Volume 4, Issue 3, March 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

# A Novel Hybrid Crossover Operator for Genetic Algorithm

**R Lakshmi*** — *Pondicherry University, India*

**K Vivekanandan** — *Pondicherry Engineering College, India*

***Abstract— Simple Asymmetric Travelling Salesman Problem (TSP) is a combinational NP complete problem. The number (n) of cities in a tour is 5 then the possible paths are 120 (5!). When there are 25 or more cities to visit, brute force search technique fails to solve. Instead, heuristic & probabilistic search methods are more sensible in solving NP hard problems. In this paper, Genetic algorithm and crossover are analyzed and a novel crossover operator has been established by combining two crossover techniques named PMX and FOMX called as hybrid crossover. The proposed operator is tested with different crossover techniques. Inputs have taken from TSPLIB database. This paper compares the results of hybrid crossover with the results of other crossover operators and it is found that the proposed crossover outperforms the rest.***

***Keywords— Hybrid crossover, Fast Order Mapped Crossover, Genetic Algorithms, TSP, Performance Analysis***

## I. INTRODUCTION

One of the popular combinatorial problems is Travelling Salesman Problem (TSP). It belongs to NP-complete class of problems. In past many evolutionary algorithms are used in solving TSP like Genetic Algorithm. GA is a population based search optimization technique includes steps: Initialization, Selection, Crossover, Mutation and Replacement [1]. Initialization is used to generate the initial population randomly. Selection is a reproduction operator used to select the fittest individuals from the population and give it in the next generation. Crossover is used to explore the search space to get better diversity in the population. Normally Mutation is ignorable because the mutation probability is 0.001. Replacement is used to progress the generation wise population. In the past, several crossover operators are used to solve TSP problem. These are discussed in Section II with their pros and cons. A novel hybrid crossover operator is proposed and explained in Section III. Comparison of existing crossover operators with proposed crossover is carried out in Section IV. Finally section V concludes.

## II. RELATED WORK

Crossover operators are the backbone of the genetic algorithm. Reproduction makes clones of individuals with hope that it creates a better child. Partially Mapped Crossover (PMX) is the most widely used crossover operator for chromosomes having permutation encoding [3]. Crossover operator builds an offspring by swapping a subsequence of tour from one parent on another parent. The subsequence tour is picked from two random cut points namely cp1 and cp2, which serve as crossover regions for the exchanging operations, [1, 5 & 6].

Cycle crossover (CX) is used for chromosomes with permutation encoding. In Cycle crossover [9] each allele comes from one parent together with its position. It splits the elements into cycles. A cycle is a division of elements that has the property that each element always paired with another element of the same cycle when the two parents are allied. Cycle Crossover happens by selecting few cycles from one parent and the rest from an alternate parent. All the genes in the children occupy the same positions in one of the two parents. First a cycle of position values from parent 1 is obtained. Then the alleles of the cycle are put in child 1. Next cycle is taken from parent 2 and the process is continued [1 & 3].

Order Crossover (OX) builds an offspring by choosing a sub tour from one parent and preserving the order of cities on other parent. It copies the part of permutation elements within the crossover points from the cut string directly to the offspring, placing them in the same fixed position [5].

Position Based Crossover (PBX) selects random positions in the parent. The allele of those positions is copied into their offspring. Before copying the allele, the occurrence of that allele should be checked in the offspring. If it is already copied then take the next random position and copy that in the offspring. This process is continued up to the length of the chromosome [8].

Sorted Match Crossover [11] searches for sub tours in both the parents who have the same length and same set of cities; it starts and copies cities in the same order. If such sub tours are there the costs of these are determined. The offspring is constructed from the parent who contains the sub tour with the highest cost by substituting the sub tour for the sub tour with the lowest cost.

Maximal Preservative Crossover (MPX) is similar to PMX Crossover. It first selects a random sub string of the first parent whose length is greater than or equal to 10 (except for small problem instances) and smaller than or equal to the problem size divided by 2. These restrictions assure that enough information is there to exchange between the parent strings without losing too much information from any of the both parents. The selected cities (sub tour) are removed from

© 2014, IJARCSSE All Rights Reserved 

the second parent. The sub tour chosen from parent1 is copied into the first part of the offspring. Finally the end of the offspring is filled up with cities in the same order as they appear in the second parent [11].

The partially mapped crossover [14] operator passes on ordering and value information from the parent tours to the offspring tours. A segment of one parent's string is mapped onto a segment of the other parent's string and the remaining information is exchanged.

