via the roulette wheel is used i.e. the probability p_i of the i-th chromosome to be selected is inversely proportional to its cost value f_i. Consequently, the individual with low cost has a higher probability of contributing one or more offspring of the next generation.

The wheel rotation process can be simulated by an accumulating process. After obtaining a random number $r \in [0, f_s]$, where f_s is the total cost of the population $f_s = \sum_{i=1}^{P} f_i$, we select an individual k which satisfies that $\sum_{i=1}^{k} f_i < r < \sum_{i=1}^{k+1} f_i$.

2.5 Crossover operator

The strength of genetic algorithms arises from the structured information exchange by crossover combinations of low cost individuals. In the presented algorithm the CX operator is used, which works by dividing the chromosome elements into cycles. A cycle is a subset of elements which has the property that each element is always paired with another element of the same cycle of the two aligned parents. For two parents (P1 and P2), CX generates an offspring (O) by choosing a subsequence of a tour from one parent while preserving the relative order of cities from the other parent. The crossover procedure can be described as follows [3]:

1. Start with the first unused position of O and the first allele of P1.
2. Look at the allele in the same position in P2.
3. Go to the position with the same allele in P1.
4. Add this allele to the cycle.
5. Repeat steps 2 through 4 until you arrive at the first allele of P1.

The cycle crossover is best explained with an example. Let us assume that we have parents P1 = [2 4 5 1 3] and P2 = [1 5 4 2 3]. The offspring (O1) is then created as follows:

1. Select a first element of P1. As P1(1) = 2 the first allele of the offspring equals 2: O1 = [2 $-$ $-$ $-$ $-$].
2. As P2(1) = 1 and the position of city 1 in P1 is 4, the offspring O1 is augmented to become: O1 = [2 $-$ $-$ 1 $-$].
3. P2(4) = 2, however the offspring already includes city 2. Hence, we copy the remaining cities to the offspring from P2. As a result we get: O1 = [2 5 4 1 3].

In cycle crossover, we often create a second offspring (O2) by using the CX process with the roles of P1 and P2 reversed:

1. Select a first element of P2. As P2(1) = 1 the first allele of the offspring is equal to 1: O2 = [1 $-$ $-$ $-$ $-$].

2. As P1(1) = 2 and the position of city 2 in P2 is 4, the offspring O2 is augmented to become: O2 = [1 − − 2 −].

3. P1(4) = 1, however the offspring already includes city 1. Hence, we copy the remaining cities to the offspring from P1. Finally we get: O2 = [1 4 5 2 3].

The drawback of CX is that sometimes it produces offspring which are the same as parents. For example, for the following two parents P1 = [3 4 1 2 5] and P2 = [4 2 5 1 3] the CX generates offspring O1 = [3 4 1 2 5] and O2 = [4 2 5 1 3], being identical as their parents.

2.6 Mutation operator

After the generation of offspring has been determined, the chromosomes are subjected to a low rate mutation process. In genetic algorithms, mutation is realized as a random deformation of alleles with a certain probability p_m. In our algorithm we apply a simple two point mutation, which randomly selects two elements of the chromosome and swaps them. For example, by selecting the mutation points as 1 and 3 offspring O = [4 2 5 1 3] becomes O = [5 2 4 1 3].

3 Tasks

1. Implement the described GA algorithm to solve the traveling salesman problem. Use a map of cities provided by a tutor ($N = 10$) and the GA parameters: $P = 250, n = 0.8, p_m = 0.2, T_{\max} = 1000$. What was the minimal total distance traveled? What is the sequence of cities to be visited ensuring the minimal total distance traveled? Show the results in a graphic form (as a network of connections).

2. Investigate the influence of parameters P, n and p_m on the mean minimal total distance traveled calculated for 10 trials. Change the values of P within a set $\{100, 300, 500\}$, $n \in \{0.5, 0.7, 0.9\}$ and $p_m \in \{0.1, 0.3, 0.5\}$.

References

[1] D.L. Applegate, R.E. Bixby, V. Chvatal, W.J. Cook, *The traveling salesman problem a computational study.* Princeton University Press, 2007.

[2] C. Darwin, *On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life.* London: John Murray, 1869.

[3] A.E. Eiben, J.E. Smith, *Introduction to evolutionary computing.* Springer-Verlag Berlin Heidelberg, 2015.

[4] D.E. Golberg, *Genetic algorithms in search, optimization and machine learning.* Addison-Wesley Publishing Company, 1989.

[5] J.H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.* University of Michigan Press, Oxford, UK, 1975.

[6] A. Hussain, Y.S. Muhammad, M.N. Sajid, I. Hussain, A.M. Shoukry, S. Gani, *Genetic algorithm for traveling salesman problem with modified cycle crossover operator.* Computational Intelligence and Neuroscience, vol. 2017, 2017.

[7] I.M. Oliver, D.J.D. Smith, R.C.J. Holland, *Study of permutation crossover operators on the traveling salesman problem.* Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms: July 28-31, 1987 at the Massachusetts Institute of Technology, Cambridge, MA, USA, 1987.

[8] J.-Y. Potvin, *Genetic algorithms for the traveling salesman problem.* Annals of Operations Research, vol. 63, pp. 339370, 1996.