A COMPARATIVE ANALYSIS OF PMX, POS AND OX CROSSOVER OPERATORS FOR SOLVING TRAVELLING SALESMAN PROBLEM

1Sarika Goel, 2Vaishali Wadhwa
1,2N.C.College of Engineering, Israna, Panipat, Haryana, India

Abstract - A very famous known NP-Complete problem is Travelling Salesman problem i.e. current algorithms require exponential time to reach a solution so, it require innovative solutions. A number of new solutions are demanded by this problem so that it can be solved in small amount of time. Genetic Algorithm is one of the best methods which are used to solve various NP-Complete problems such as TSP. A number of crossover operators have been proposed for TSP problem. This paper represents the comparative analysis of the PMX, OX and POS operators. Experimental results shows that the PMX crossover operator is best among the above three mentioned crossover operators as it find the shortest path, which is best among the three.

Keyword - TSP, NP-hard, Genetic Algorithm, Mutation, Selection, Crossover, PMX, OX, POS.

I. INTRODUCTION

Genetic Algorithm (GA) is an approximate and optimizing algorithm which is based on the biological evolution process to find the shortest tour in short instant of time. Crossover operators are the backbone of a genetic algorithm. While using the Genetic Algorithms the problem of trapping into local optima is resolved. Crossover operators are used to produce the offspring from the existing parents by maintaining the partial tours.

The partially matched crossover (PMX) [1] was proposed by Goldberg and Lingle which produces the offspring by selecting a sub sequence of a tour from one parent and maintains the order and position of cities in other parent. The Position based crossover operator (POS) [2] was proposed by Syswerda which produces an offspring by selecting a random set of positions in the parent tours. However, this operator imposes the position of the selected cities on the corresponding cities of the other parent. The ordered crossover (OX)[3] was proposed by Davis which produces an offspring by selecting a sequence of parent and preserve the relative order of cities in other parent.

II. GENETIC ALGORITHM WITH PMX, POS, OX OPERATORS

John Holland proposed Genetic Algorithm in 1975 [4]. In the field of artificial intelligence a genetic algorithm is a search heuristic that mimics the process of natural evolution. Genetic Algorithm belongs to class of evolutionary algorithm.GA begin with various problem solution which are encoded into population, a fitness function is applied for evaluating the fitness of each individual, after that a new generation is created through the process of selection, crossover and mutation. After the termination of genetic algorithm, an optimal solution is obtained. If the termination condition is not satisfied then algorithm continues with new population.

The basic steps of genetic algorithm used are given below:

(A) Initialization
An initial population is generated from many individual solutions. A problem depends upon size of the population that contains several hundreds or thousands of possible solutions.

(B) Encoding Scheme
In this we represent each city with number, for example if there are 10 cities then each city is represented by permutation of integers from 1 to 10 such as 9 1 10 2 4 3 6 7 5 8, and no number will be repeated. For N number of cities, the cities are represented by permutation of integers from 1 to N.

(C) Fitness Function
In this each solution or chromosome is assigned a fitness value. For TSP problem, the fitness value is assigned to each solution by calculating the distance between the cities in each solution.

(D) Selection
Selection operation simulates the natural law of the survival of the fittest in the population evolutionary process. The chromosomes selected with largest fitness value. The solution with minimum distance is selected for crossover operator. Here we use the Roulette Wheel selection technique. The principle of Roulette wheel selection is a linear search through roulette wheel with the slots in the wheel weighted in proportion to the individual’s fitness value. This technique comprises of following steps [5]:

1. Sum the total expected value of the individuals in the population. Let it be A.
2. Repeat N times:
(i) Choose a random number \( r \) between 0 and \( A \).
(ii) Loop through the individuals in the population, summing the expected value, until the sum is greater than or equal to \( r \). The individuals whose expected value put the sum over this limit is one selected.

(E) Crossover

Crossover operators are the backbone of the genetic algorithm. Reproduction makes clones of good strings but does not create new ones. Crossover operators are applied to mating pool with hope that it creates a better offspring. Here three crossover operators PMX, POS and OX are discussed.

Partially Matched Crossover: In partially matched crossover operator two crossover points are selected randomly from the parentis chromosomes to produce the offspring. The two crossover points give a matching selection which is used to affect a cross through position by position exchange operations [5].

The Pseudo code for PMX Genetic algorithm under TSP problem

1. Start
2. Generate the random population by using randperm function.
3. \( X=1 \)
4. Repeat step i to vi while (\( X!100 \))
   (i) Evaluate the fitness of each single chromosome using fitness function in which the weight between each individual city is summed up.
   (ii) Individual with largest fitness value is selected by using the Roulette wheel selection procedure.
   (iii) Apply the PMX crossover for producing the off springs with crossover probability i.e. \( P_C =1 \).
   For example ,consider two parents
   \[
   P_1: \quad 2 \quad 1 \quad 5 \quad 4 \quad | \quad 7 \quad 8 \quad 9 \quad 3 \quad | \quad 6 \quad 10
   \]
   \[
   P_2: \quad 1 \quad 5 \quad 4 \quad 6 \quad | \quad 10 \quad 2 \quad 8 \quad 7 \quad | \quad 3 \quad 9
   \]
   Finally we have the off springs as follows:
   \[
   O_1: \quad 9 \quad 1 \quad 5 \quad 4 \quad | \quad 10 \quad 2 \quad 8 \quad 7 \quad | \quad 6 \quad 3
   \]
   \[
   O_2: \quad 1 \quad 5 \quad 4 \quad 6 \quad | \quad 7 \quad 8 \quad 9 \quad 3 \quad | \quad 10 \quad 2
   \]
   (iv) If \( X\%10 = 0 \)
       Apply the interchanging mutation to prevent the algorithm to trapped in local optima with mutation probability \( P_M = 0.1 \).
       (v) The weak chromosomes are replaced by using weak replacement function.
       (vi) \( X=X+1 \);
   5. After 1000 iterations the algorithm will terminate.
   6. End

Ordered Crossover: In ordered crossover operator two cut points are randomly selected from parentis chromosomes. Here to produce the offspring \( O_1 \) the genes between the cut points are replaced by the genes in the second parent.