Alternating position crossover (AP) simply creates an offspring by selecting alternately the next element of the first parent and the next element of the second parent, omitting the elements already present in the offspring [13].

Heuristic Crossover is a crossover which emphasizes edges. These create offspring by selecting a first random city as the current city. Then edges incident to current city are chosen and some probability distribution is defined on new edges based on their costs. And then edges are selected on distribution. If uniform probability distribution is chosen, the offspring inherits about 30% of the edges of every parent, and about 40% of the edges are randomly selected [14].

## III. PROPOSED CROSSOVER

All The proposed method is a hybrid crossover operator tries to avoid the disadvantages of above crossover techniques. The main initiative of proposed crossover is to combine both the functionalities of PMA and FOMX to form a new one. By doing so, it is expected to get better results than using them individually. Joining PMX and FOMX together to get two new strings and appending them into the population results better than the existing crossover techniques. The reason why the expected results are better can be summarized as: PMX does things point-by-point, whereas FOMX applies swapping procedure within the crossover sites. So using both together can overcome the individual discrepancies and will result in an operator which works effectively including both functionalities.

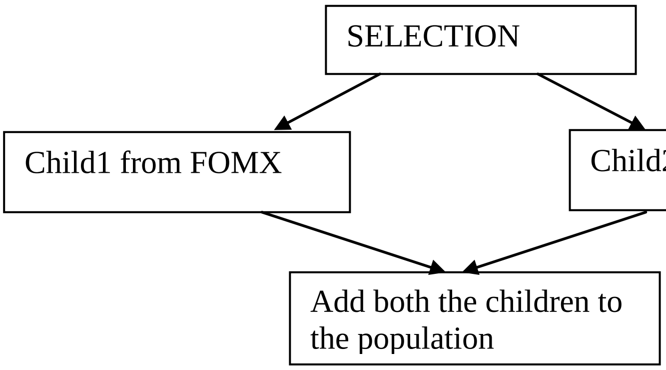


Fig. 1 Model of Proposed Crossover

For example, if two parents are selected as below for crossover:
Parent1: 1 2 3 4 5 6 7 8
Parent2: 3 7 5 1 6 8 2 4

Suppose that the first cut point is selected between the third and the fourth string element, and the second one between the sixth and seventh. After applying pmx, the following two offspring are produced
offspring1: 4 2 3 1 6 8 7 5
offspring2: 3 7 8 4 5 6 2 1
While FOMX produces following children:
Child1: 1 2 3 4 6 5 7 8
Child2: 3 7 5 1 8 6 2 4
Whereas Proposed Crossover produces following children:
Child1: 3 7 1 5 2 4 6 8
Child2: 3 7 8 4 5 6 2 1

## IV. COMPUTATIONAL EXPERIMENTS AND RESULTS

### A. *Experimental Setup*

This paper, 4 different inputs for benchmark TSP instances are used for testing. All experiments are implemented in MATLAB R2011a and problem instances are taken from TSPLIB, which can be found at http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/. The following parameters are used in this implementation:

Population size (N):100
Number of generations (ngen) :1000
Selection method: Tournament Selection (TS)
Crossover: PMX, OX, CX, Proposed Hybrid crossover
pc=0.7.
Mutation: Swap
Termination criteria: Execution stops on reaching ngen generations.
Fitness Function: Objective value of function (Minimum tour length)

*© 2014, IJARCSSE All Rights Reserved*

*B. Experimental Results*

Problems instances and results are recorded in following tables and figures:

