Using new variation Crossover operator of Genetic Algorithm for Solving the Traveling Salesmen Problem

Namit Gupta[#] and Rajeev Kumar[*]

[#]Assistant professor, Teerthanker Mahaveer University, Moradabad
[*]Assistant Professor Department of Computer Science
Email ID: namit.k.gupta.coe@tmu.ac.in

Abstract – Genetic algorithm (GAs) has been used as a search technique of many NP problems. Genetic algorithms have been successfully applied to many different types of problems, though several factors limit the success of a GA on a specific function. Problem required are good, but optimal solutions are not ideal for GAs. It is depended on the selection operator, crossover and mutation rates. In this paper Roulette Wheel Selection (RWS) operator with different crossover and mutation probabilities, is used to solve well known optimization problem Traveling Salesmen Problem (TSP). We have proposed a new crossover operator which is variation of Order Crossover (OX) and found results are better than existing crossover operator.

[Namit Gupta and Rajeev Kumar. **Using new variation Crossover operator of Genetic Algorithm for Solving the Traveling Salesmen Problem.** *J Am Sci* 2018;14(8):94-97]. ISSN 1545-1003 (print); ISSN 2375-7264 (online). http://www.jofamericanscience.org. 13. doi:10.7537/marsjas140818.13.

Keywords: TSP, GAs, SUS, RWS.

1. Introduction

Genetic algorithm is search algorithms based on the mechanics of natural selection and natural genetics [1], introduced by *Professor John Holland* (1960's) and *Dr. David E. Goldberg* (1989's) [2]. It is a promising heuristic approach to locate near optimal solution in large search space. It uses selection, crossover and mutation operators to solve optimization problems using a survival of the fittest idea.

The idea of the traveling salesman problem (TSP) is to find a tour of a given number of cities, visiting each city exactly once and returning to the starting city where the length of this tour is minimized.

The first instance of the traveling salesman problem was from Euler in 1759 whose problem was to move a knight to every position on a chess board exactly once [10].

The traveling salesman first gained fame in a book written by German salesman BF Voigt in 1832 on how to be a successful traveling salesman [10]. He mentions the TSP, although not by that name, by suggesting that to cover as many locations as possible without visiting any location twice is the most important aspect of the scheduling of a tour. The origins of the TSP in mathematics are not really known - all we know for certain is that it happened around 1931. The standard or symmetric traveling salesman problem can be stated mathematically as follows:

Given a weighted graph G = (V, E) where the weight cij on the edge between nodes i and j is a non-negative value, find the tour of all nodes that has the minimum total cost. Currently the only known method guaranteed to optimally solve the traveling salesman problem of any size, is by enumerating each possible tour and searching for the tour with smallest cost. Each possible tour is a permutation of 123... n, where n is the number of cities, so therefore the number of tours is n! When n gets large, it becomes impossible to find the cost of every tour in polynomial time.

TSP is one of the well known combinatorial optimization problem in which we have to find the tour of all nodes that has the minimum total cost [3, 4]. When no. of cities gets large, it becomes exhaustive search and it is impracticable to find the cost of every tour in polynomial time. Many different methods of optimization have been used to solve the TSP such as Hill Climbing [4], Tabu Search [5], Simulated Annealing [6], Particle Swarm [7], Ant Colony [8] and Genetic Algorithm [9, 4] etc. Here we have used GA to solve TSP, which is heuristically, good solution in reasonable time & establishing the degree of goodness. In this paper, we have considered a symmetric weighted graph and for a cost matrix Euclidean method is used.

In this paper, RWS is used, which is most democratic selection method, in which highest fitted individual is selected for the crossover and mutation.

The crossover recombines two individuals to generate new ones which might have a better performance. Most commonly used crossover methods to solve TSP problems are Partially Matched Crossover (PMX), Order Crossover (OX) & Cycle Crossover (CX) [1]. In this paper we have proposed variation of order crossover and compared its results with existing ones.

