

# Innovative PMX - CX- Operators for GA TO TSP

Dr.Mrs.G.Nirmala<sup>1</sup>, Mr.D.Ramprasad<sup>2</sup>

<sup>1</sup> Associate professor in Mathematics, PG and Research Department of Mathematics, K.N.G.Arts College for women (Autonomous)

<sup>2</sup> Assistant professor in Mathematics, PG and Research Department of Mathematics, A.V.V.M Sri Pushpam college (Autonomous)

**Abstract-** This paper analyses process of Genetic algorithm and its operators. Several cross over operators like partially matched cross over, circular motion of cross over, order cross over are discussed. GA's are powerful methods of optimization used successfully in travelling salesman problem. Various cross over operators in the context of the travelling salesman problem are discussed. The results of different GA cross over operators for the TSP are compared and presented.

**Index Terms-** Genetic algorithms, travelling salesman problem, order cross over, mutation

## I. INTRODUCTION

Genetic algorithm first proposed by John Holland is a derivative free stochastic optimization approach based on the concept of biological evolutionary process. Genetic algorithms are an optimization technique based on natural evolution. They include the survival of the fittest idea into a search algorithm which provides a method of searching which does not need to explore every possible solution in the feasible region to obtain a good result. Genetic algorithms are based on the natural process of evolution. In nature, the fittest individuals are most likely to survive and mate; therefore the next generation should be fitter and healthier because they were bred from healthy parents. This same idea is applied to a problem by first 'guessing' solutions and then combining the fittest solutions to create a new generation of solutions which should be better than the previous generation. We also include a random mutation element to account for the occasional 'mishap' in nature.

### Travelling salesman problem

The travelling salesman problem is one of the most famous combinatorial optimization problems. The objective of the travelling salesman problem is to minimize the total distance travelled by visiting all the cities once and only once and then returning to the depot city.

A common application of the travelling salesman problem is the movement of people, equipment and vehicles around tours of duty to minimize the total travelling cost.

The travelling salesman problem is used in various fields such as operations research, computer science, discrete mathematics and graph theory and so on.

## II. METHODS OF SOLUTION

### Definition:

A set of cities, and known distances between each pair of cities, the travelling salesman problem is the problem of finding

a tour that visits each city exactly once and that minimizes the total distance travelled.

### Definition:

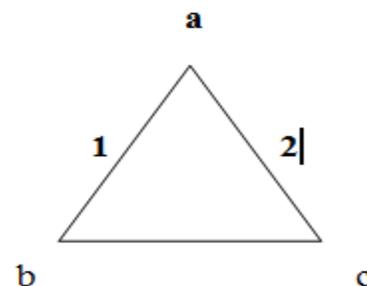
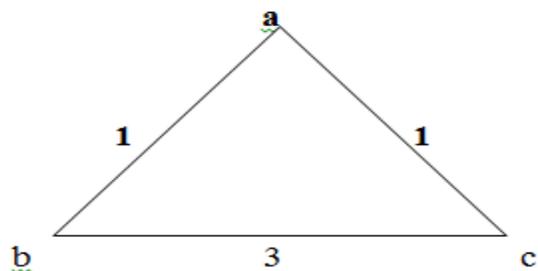
The methods for solving the travelling salesman problem usually can be divided into three basic parts:

- i. a starting point
- ii. a solution generation scheme and
- iii. a termination rule

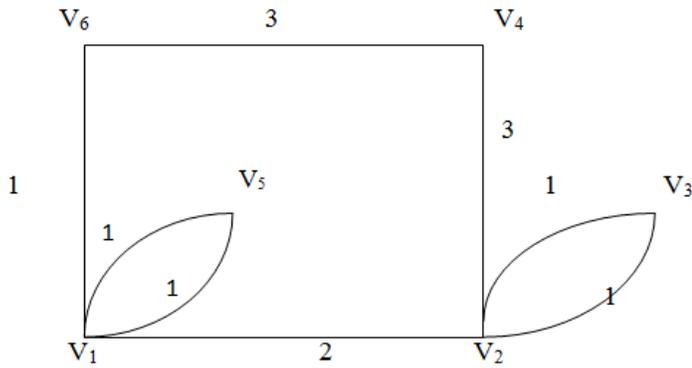
### An improved approximation algorithm for the travelling salesman problem