TABLE I

COMPRISON OF CROSSOVER OPERATOR FOR GR17

| 10 Runs | FOMX | PMX | PBX | AX | MPX | GX | HX | SOX | OX | CX |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2184 | 2085 | 2090 | 2090 | 2085 | 2215 | 2175 | 2621 | 2085 | 2182 |
| 2 | 2085 | 2153 | 2085 | 2088 | 2090 | 2153 | 2321 | 2159 | 2090 | 2136 |
| 3 | 2090 | 2085 | 2103 | 2088 | 2103 | 2185 | 2100 | 2121 | 2090 | 2085 |
| 4 | 2103 | 2085 | 2090 | 2085 | 2103 | 2085 | 2090 | 2103 | 2085 | 2120 |
| 5 | 2123 | 2103 | 2085 | 2103 | 2090 | 2154 | 2103 | 2210 | 2103 | 2153 |
| 6 | 2154 | 2123 | 2085 | 2120 | 2085 | 2085 | 2132 | 2232 | 2103 | 2120 |
| 7 | 2085 | 2120 | 2085 | 2171 | 2085 | 2456 | 2134 | 2085 | 2103 | 2153 |
| 8 | 2085 | 2123 | 2085 | 2085 | 2085 | 2182 | 2103 | 2090 | 2085 | 2120 |
| 9 | 2085 | 2123 | 2103 | 2085 | 2090 | 2100 | 2085 | 2090 | 2085 | 2182 |
| 10 | 2085 | 2136 | 2090 | 2154 | 2103 | 2232 | 2089 | 2085 | 2090 | 2085 |
| BC | 2085 | 2085 | 2085 | 2085 | 2085 | 2085 | 2085 | 2085 | 2085 | 2085 |
| AC | 2107.9 | 2113.6 | 2090.1 | 2106.9 | 2091.9 | 2184.7 | 2133.2 | 2179.6 | 2091.9 | 2133.6 |
| WC | 2184 | 2153 | 2103 | 2171 | 2103 | 2456 | 2321 | 2621 | 2103 | 2182 |

The table1 shows the optimal distance generated from each crossover operator. The results have been taken for 10 runs why because in the sense for every execution it returns the optimal minimum distance. The question is which is best and very close to the actual distance. To minimize the error rate and also to find the best minimum distance of a tour 10 sample of output data are compared. Similarly the remaining tables show the optimal distance of other benchmark instances.

TABLE II

COMPRISON OF CROSSOVER OPERATOR FOR WI29

wi29

| No. of Runs | Proposed | PMX | PBX | OBX | AX | MPX | GX | HX | UX | SOX | OX | CX |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 34367 | 31815 | 29018 | 29918 | 31414 | 28028 | 37367 | 33085 | 34367 | 33988 | 28028 | 31815 |
| 2 | 33219 | 34112 | 29018 | 29918 | 30131 | 32657 | 40333 | 33988 | 33123 | 31414 | 29382 | 33988 |
| 3 | 33382 | 30810 | 29018 | 28572 | 33001 | 28178 | 39217 | 32657 | 32467 | 28028 | 28762 | 31516 |
| 4 | 33238 | 34515 | 28762 | 30166 | 33213 | 29018 | 43335 | 33213 | 32150 | 29918 | 29018 | 31815 |
| 5 | 31770 | 31815 | 30517 | 31004 | 32133 | 28028 | 37367 | 30131 | 36008 | 29018 | 28851 | 30102 |
| 6 | 31679 | 29303 | 29461 | 31414 | 31012 | 28851 | 31414 | 30517 | 34515 | 36008 | 29018 | 31828 |
| 7 | 36008 | 32150 | 29485 | 29018 | 29646 | 28851 | 35795 | 31815 | 32093 | 33238 | 28028 | 32467 |
| 8 | 30160 | 31815 | 31004 | 29485 | 32093 | 28851 | 40233 | 32248 | 35975 | 33219 | 28762 | 32150 |
| 9 | 28802 | 29246 | 28762 | 31451 | 32248 | 32657 | 43201 | 31455 | 40333 | 34367 | 28762 | 32117 |
| 10 | 27792 | 30713 | 28762 | 31815 | 35769 | 29018 | 33085 | 35675 | 31414 | 31815 | 28178 | 32117 |
| BC | 27792 | 29246 | 28762 | 28572 | 29646 | 28028 | 31414 | 30131 | 31414 | 28028 | 28028 | 30102 |
| AC | 32041.7 | 31629.4 | 29380.7 | 30276.1 | 32066 | 29413.7 | 38134.7 | 32478.4 | 34244.5 | 32101.3 | 28678.9 | 31991.5 |
| WC | 36008 | 34515 | 31004 | 31815 | 35769 | 32657 | 43335 | 35675 | 40333 | 36008 | 29382 | 33988 |

