INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

A Novel Methodology for Genetic Algorithms in Crossover Operation: Segment Replacement Opeartor

R. Lakshmi
Pondicherry University, India
K. Vivekanandan
Pondicherry Engineering College, India

Abstract:
Genetic Algorithms (GA) are robust and efficient search and optimization techniques inspired by Darwin's theory of natural evolution. GA is composed of genetic operators and genetic parameters. Most of the approaches are random in nature which reduces the performance of the simple genetic algorithm. In this paper, a novel approach for crossover operator called segment replacement Operator (SRO) is attempted. In order to evaluate the efficiency and feasibility of the proposed technique, TSP problem is chosen and a comparison between the results of this study used in GAs is made through a number of test experiments with various parameter settings. Results of this study clearly show the significant differences between the proposed operator and the other existing operator techniques

Key words: *Genetic Algorithm, Segment Replacement Algorithm, Crossover, Mutation, Performance Analysis, Partially Mapped Crossover*

1. Introduction

Genetic Algorithms are good at taking large, potentially huge search spaces and navigating them, looking for optimal combinations of things, solutions. Adaptation of natural system into artificial system is to solve any kind of problem. Holland developed this idea in his book "Adaptation in natural and artificial systems". He described how to apply the principles of natural evolution to optimization problems and built the first GAs. Holland's theory has been further developed and now GAs standup as a powerful tool for solving search and optimization problems. Genetic algorithms (GAs) represent general-purpose search and optimization technique based on evolutionary ideas of natural selection and genetics [1]. The two most commonly employed genetic search operators are crossover and mutation. Crossover produces offspring by recombining the information from two parents. Mutation prevents convergence of the population by flipping a small number of randomly selected bits to continuously introduce variation. The driving force behind GAs is the unique cooperation between selection, crossover and mutation operator. A genetic operator is a process used in GAs to maintain genetic diversity. In permutated individuals Crossover operator produces duplicate genes in their offspring. Generally the existing crossover techniques follow gene level comparison or sliding method to copy the genes into their offspring which takes more computation time and occupies a large space with respect to the algorithm. Because of their inefficiency, GA will not perform better to produce the optimal solutions. To overcome these drawbacks, this paper introduces a new technique called Segment Replacement and the performance of this technique. The rest of the paper is organized as follows. In section 2, definitions and concepts of the different crossover operators are overviewed. Section 3 discusses the proposed method. In section 4, a number of the trails widely used in performance evaluation of GA operators are defined. Finally, conclusions are discussed in section 5.

2. Crossover Operators

The crossover operator is a genetic operator that combines (mates) two chromosomes (parents) to produce a new chromosome (offspring). The idea behind the crossover operator is that the new chromosome may be better than both of the parents if it takes the best characteristics from each of the parents. Crossover occurs during evolution according to a user-definable crossover probability. For the purpose of this work, only crossover operators that operate on two parents and have no self-adaptation properties will be considered. The SRO technique is applied either in single point crossover or in two point crossover operation. The overview of these operators is given below.

2.1. Single Point Crossover (SPC)

When performing crossover, both parental chromosomes are split at a randomly determined crossover point (cp). Subsequently, a new offspring gene is created by joining the first part of the first parent with the second part of the second parent **[5, 3].** A single crossover point on both parents' strings is selected. All data beyond that point in either string is swapped between the two parent chromosomes.

2.2. Two point Crossover (TPC)

Apart from single point crossover, many different crossover operators have been introduced, often involving more than one cut point. It should be noted that adding further crossover points reduces the performance of the GA. The problem with adding an additional crossover point is that the building blocks are more likely to be disrupted. However, an advantage of having more crossover points is that the problem space may be searched more thoroughly. In two-point crossover (TPC), two crossover points are chosen randomly and the genes between these points are exchanged between two mated parents [6, 4]. Redundancy is ignorable when we apply a crossover operator to binary coded information, but it is not so in real coded representation in spite it gives invalid strings. There is no faster method to correct an invalid string to valid string, the following proposed new technique works faster in crossover operation and significantly improve the performance of the simple GA with respect to speed and optimal solutions. The working principle of the SRO technique of TSP problem is discussed in the following section.