The Pseudo code for Ordered Genetic algorithm under TSP problem

1. Start
2. Generate the random population by using randperm function.
3. \( X=1 \)
4. Repeat step i to vi while (\( X!100 \))
   (i) Evaluate the fitness of each single chromosome using fitness function in which the weight between each individual city is summed up.
   (ii) Individual with largest fitness value is selected by using the Roulette wheel selection procedure.
   (iii) Apply the ordered crossover for producing the off springs with crossover probability i.e. \( P_C =1 \).
   For example, consider two parents
   \[
   P_1: \quad 2 \quad 1 \quad 5 \quad 4 \quad | \quad 7 \quad 8 \quad 9 \quad 3 \quad | \quad 6 \quad 10
   \]
   \[
   P_2: \quad 1 \quad 5 \quad 4 \quad 6 \quad | \quad 10 \quad 2 \quad 8 \quad 7 \quad | \quad 3 \quad 9
   \]
   Finally we have the off springs as follows: \( O_1: \)
   \[
   5 \quad 4 \quad 9 \quad 3 \quad | \quad 10 \quad 2 \quad 8 \quad 7 \quad | \quad 6 \quad 1
   \]
   \[
   O_2: \quad 4 \quad 6 \quad 10 \quad 2 \quad | \quad 7 \quad 8 \quad 9 \quad 3 \quad | \quad 1 \quad 5
   \]
   If \( X\%10 = 0 \)
   Apply the interchanging mutation to prevent the algorithm to trapped in local optima with mutation probability \( P_M = 0.1 \).
   (v) The weak chromosomes are replaced by using weak replacement function.
   (vi) \( X=X+1 \);
   5. After 1000 iterations the algorithm will terminate.
   6. End

Position based operator: In Position based crossover operator random set of positions are selected in the parent tours. However, this operator imposes the position of the selected cities on the corresponding cities of the other parent.

The Pseudo code for Cyclic Genetic algorithm under TSP problem

1. Start
2. Generate the random population by using randperm function.
3. \( X=1 \)
4. Repeat step i to vi while (\( X!100 \))
   (i) Evaluate the fitness of each single chromosome using fitness function in which the weight between each individual city is summed up.
   (ii) Individual with largest fitness value is selected by using the Roulette wheel selection procedure.
   (iii) Apply the POS crossover for producing the off springs with crossover probability i.e. \( P_C =1 \).
Consider the following parents:

P1: 1 2 3 4 5 6 7 8
P2: 2 4 6 8 7 5 3 1

The offspring produced are:

O1: 1 4 6 2 3 5 7 8
O2: 4 2 3 8 7 6 5 1

(iv) If $X \% 10 = 0$
Apply the interchanging mutation to prevent the algorithm to trapped in local optima with mutation probability $PM = 0.1$.

(v) The weak chromosomes are replaced by using weak replacement function. (vi) $X = X + 1$.

(5) After 1000 iterations the algorithm will terminate.

(6) End

(F) Mutation

After the crossover operation, the chromosomes are subjected to mutation process. Mutation prevents the algorithm to be trapped in local optima. Here interchanging mutation is used. In this mutation process two random position of chromosomes is chosen and the bits corresponding to that position are interchanged the mutation probability $PM$ is taken as 0.1.

(G) Termination

When the algorithm has run a given number of iterations, it stops and output the best solution. This generational process is repeated until a termination condition has been reached. Here the algorithm terminates after 1000 iterations.

III. SIMULATION RESULTS:

The following graph shows the distance find out by PMX, OX and POS operator with 25 number of cities and $PC=1$ and $PM = 0.1$

The shortest path find out by PMX, OX and POS operator is shown below for 30 numbers of cities:

Table 2: Distance measured by PMX, OX and POS

<table>
<thead>
<tr>
<th>Cases</th>
<th>PMX</th>
<th>POS</th>
<th>OX</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>462</td>
<td>510</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>525</td>
<td>557</td>
<td>607</td>
</tr>
<tr>
<td>3</td>
<td>420</td>
<td>526</td>
<td>563</td>
</tr>
<tr>
<td>4</td>
<td>512</td>
<td>587</td>
<td>678</td>
</tr>
<tr>
<td>5</td>
<td>546</td>
<td>601</td>
<td>644</td>
</tr>
<tr>
<td>6</td>
<td>563</td>
<td>571</td>
<td>641</td>
</tr>
</tbody>
</table>

The following table gives the briefing of experiments: Table 3:

Result of PMX, POS and OX

<table>
<thead>
<tr>
<th>No. of Cities</th>
<th>$PC$ Crossover Probability</th>
<th>$PM$ Mutation Probability</th>
<th>Average Distance By PMX</th>
<th>Average Distance By OX</th>
<th>Average Distance By POS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
REFERENCES