TABLE III

COMPRISON OF CROSSOVER OPERATOR FOR SWISS42

| No. of Runs | CX | PMX | PBX | AX | MPX | HX | SOX | OX | HYBRID |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1644 | 1738 | 1700 | 1752 | 1331 | 1755 | 1539 | 1437 | 1830 |
| 2 | 1692 | 1787 | 1699 | 1808 | 1273 | 1722 | 1689 | 1506 | 1745 |
| 3 | 1623 | 1742 | 1662 | 1754 | 1301 | 1877 | 1746 | 1440 | 1811 |
| 4 | 1880 | 1727 | 1756 | 1500 | 1317 | 1830 | 1898 | 1410 | 1632 |
| 5 | 1657 | 1733 | 1447 | 1819 | 1384 | 1965 | 1690 | 1393 | 1822 |
| 6 | 1527 | 1668 | 1730 | 1752 | 1324 | 1855 | 1589 | 1513 | 1522 |
| 7 | 1738 | 1735 | 1828 | 1807 | 1356 | 1766 | 1765 | 1484 | 1473 |
| 8 | 1640 | 1769 | 1747 | 1836 | 1388 | 1933 | 1983 | 1455 | 1356 |
| 9 | 1629 | 1683 | 1651 | 1608 | 1349 | 2088 | 1789 | 1403 | 1273 |
| 10 | 1802 | 1753 | 1517 | 1704 | 1408 | 1933 | 1687 | 1384 | 1273 |
| BC | 1527 | 1668 | 1447 | 1500 | 1273 | 1722 | 1539 | 1384 | 1273 |
| AC | 1683.2 | 1733.5 | 1673.7 | 1734 | 1343.1 | 1872.4 | 1737.5 | 1442.5 | 1573.7 |
| WC | 1880 | 1787 | 1828 | 1836 | 1408 | 2088 | 1983 | 1513 | 1830 |

*© 2014, IJARCSSE All Rights Reserved*

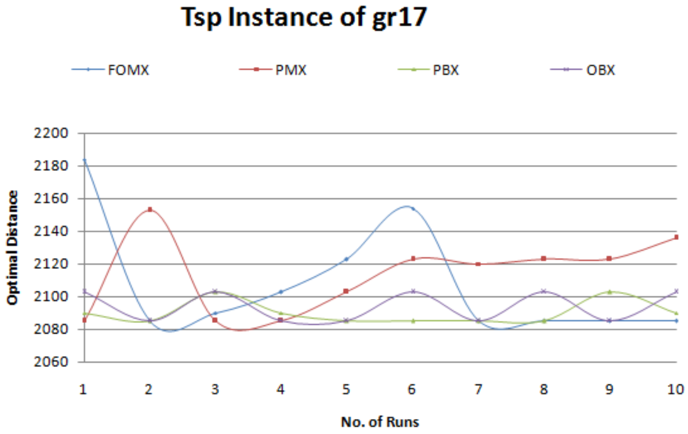


Fig. 2

All graphs figure 2-5 are performance graph for different tsp instances. When the number of cities are more the hybrid crossover operator outperforms the other operators which is shown in the performance graphs.