- i. Find a minimum-weight spanning-tree  $T$  of fig (a).
- ii. Construct the set  $V'$  of vertices of odd-degree in  $T$  and find a minimum-weight perfect matching  $M$  for  $V'$ .
- iii. Construct the Eulerian graph fig (b) obtained by adding the edges of  $M$  to  $T$ .
- iv. Find an Eulerian circuit  $C_0$  of fig (b) and index each vertex according to the order,  $L(v)$ , in which  $v$  is first visited in a trace of  $C_0$ .
- v. Output the following approximate minimum-weight Hamiltonian circuit:

$$C = (v_{i1}, v_{i2}, \dots, v_{in}, v_{i1}) \text{ where } L(v_{ij}) = j.$$



**An application of the minimum-weight matching algorithm for the travelling salesman problem**



V <sub>4</sub>	5
V <sub>5</sub>	2
V <sub>6</sub>	6

$C_0 = (V_1, V_5, V_1, V_2, V_3, V_2, V_4, V_6, V_1)$        $C = (V_1, V_5, V_2, V_3, V_4, V_6, V_1)$

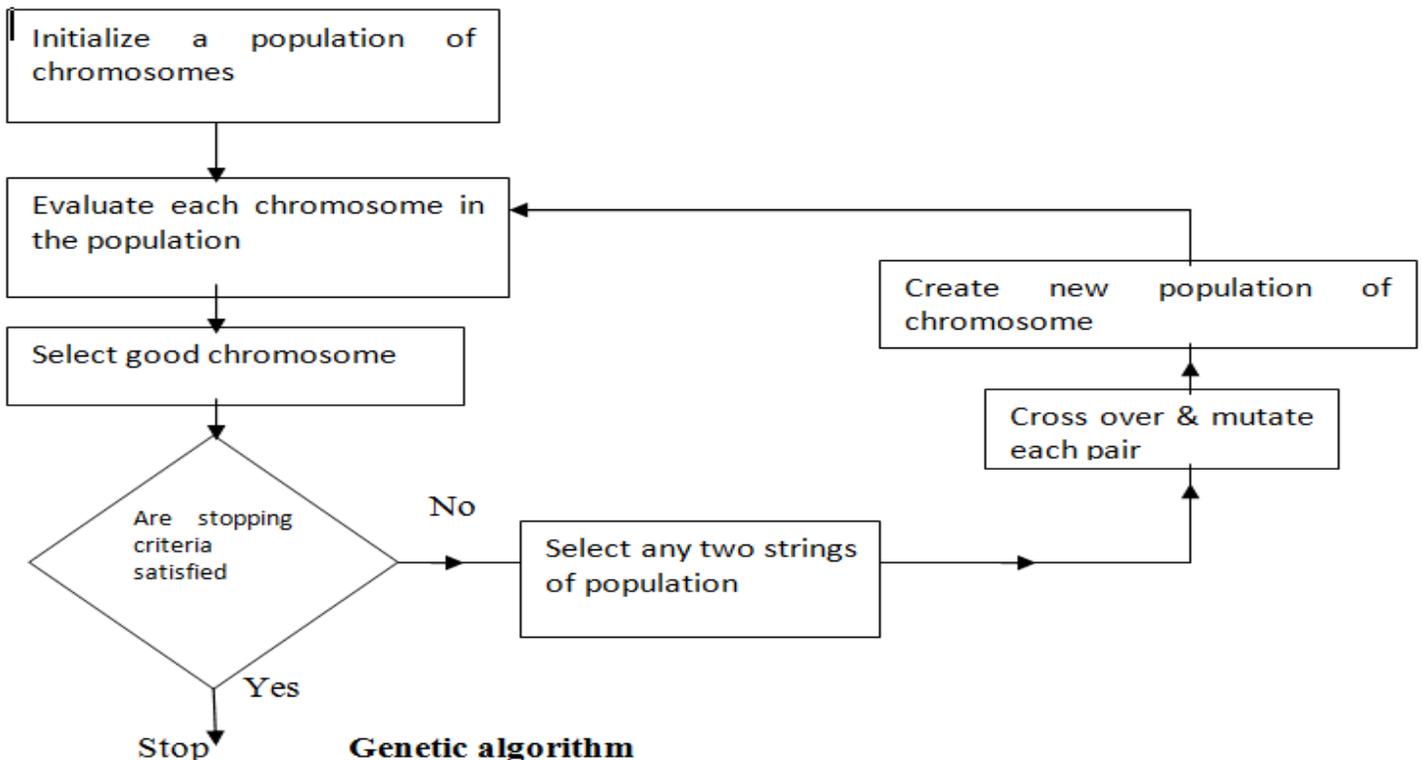
V	L(V)
V <sub>1</sub>	1
V <sub>2</sub>	3
V <sub>3</sub>	4