2. Main Process Of GA For TSP

Pure Genetic Algorithm is started with a set of solutions (chromosomes/individuals) called populations. It is chosen from collection of candidate solutions to a problem (search space) [2]. Solution for one population are taken and used to form a new population. This is motivated by a hope that new population will be better than the old one. Solutions, which are selected to form new population (offspring), are selected according to their fitness.

2.1. *Encoding:*

As Travelling Salesman Problem is ordering problem so we have used permutation encoding. In permutation encoding, every chromosome is a string of numbers, which represents number in a sequence, number represents each city. The idea of TSP is to find cheapest round-trip tour a salesman has to take by starting from any city and visiting all the cities once and ending at the starting city. Let us consider five cities. A Chromosomes (possible solution) which is represented by (1, 5, 3, 4, 2) e.g. (1, 5, 3, 4, 2) says order of cities, in which salesman will visit 1→5→3→4→2→1 them.

2.2. *Fitness function:* 2 2

As TSP is a minimization problem so to convert it into maximization problem we have considered fitness function f (x) =1/d, where d calculates cost (or distance/length) of the tour represented by a chromosome. The fitness function that characterizes each chromosome represents the total length of the route from the first to the last gene (city) moving according to the order of the genes in the chromosome.

If the cities are represented with x and y coordinates in 2D coordinate systems, then we calculate the distance between them according the equation [3]:

2.3. *Elitism (survivor selection):*

When creating new population by crossover and mutation, there is a big chance, that we will lose the best chromosome. Elitism is name of the method, which first copies the best chromosome (or a few best chromosomes) to new population. The rest is performed in classical way. Elitism can very rapidly increase performance of GA, because it prevents from loss of the best found solution. The elitism rate directly depends on the size of the population and the rate of elitism should be decreased when the population size increases.

. In this paper, we have chosen two best fitted chromosomes according to their fitness and all the chromosomes were used in subsequent procedure. In the resultant population to worst chromosomes were replaced by to best selected chromosomes and repeat this for all generation.

2.4. Selection*:*

After deciding the methods of encoding and elitism, the decision for selection technique is to be made. Various selection methods are available such as Roulette-Wheel Selection [1], Tournament selection, Rank Selection, Steady-State Selection, Boltzmann Selection etc. In this paper we have considered Roulette-Wheel & Stochastic Universal Selection technique for selection. Selection is to evaluate each individual and keeps only the fittest ones among them. In addition to those fittest individuals, some less fit ones could be selected according to a small probability. The others are removed from the current population [2].

Roulette-wheel (Goldberg 1989a)

In this method each individual is assigned a slice of a circular "roulette wheel", the size of the slice being proportional to the individual's fitness. The wheel is spun *N* times, where N is number of individuals in the population. On each spin, the individual under the wheel's marker is selected to be in the pool of parents for the next generation.

$$\sqrt{d \quad (x_1 \quad x_2)^2}$$

$$(y_1 \quad y_2)$$

Stochastic Universal selection (James Baker 1987)

To minimize the "spread" (the range of possible actual values, given an expected value) in RWS often SUS technique is used. Instead of spinning the roulette.

Wheel *N* times to select *N* parents; SUS spins the wheel once, but with *N* equally spaced pointers, which are used to select the *N* parents. Under this method, each individual *i* is guaranteed to reproduce at least *ExpVal* (*i*, *t*) times but not more than *ExpVal* (*i*, *t*).

If fitness variance in the population is high and a small numbers of individuals are much fitter than the other.

2.5. Crossover (Rearranging operator):

The crossover operation continues until the specified crossover rate is met. The crossover rate for binary chromosome is as high as 80-90%, whereas the crossover rate used here is 50% due to the decimal chromosome.

New variation in Order Crossover:

We have proposed variant of order crossover (Ox6), which is similar as to cut point selection of the first variation (Ox2), except the repairing procedure. Two cut points are selected and the elements between them are copied. The rest elements are copied from the beginning of the second parent respecting their relative order omitting those which already exist in the first parent.

Consider the above example.

