Volume 2, Issue 8, August 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Novel Knowledge Based Tabu Crossover In Genetic Algorithms

Rakesh Kumar
Kurukshtera University, Kurukshetra
Haryana, India

Jyotishree
Guru Nanak Girls College, Yamunanagar
Haryana, India

Abstract— Genetic algorithms are optimisation algorithms and mimic the natural process of evolution. Important operators used in genetic algorithms are selection, crossover and mutation. Selection operator is used to select the individuals from a population to create a mating pool which will participate in reproduction process. Crossover and mutation operators are used to introduce diversity in the population. This paper studies different crossover operators and then proposes a novel crossover operator that incorporates knowledge based on existing population and uses the principle of Tabu search. Performance of the proposed tabu crossover is compared with existing PMX operator in genetic algorithm using MATLAB code.

Keywords— crossover, genetic algorithm, hybridisation, memetic algorithm, tabu list,

I. INTRODUCTION

Genetic Algorithms are adaptive heuristic search algorithms based on evolutionary ideas of natural selection and genetics [1]. The basic technique of genetic algorithm is to simulate processes in natural systems necessary for evolution. They are general purpose search algorithms that use principles inspired by natural population genetics to evolve solutions to problems. They follow the biological principle – survival of the fittest [2,3] which leads to better adaptation of species to their environment. In natural systems, crossing-over is a complex process that occurs between pairs of chromosomes. Two chromosomes are physically aligned, breakage occurs at one or more corresponding locations on each chromosome, and homologous chromosome fragments are exchanged before the breaks are repaired. This results in a recombination of genetic material that contributes to variability in the population. In genetic algorithms, crossover operator exchanges substrings between chromosomes represented as linear strings of symbols[4]. Simplest crossover may be exchanging genetic material of two strings with respect to single crossover point. Crossover can be quite complicated and depends mainly on the encoding of chromosomes. Specific crossover made for a specific problem can improve performance of the genetic algorithm. Crossover combines parental solutions to form offspring with a hope to produce better solutions. Crossover operators are critical in ensuring good mixing of building blocks [5].The main focus of this paper is to study different crossover operators and then introduce a new crossover operator that augments certain knowledge component in crossover operation for better results.

II. TYPES OF CROSSOVER

A. Single Point Crossover

Single point crossover is the pioneer crossover technique used in the past [3,6]. In this crossover, a single crossover point on both parent chromosomes is selected by choosing a random number between (1, lc-1) where lc is length of chromosome. Both the parent chromosomes are split at the crossover point chosen and all data beyond that point in either chromosome is swapped between the two parent chromosomes [7,8].

B. N-point crossover

This operator was first implemented by De Jong in 1975. It is generalized form of single-point crossover differing in number of crossover points [6]. For two point crossover, the value of N is 2. The value of N may vary from 1 to N-1. The basic principle of crossover process is same as that of one point crossover i.e. to exchange genetic material of the two parents beyond the crossover points.

C. Uniform crossover.

In contrast to previous crossover operators, Uniform crossover operator does not divide the parent chromosome into segments for recombination. Rather, it treats each gene of the chromosome independently to choose for the offspring. In Uniform crossover, number of crossover points is not fixed initially. It recombines genes of parent chromosomes on the basis of crossover mask. It selects x number of crossover points in the chromosome where the value of x is a random value less than the length of the chromosome. Crossover mask is generated according to this random value. In this crossover, each gene in the offspring is created by copying the corresponding gene from one of the parents. The selection of the corresponding parent is undertaken via a randomly generated crossover mask [7,9,10]. At each index, the offspring gene is taken from the first parent if there is a 1 in the mask at this index, and if there is a 0 in the mask at this index, the gene is taken from the second parent.

D. Partially Matched Crossover (PMX)

Partially Matched or Mapped Crossover (PMX) is the most widely used crossover operator for chromosomes having permutation encoding. It was proposed by Goldberg and Lingle [11] for Travelling Salesman Problem. In Partially Matched Crossover, two chromosomes are aligned, and two crossover points are selected randomly. The two crossover points give a matching selection, called swab, which is used to affect a cross through position-by-position exchange operations. [1,7,8].

E. Order Crossover (OX)

It is also used for chromosomes with permutation encoding and was proposed by Davis [12]. The order crossover begins in a manner similar to PMX by choosing two crossover points. But instead of using point-by-point exchanges as PMX does, order crossover applies sliding motion to fill the left out holes by transferring the mapped positions. It copies the sublist of permutation elements between the crossover points from the cut string directly to the offspring, placing them in the same absolute position [7,8].