**Working of Genetic algorithm**

**Genetic algorithm works with a population of strings called chromosomes.**

Genetic algorithms range from being very straightforward to being quite difficult to understand. Before proceeding, a basic explanation is required to understand how genetic algorithms work. We will use the following problem. We want to maximize the function  $f = -2x^2 + 4x - 5$  over the integers in the set  $\{0, 1, \dots, 15\}$ . By calculus or brute force we see that  $f$  is maximized when  $x=1$ .

The basic way of encoding a problem using a string of zeros and ones, which represent a number in its binary form. We can also use a string of letters, for example  $C_1C_2C_3C_4C_5$ , or a string of integers, 12345, or just about any string of symbols as long as they can be decoded into something more meaningful.



Imagine we had a problem involving a graph and we needed to encode the adjacency list of the graph. We could create the adjacency matrix, which consists of a one in the  $i, j^{\text{th}}$  position if there is an arc from node  $i$  to node  $j$  and a zero otherwise. We could then use the matrix as is or we else could concentrate the rows of the matrix to create one long string of zeros and ones. Notice this time, however, the string is not a binary representation of a number.

This leads us to the first method of encoding a tour of the travelling salesman problem. We do have a graph such as the one described above and we can encode it in the same way, only our matrix will have a one in the  $i, j^{\text{th}}$  position if there is an arc from node  $i$  to node  $j$  in the tour and a zero otherwise. For example, the matrix

$$\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

represents the tour that goes from city 1 to city 3, city 3 to city 2 and city 2 to city 1. This encoding is known as matrix representation. The travelling salesman problem can also be represented by a string of integers in two different ways. The first is by the string

$$v = a_1 a_2 \dots a_n$$

which implies that the tour goes from  $a_1$  to  $a_2$  to  $a_3$ , etc and from  $a_n$  back to  $a_1$ . Notice that the strings  $v_1 = 1234$  and  $v_2 = 2341$  are equivalent in this representation.

The second way to represent the travelling salesman problem is with cycle notation with an integer string

$$v = b_1 b_2 \dots b_n$$

where the tour goes from city  $i$  to city  $b_i$ . That is, the string  $v_1 = 3421$  means that the tour goes from city 1 to city 3, city 3 to city 2, city 2 to city 4 and city 4 to city 1. Note that not every possible string here represents a legal tour, where a legal tour is a tour that goes to every city exactly once and returns to the first city. It is possible for us to have a string that represents disjoint cycles, for example,  $v_2 = 3412$  implies that we go from city 1 to city 3 and back to city 1 and from city 2 to city 4 and back to city 2.

**Crossover**

Several crossover methods have been developed for the travelling salesman problem. We describe several of them. We start by looking at partially matched crossover (PMX). Recall the two-point crossover and assume we were to use this with the integer representation defined for the travelling salesman problem. If we performed a two-point crossover on the chromosomes of the TSP.

Parent  $v_1 = 1234 | 567 | 8$   
 Parent  $v_2 = 8521 | 364 | 7$

we would get

Child  $v_1' = 1234 | 364 | 8$   
 Child  $v_2' = 8521 | 567 | 7$

which are obviously illegal because  $v_1'$  does not visit cities 5 or 7 and visits cities 4 and 3 twice. Similarly  $v_2'$  does not visit cities 4 or 3 and visit cities 5 or 7 twice. PMX fixes this problem by noting that we made the swaps  $3 \leftrightarrow 5, 6 \leftrightarrow 6$  and  $4 \leftrightarrow 7$  and then

repeating these swaps on the genes outside the crossover points, giving us

$$v_1'' = 12573648$$

$$v_2'' = 83215674$$

In other words, we made the swaps,  $3 \leftrightarrow 5, 6 \leftrightarrow 6, 4 \leftrightarrow 7$  and the other elements stayed the same.  $v_1''$  and  $v_2''$  still consist of parts from both the parents  $v_1$  and  $v_2$  and are now both legal.