O1 = (- - _ 3 4 5 _ - - - -) and

O2 = (- - _ 7 1 2 _ - - - -)

The sequence of cities in the second parent is 7 1 2 4 9 3 6 8 5 Removing 3, 4 and 5 we get 7 1 2 4 9 6 8 Placing this sequence in the first offspring from left to right according to the order we have

O1 = (7 1 3 4 5 2 4 9 8) Similarly

O2 = (3 4 7 1 2 5 6 8 9).

2.6. ***Mutation:***

The mutation in GAs works on a single chromosome at a time and alters the genes of the chromosomes randomly. To solve TSP using GA, the mutation plays the primary role in arriving at the solution. Its purpose is to maintain the population diversified enough during the optimization process. In this paper we have used Inversion Mutation (IVM). In which GAs chooses a particular chromosome is chosen at random for mutation and two indices within the chromosome are randomly selected. In Table.1 randomly select 2 and 5 indices and the order of genes within the indices is reversed.

Table 1

A	1 2 **3** 4 5 **6** 7
A'	1 2 **6 5 4 3** 7

The probability of mutation generally is very low and it is of the order of one tenth of a percent for binary chromosome. But in the case of decimal chromosomes, the mutation rate goes up to of the order of 85%.

3. Discussion

3.1. Control Parameter

GA performs well in cases where it is more important to find a good solution rather than the absolutely best solution [5]. Population Size is determines how many chromosomes and thereafter, how much genetic material is available for use during the search. If there is too little, the search has no chance.

3.2. Termination criteria:

It is Important to remember that GA in most cases provide only approximately correct solution, not optimal **begin** GA-TSP

Create initial population (c library Random Function)

while (generation count < *gen*) and

(*All individual cost are not equal*)

do*/*gen = max. Number of generations. */*

begin

Elitism (2 best tours) Selection

Crossover

Mutation

Replace worst with best tour Increment generation count

end

Output the best and worst individuals found

end GA-TSP

4. Conclusion

Genetic algorithms appear to find good solutions for the traveling salesman problem, however it depends very much on the way the problem is encoded and which crossover and mutation methods are used. It seems that the methods that use heuristic information or encode the edges of the tour (such as the matrix representation and crossover) perform the best and give good indications or future work in this area.

Overall, it seems that genetic algorithms have proved suitable for solving the traveling salesman problem. As yet, genetic algorithms have not found a better solution to the traveling salesman problem than is already known, but many of the already known best solutions have been found by some genetic algorithm method also.

It seems that the biggest problem with the genetic algorithms devised for the traveling salesman problem is that it is difficult to maintain structure from the parent chromosomes and still end up with a legal tour in the child chromosomes. Perhaps a better crossover or mutation routine that retains structure from the parent.

Chromosomes would give a better solution than we have already found for some traveling salesman problems.

References

1. Goldberg, D.E. (1989) "Genetic Algorithms in search, optimization machine learning", Pearson Education.
2. Zbigniew Michalewicz. Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag, 2nd edition, 1994.
3. Mitchell M. "An Introduction to Genetic Algorithms", Prentice-Hall of India.
4. Goldberg, D.E. (1989) "Genetic Algorithms in search, optimization machine learning", Pearson Education.
5. Ellis Horowitz, Sartaj Sahni, Rajasekarn S."Fundamentals of Computer Algorithms", Galgotia Publication pvt. ltd.
6. Stojanovska, I., Dika, A. "Impact of the Number of Generations on the Fitness value and the Time Required to Find the Optimal Solution in Standard GA Applications", Proc of the 2010

IEEE, pp. 978-1-4244-5540-9(2010).
7. Thamilselvan, R., Balasubramanie, P., "A Genetic Algorithm with a Tabu Search (GTA) for Traveling Salesman Problem", International Journal of Recent Trends in Engineering, Vol. 1, No. 1, May 2009 ACEEE.
8. Laarhoven, P.V. and Aarts, E.H.L.: Simulated Annealing: Theory and Applications, Kluwer Academic (1987).
9. Kennedy, J and Eberhart, R.C.: Swar Intelligence.
10. Networks, Fuzzy Logic, and Genetic Algorithms.

8/25/2018