3. Segment Replacement Operator

Genetic algorithm first creates an initial population. The fitness of each individual is computed using its fitness function. The highly fit individuals are reproduced in the next generation by means of selection operator. The selection pressure, determines the number of individuals is to be selected for the next generation. Then apply a crossover operator to mate their children. The crossover operation produces redundancy in new strings; a special mechanism is required to repair those individuals. This research provides such mechanism called SRO. The algorithm for the SRO is given below

```
Loci : = position; cs := crossover sie;
Ls := left segment; rs := right segment;
Nch := new child; cp1 := crossover point1;
Cp2 := crossover point2;
Get child(ch1, ch2, ch3…chn)
For all child
Check duplicates(chi);
If(gene_duplicate_loci < cp1)
Call pmx(ls)
Nchi = copy(ls, cs, rs);
Else
Call pmx(rs)
Nchi = copy(ls, cs, rs)
Return(nchi)
```

Figure 1: Pseudo Code of SRO

The GA approach to TSP problem has to be mapped effectively in terms of chromosome representation, determining the objective function, selection of reproduction and recombinant operators respectively [18, 19].

Each gene in a chromosome is a city. The tour consists of 17 numbers of cities. This dataset has been taken from the TSPLIB which is a standard library for tsp problems provide different tsp instances. The execution of two point crossover operation is shown in the above figure2. The figure2 also shows the concept of SRO after performing tpc operation. The working principle of SRO is shown in the figure3.

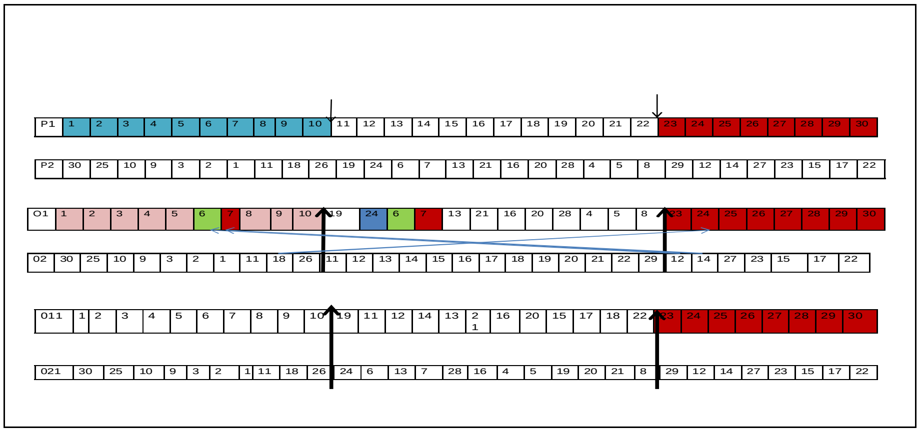

Figure 2: SRO in TPC Operation

Working Principle of SRA

Cp1 = 11; cp2 = 22;
for all cutlength
after '2' point xo, duplicate_gene_loci = 24;
if (24 < 11) then copy right segment
else call pmx to change city no. 24^{th} by city no. 11
copy (ls,cs,rs);
O1: 0 1 2 3 4 5 6 7 8 9 10| **19 24 6 7 13 21 16 20 28 4 5 8**|23 11 25 26 27 28 29 30

Figure 3: Working model of SRO

In crossover operation the two cut points are chosen randomly, the position of the first cut point is at 11. The position of second cut point is at 22. Swap the genes from cut point1 to cut point2. After exchanging the genes check for duplicate entries in offspring. If duplicate gene exists, the loci of the duplicate gene are compared to the cut points. If the loci value (not the allel) is greater than the cut point 2 then no need to make any changes in genes which is on the left side of the crossover point1, just copy the entire left segment into the offspring1. Call the pmx operator is another crossover operator which removes duplicate city in a tour which is located on the right side of the cut point2. After removing the duplicate cities the invalid tour becomes a valid tour and the same can be copied into their children for the next generation. This process is continued till all the duplicates are removed. To remove duplicates, the existing techniques do >n(n-1) comparisons, but the proposed technique does cut length, the number of comparisons which means very few comparisons are required to build offspring. This way the SRO technically and drastically increases the speed of GA to a maximum extent.