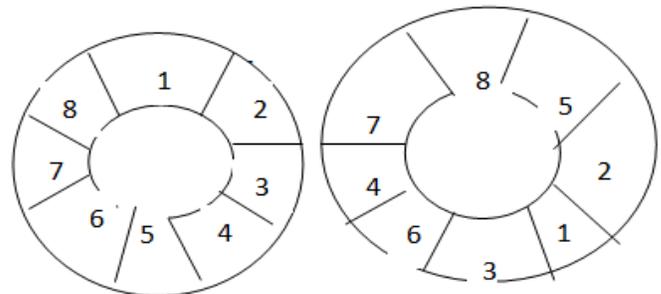
This crossover would make more sense when used with the cycle representation, since in this case it would preserve more of the structure from the parents. If, as in our example, we used the first integer representation, the order that the cities were visited would have changed greatly from the parents to the children - only a few of the same edges would have been kept. With cycle notation a lot more of the edges would have been transferred. However, if we use this crossover routine with cycle representation we do not necessarily get a legal tour as a result. We would need to devise a repair routine to create a legal tour from the solution that the crossover gives us, by changing as little as possible in order to keep a similar structure.

Cycle crossover (CX) works in a very different way. First of all, this crossover can only be used with the first representation we defined, that is, the chromosome  $v = 1234$  implies that we go from city 1 to city 2 to city 3 to city 4. This time we do not pick a crossover point at all. We choose the first gene from one of the parents

Parent  $v_1 = 123456788$   
 Parent  $v_2 = 85213647$

say we pick 1 from  $v_1$

$$v_1' = 1 \text{ --- } 5$$



we must pick every element from one of the parents and place it in the position it was previously in. Since the first position is occupied by 1, the number 8 from  $v_2$  cannot go there. So we must now pick the 8 from  $v_1$ .

$$v_1' = 1 \text{ --- } 8$$

This forces us to put the 7 in position 7 and the 4 in position 4, as in  $v_1$ .

$$v_1' = 1 \text{ -- } 4 \text{ -- } 78$$

Since the same set of positions is occupied by 1, 4, 7, 8 in  $v_1$  and  $v_2$ , we finish by filling in the blank positions with the elements of those positions in  $v_2$ . Thus



2. However, if the number of elements in the resultant tour is less than 2 then add the missing edges in the tour by greedy algorithm. Considering two tours

$$T1: C_1 C_5 C_3 C_4 C_2 C_1 = 17$$

$$T2: C_1 C_2 C_5 C_3 C_4 C_1 = 22$$

**Value of the assignment:**

Consider the weighted matrix of the given problem and solve it by using assignment algorithm and called the optimal total cost as a value of the assignment problem.

The value of the assignment of the above problem is 21.

Initial population:

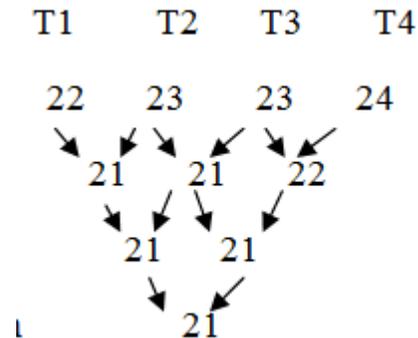
$$T1 : C_1 C_4 C_2 C_5 C_3 C_1 = 22$$

$$T2 : C_1 C_2 C_5 C_3 C_4 C_1 = 23$$

$$T3 : C_1 C_2 C_4 C_3 C_5 C_1 = 23$$

$$T4 : C_1 C_4 C_3 C_2 C_5 C_1 = 24$$

According to the fitness we are selecting these four tours.



III. STEPS OF ALGORITHMS

- i. Randomly create the initial population of individual strings of the given TSP problem and create a matrix representation of each, must satisfy the two basic conditions as mentioned earlier.
- ii. Assign a fitness to each individual in the population using fitness criteria measure,

$$F(t) = \frac{\text{value of the assignment of the given problem}}{\text{value of the string}}$$

The selection criterion depends upon the value of the strings if it is close to 1.

- iii. Create new off-spring population of strings from the two existing strings in the parent population by applying cross over operation.
- iv. Mutate the resultant off-springs if required.