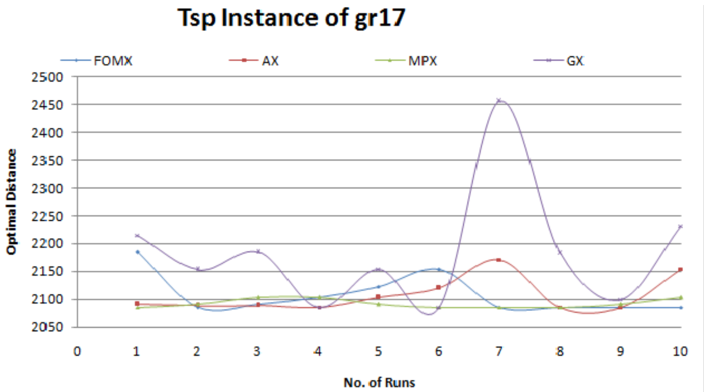


Fig. 3

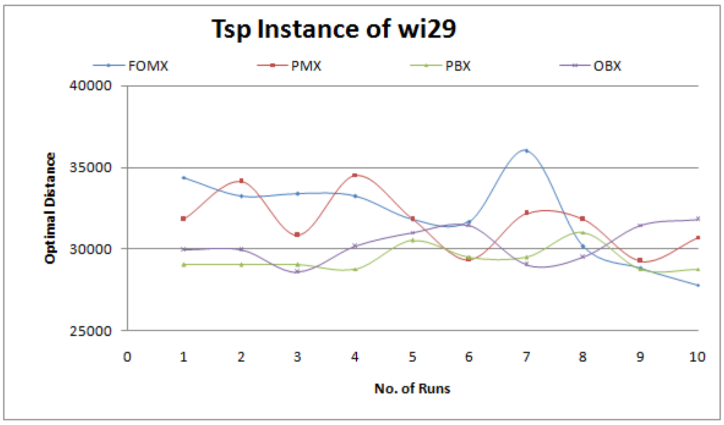


Fig. 4

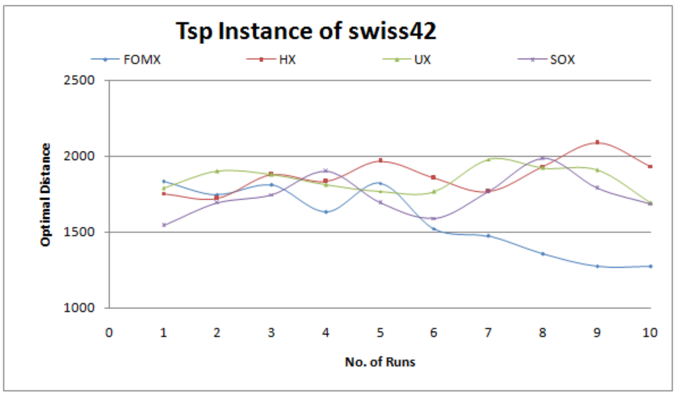


Fig. 5

## V. Conclusions

This paper compares proposed crossover operator with existing crossover operators on benchmark TSP problems like gr17, wi29, swiss42 and eil101. It was found that the proposed crossover yields better results than existing crossovers. Proposed crossover has features of PMX and FOMX both, so helpful in improving the solution quality. Also improve the performance of genetic algorithm in terms of convergence and number of iterations. Proposed crossover can be tested and implemented in different combination of selection and mutation. Hybridization of crossover has increased the existing technique in genetic algorithms and amplified the search performance.

## References

[1] Goldberg D.E. "Genetic algorithms in search, optimization, and machine learning", Addison Wesley Longman, Inc., ISBN 0-201-15767-5, 1989.

[2] Grefenstette, J. J., "Incorporating Problem Specific Knowledge into Genetic Algorithms". In L. Davis, ed., Genetic Algorithms and Simulated Annealing, 42-60, Los Altos, CA,Morgan Kaufmann, 1987.

[3] Goldberg D.E. and Lingle R. "Alleles, loci and the travelling salesman problem," Proceedings of an International Conference on Genetic Algorithms, Morgan Kauffman, 1985, pp 10-19.

[4] Davis L, "Handbook of Genetic Algorithms", New York, Van Nostrand Reinhold, 1991.

[5] Whitley Darrell, "Functions as Permutations: Regarding No Free Lunch", Walsh Analysis and Summary Statistics. PPSN, 2000, 169-178

[6] Sivanandam S.N. and Deepa S. N. "Introduction to Genetic Algorithms", Springer, ISBN 9783540731894, 2007.

[7] Eiben A.E. and Smith J.E. "Introduction to Evolutionary Computing", Springer, Heilderberg, Germany, 2003.

[8] Syswerda G., "Schedule optimization using genetic Algorithms", in Davis L. Ed. Handbook of Genetic Algorithms, Van NostrandReinhold, New York, 2008, pp.-332-349.

[9] Oliver I.M., Smith D.J. and Holland J.H. "A study of permutation crossover operators on the travelling salesman problem", In Proceedings of the Third International Conference on Genetic. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. (*references*).

[10] Kumar Rakesh, Jyotishree, 2012, "Novel Knowledge Based Tabu Crossover In Genetic Algorithms", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 8, August 2012 ISSN: 2277 128X.

[11] Muhlenbein H. Gorges-Schleuter M. Kramer O. "Evolution algorithms in Combinatorial Optimization", Parallel Computing 7, 1988, pp. 65-85.

[12] Muhlenbein H. and Kinderman J. "the Dynamics of Evolution and learning – Towards Genetic Neural Networks", in J. Pfeiffer (Ed.), Connectionism in Perspectives, 1989.

[13] Larranaga P., Kuijpers C.M.H., Poza M., y Murga R.H., "Decomposing Baysian Networks: Triangulation of the Moral Graph with genetic Algorithms" Statistics and Computing, 2009.

[14] P. Larra˜ naga, et. Al, "Genetic algorithms for the Travelling Salesman Problem: A Review of Representations and Operators", Artificial Intelligence Review, 1999.

*© 2014, IJARCSSE All Rights Reserved* 