4. Experimental Results

The figure4 shows an example for the dataset gr17 obtained from TSPLIB benchmark. But in experiment, the dataset nu734 is used for the same tsplib database.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	633	257	91	412	150	80	134	259	505	353	324	70	211	268	246	121
1	633	0	390	661	227	488	572	530	555	289	282	638	567	466	420	745	518
2	257	390	0	228	169	112	196	154	372	262	110	437	191	74	53	472	142
3	91	661	228	0	383	120	77	105	175	476	324	240	27	182	239	237	84
4	412	227	169	383	0	267	351	309	338	196	61	421	346	243	199	528	297
5	150	488	112	120	267	0	63	34	264	360	208	329	83	105	123	364	35
6	80	572	196	77	351	63	0	29	232	444	292	297	47	150	207	332	29
7	134	530	154	105	309	34	29	0	249	402	250	314	68	108	165	349	36
8	259	555	372	175	338	264	232	249	0	495	352	95	189	326	383	202	236
9	505	289	262	476	196	360	444	402	495	0	154	578	439	336	240	685	390
10	353	282	110	324	61	208	292	250	352	154	0	435	287	184	140	542	238
11	324	638	437	240	421	329	297	314	95	578	435	0	254	391	448	157	301
12	70	567	191	27	346	83	47	68	189	439	287	254	0	145	202	289	55
13	211	466	74	182	243	105	150	108	326	336	184	391	145	0	57	426	96
14	268	420	53	239	199	123	207	165	383	240	140	448	202	57	0	483	153
15	246	745	472	237	528	364	332	349	202	685	542	157	289	426	483	0	336
16	121	518	142	84	297	35	29	36	236	390	238	301	55	96	153	336	0

Figure 4: Intercity Distance Table for Tsp Instance: gr17 from TSPLIB

Experiment No.	**GA with PMX**		**GA with SRO and PMX**	
	No. of generations to Converge	**Time (hours)**	**No. of Generations to Converge**	**Time (hours)**
1	9989	4.10	6567	1.30
2	9969	4.00	6421	1.22
3	9944	3.98	5853	1.12
4	9925	3.50	7001	1.55
5	9921	3.25	6978	1.47
6	9919	3.27	5421	1.0
7	9878	3.16	4432	.58
8	9578	3.15	4536	2.10
9	9424	3.06	4229	1.30
10	9132	3.00	4290	1.30

Table 1: Experimental Results

4.1. Experimental Setup

Processor: i5, Tool: matlab 2010, Population size: 500, No of cities: 734, Selection rate: 15%, Mutation rate: 10%. Crossover Probability: 75%. The important factor obtained by the SRO technique is minimum run time requirements. It also increases the convergence speed of GA and produces minimum optimal length of the tour. The performance analysis is made on number of generations to converge and the time taken for execution. The results have been taken for 10 runs in order to find the best optimal length of the tour, which is very close to the actual distance of that tour.

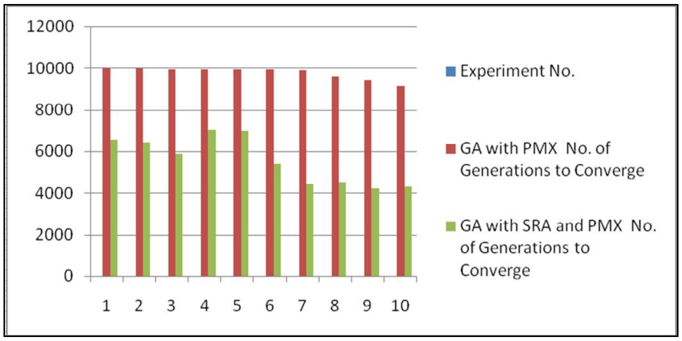

Figure 5: Performance Analysis Graph1

The analysis has been made between ga with pmx algorithm and ga with SRO and pmx [8] algorithm. The graph1 takes the number of generations as parameters and compared. The graph2 compares the differences in run time in hours. From the performance graphs, it is clearly proved that the proposed technique SRO in genetic algorithm outperforms the pmx ga in terms of convergence speed and the quality of solutions. The optimal path produced in a lesser time by the proposed technique is nearing the actual distance calculated, so the error rate is minimized. The error rate is calculated from the following formula