F. Cycle Crossover

Cycle crossover is used for chromosomes with permutation encoding. Cycle crossover performs recombination under the constraint that each gene comes from the parent or the other [13]. The basic principle behind cycle crossover is that each allele comes from one parent together with its position. It divides the elements into cycles. A cycle is a subset of elements that has the property that each element always occurs paired with another element of the same cycle when the two parents are aligned. Cycle Crossover occurs by picking some cycles from one parent and the remaining cycles from the alternate parent. All the elements in the offspring occupy the same positions in one of the two parents. First a cycle of alleles from parent 1 is created. Then the alleles of the cycle are put in child 1. Other cycle is taken from parent 2 and the process is repeated [1,7].

G. Edge Crossover

Edge crossover is based on the idea that an offspring should be created using only edges present in one or more parent [8,14]. Edge table is created which for each element lists the other elements linked to it in the two parents.'+' sign in edge table indicates the presence of edge in both parents.

H. Arithmetic Crossover

In case of real-value encoding, arithmetic crossover is used. Arithmetic crossover operator defines a linear combination of two chromosomes [15]. Two chromosomes are selected randomly for crossover and produce two offsprings which are linear combination of their parents as per the follwing computation:

$$C_{igen+1} = a.C_{igen} + (1-a).C_{jgen}$$
$$C_{jgen+1} = a.C_{jgen} + (1-a).C_{igen}$$

where C_{gen} an individual from the parent generation , C_{gen+1} an individual from child generation , a is the weight which governs dominant individual in reproduction and it is between 0 and 1.

I. Multiparent Crossover

Mutiparent crossover does not exist in nature. In this case, number of parents is more than teo and this scheme tests the cases which do not exist in nature. Technically, multiparent crossover will amplify the effect of recombination. Multiparent crossover operators are categorised on the basis of allele frequencies or on segmentation or numerical operations on real valued alleles [16,17,18].

III. HYBRIDISATION IN GENETIC ALGORITHMS & MEMETIC ALGORITHMS

Genetic algorithm is population based global search technique. Genetic algorithms blindly search through the state space having no knowledge reference by exploiting only the coding and objective function in each generation of population. Their indifference towards problem specific information serves as a blessing as well as a curse [1]. Incorporating problem specific information in genetic algorithm at any level of genetic operation would form a hybrid genetic algorithm. One such technique of hybridisation of knowledge and global genetic algorithm is memetic algorithms. Memetic algorithms are evolutionary algorithms that apply separate local search process to refine individuals. They are inspired by richard dawkins's concept of meme [19] which represents a unit of cultural evolution that can exhibit local refinement. Memetic algorithms are also known as hybrid evolutionary algorithms [20]. In simple words, memetic algorithms can be defined as genetic algorithms that include non-genetic local search to improve genotypes. Memetic algorithms can blend the functioning of genetic algorithms with several heuristic search techniques like simulated annealing, tabu search etc.

IV. TABU SEARCH AND TABU LIST

Tabu search follows three main strategies [21] - Forbidding strategy that controls what enters the tabu list, Freeing strategy that controls what exits the tabu list and when, Short-term strategy that manages interplay between the forbidding strategy and freeing strategy to select trial solutions. A chief way to exploit memory in tabu search is to classify a subset of the moves in a neighborhood as forbidden or tabu [22]. A tabu list records forbidden moves, which are referred to as tabu moves . It can be applied to both discrete and continuous solution spaces. It is applied for larger and more difficult problems like scheduling, quadratic assignment and vehicle routing. Tabu search obtains solutions that rival and often surpass the best solutions previously found by other approaches [22].

V. PROPOSED TABU CROSSOVER

It has been observed that crossover operator largely depends on the type of encoding used. In case of permutation encoding , large number of crossover operators have been portrayed. Still there is a scope of innovation and improvement in terms of performance. Permutation encoding is applied mostly in ordered applications like TSP which require large number of computations. So, one needs to devise the crossover operator that is better in terms of performance and results in less number of iterations and eliminates redundant solutions. In this paper, a new crossover operator is proposed which incorporates certain knowledge component during crossover operation to generate better offsprings using the principle of tabu search. The proposed crossover operator considers two types of lists of pair of genes. The first list includes the best of gene pairs from both the parents that are desirable and that can be retained in the offspring chromosomes. The second list consists of pair of genes that are most undesirable in terms of fitness from both the parents and should be avoided in the offspring chromosomes. The second list works like tabu lists formed in case of tabu search. The proposed crossover operator includes the few pairs representing best gene pairs from the first list and avoids the worst gene pairs from the second list while forming the offspring. While doing so, if any gene position does not get filled up, then it is filled according to the order of gene in the parent chromosome. The proposed crossover operator has the benefit of including the best ones in search while avoiding the worst ones. It can be shown as:

Parent A	4	9	2	6	5	3	8	10	1	7
Parent B	5	3	1	2	4	8	6	9	7	10

List 1 (best gene pairs) 8-10, 3-1
List 2 (worst gene pairs) 3-8, 6-9 (Tabu List)

Child A	5	3	1	2	4	6	8	10	9	7
Child B	4	3	1	9	2	6	5	7	8	10

It is clear from this example that the two best gene pairs are retained in both the child chromosomes. The gene pairs in tabu list are avoided in the child chromosomes. This would lead to better results in terms of fitness value.