After the cross over and mutation off-spring population has the fitness higher than the parents.

- v. Call the new off-springs as parent population and continue the steps (iii) and (iv) until we get a single off-spring that will be an optimal or near optimal solution to the problem.

**Example:**

Consider the weighted matrix,

	C1	C2	C3	C4	C5
C1	∞	4	7	3	4
C2	4	∞	6	3	4
C3	7	6	∞	7	5
C4	3	3	7	∞	7
C5	4	4	5	7	∞

Initial population:

After 1<sup>st</sup> cross over and mutation

After 2<sup>nd</sup> cross over and mutation

After 3<sup>rd</sup> cross over and mutation

The resultant tour will be

$$C_1 C_5 C_3 C_2 C_4 C_1.$$

IV. CONCLUSION

In this paper Genetic algorithms have proved that they are suitable for solving TSP. The operators PMX, CX, OX etc., playing an important role by developing Robust Genetic algorithms. The TSP problem with the GA devised is that it is difficult to maintain structure from the parent chromosomes and still end up with a legal tour in the child chromosomes.

REFERENCES

- [1] Goldberg D., Computer-Aided Pipeline Operations Using Genetic Algorithms and Rule Learning. Part I; Genetic Algorithms in pipeline Optimization, Engineering with Computer 3, 1987.
- [2] D.E. Goldberg, Genetic Algorithm in Search, Optimization and Machine Learning. Addison- Wesley, 1989.
- [3] Gerard Reinelt. The Traveling Salesman: Computational Solutions for TSP Applications. Springer-Verlag, 1994.
- [4] Dr.G.Nirmala and Mr.D.Ramprasad “ Genetic Algorithm Approach for minimizing waiting times in Railway Networks” Narosa Publication (2010), ISBN 978-81-8487-074-9.
- [5] P. Miliotis, Using cutting planes to solve the symmetric travelling salesman problem, Mathematical programming 15 (1978) 177–188.
- [6] Dr.G.Nirmala and Mr.D.Ramprasad “Application of Genetic Algorithm in Graph theory “ International Journal of Scientific and Research Publications, Volume 2, Issue 7, (July 2012), ISSN 2250-3153.

- [7] J.Y. Potvin, Genetic algorithms for the travelling salesman problem, *Annals of Operations Research* 63 (1996) 339–370.
- [8] Dr.G.Nirmala and Mr.D.Ramprasad “ Genetic operators with Algebra of Equivalence relations “ International conference on Algebra and its applications,PMU journal (ISSN).
- [9] T.Ramaraj and G.Nirmala “ Genetic Algorithm to formulate Quadratic steiner tree problem of optimal Neuro fuzzy Network “ ACTA ciencia Indica 31(3) 897 – 902 (ISSN).
- [10] Holland, J.H., *Adaptation in Natural and Artificial Systems*, 2nd Edition, MIT Press, 1992.
- [11] T.Ramaraj and G.Nirmala “ Genetic Algorithm matrix Neural Network stochastic relaxation methods “ *Applied science periodicals* (2010), ISBN 978-81-8487-074-9.
- [12] M. Gen, R. Cheng, “*Genetic Algorithms and Engineering, Optimization,*” John Wiley & Sons, Inc., New York, 2000.
- [13] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys. *The Traveling Salesman*. JohnWiley and Sons, 1986.
- [14] D. E. Goldberg and R. Lingle. Alleles, loci, and the traveling salesman problem. In *Proceedings of the International Conference on Genetic Algorithms and Their Applications*, pages 154–159, 1985.
- [15] T.Ramaraj and G.Nirmala “ Genetic Algorithm for Maximum – model discrimination design – Matrix – Represented solutions “ *Science and Technology* – 2004.
- [16] S. Lin, B.W. Kernighan, An effective heuristic algorithm for travelling salesman problem, *Operations Research* (1973) 498–516.

#### AUTHORS

**First Author** – Dr.Mrs.G.Nirmala, Associate professor in Mathematics, PG and Research Department of Mathematics,K.N.G.Arts college for women (Autonomous),Thanjavur.nirmalamanokar11@yahoo.com  
**Second Author** – Mr.D.Ramprasad Assistant professor in Mathematics, PG and Research Department of Mathematics,A.V.V.M Sri Pushpam college (Autonomous),Poondi.drprasad1981@gmail.com