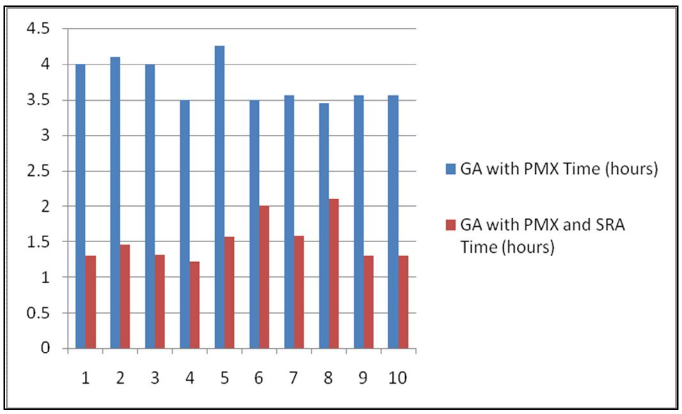

Figure 6: Performance Analysis Graph 2

$$\text{Error} = \frac{\text{Fitness} - \text{Optimal Fitness}}{\text{Optimal Fitness}}$$

$$\text{ER} = \text{Error} * 100$$

5. Conclusion

In this paper, a new methodology to build offspring for the next generation is designed and developed so called a Segment Replacement Operator (SRO) which faster the permutated GA Process. To prove the benefits of SRO and its efficacy in solving tsp is solved and the results are proven satisfactory in terms of ER and the optimal distance obtained. The experimental results are proven good when compared to the simple GA with PMX.

6. References

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. (references)
2. J. Clerk Maxwell, a Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73 S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
3. K. Elissa, “Title of paper if known,” unpublished.
4. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev. in press.
5. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

6. M. Young, the Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.
7. Oliver Kramer, Patrick Koch, "Self-Adaptive Partially Mapped Crossover", proceedings of Genetic and Evolution Computation Conference, pp. 593 – 697, ACM, July 2007.
8. R. Lakshmi, Dr. K. Vivekanandan, "An Analysis Of Recombination Operator In Genetic Algorithms", proceedings of ICOAC13 (IEEE)
9. Arthur L. Corcoran, Roger L. Wainwright, "Reducing disruption of superior building blocks in genetic algorithms", Proceedings of the, ACM SIGAPP Symposium on Applied Computing February, 2003.
10. Davis L. "Handbook of Genetic Algorithms" Van Nostrand Reinhold, 1991.
11. Ayed A. Salman, Kishan Mehrotra, and Chilukuri K. Mohan, "Adaptive linkage crossover evolutionary computation", Evolutionary computation, Vol.8 1(3), September 2000.
12. Dr.Sabry M. Abdel-Moetty "Enhanced Traveling Salesman Problem Solving using Genetic Algorithm Technique with modified Sequential Constructive Crossover Operator", IJCSNS International Journal of Computer Science and Network Security, VOL.12 No.6, June 2012
13. Chen, Y.-p., & Goldberg, D. E., "An analysis of a reordering operator with tournament selection on a GA-hard problem", Lecture Notes in Computer Science (LNCS), 2723, pp. 825–836, 2009.
14. Chen, Y.-p., Peng, W.-C., & Jian, M.-c, "Particle swarm optimization with recombination and dynamic linkage discovery", IEEE Transactions on Systems, Vol. 37(6): pp.1460–1470, 2012
15. European Graduate Student, "Workshop on Evolutionary Computation" 2012
16. S.Siva Sathya, S.Kuppuswami, Department of Computer Science, Pondicherry University, "Gene Silencing for Course Time-Tabling with Genetic Algorithm".
17. Simões, Ernesto Costa, 2003, "Improving the Genetic Algorithm‟s Performance when Using Transformation", in Proc. of the Sixth International Conference on Neural Networks and Genetic Algorithms (ICANNGA'03), Springer, pp. 175-181, April-2003
18. R. Lakshmi, Dr. K. Vivekanandhan, "Interference Induced Silencing In Travelling Salesperson Problem Using Linkage Learning Genetic Algorithm",IJESR, Volume.3,Issue-3, ISSN 2277-2685, March 2013.
19. R. Lakshmi, K. Vivekanandan, "Performance Analysis Of Linkage Learning Techniques In Genetic Algorithms", IJRET, Volume 02, issue: 12 EISSN: 2319-1163 | PISSN: 2321-7308,Dec 2013