VI. IMPLEMENTATION & OBSERVATION

In this paper, genetic algorithm is developed using MATLAB code. The test problem considered is the Travelling Salesman Problem (TSP). The problem is to find the shortest tour or Hamiltonian path through a set of N vertices so that each vertex is visited exactly once [23]. The code considers the benchmark TSP problem namely Eil51 as the test problem. The code uses the initial population depending on problem size, same crossover and mutation probability. The code checks the performance of genetic algorithm by using PMX crossover operator first and then using proposed crossover operator.

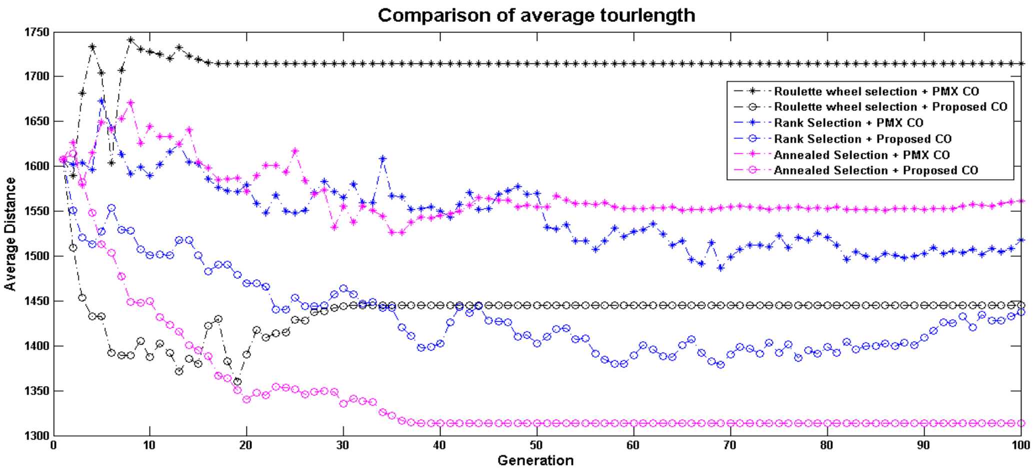

Fig.1 Comparison of Average Fitness

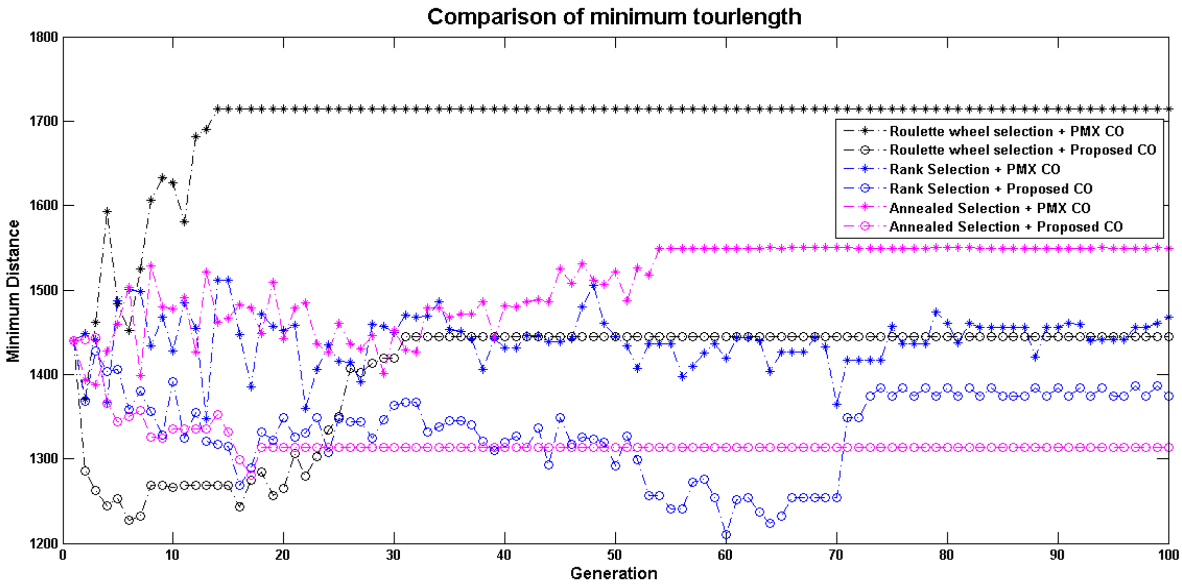

Fig.2 Comparison of Minimum Fitness

Observations were made for various runs of the code and it was found that among the three selection [24] approaches considered, the annealed selection is more promising. The results improved more in case of proposed crossover operator. It is very much clear from the graphs obtained for benchmark TSP problem that proposed crossover gives better results than PMX crossover operator. There is a large difference between the performance of genetic algorithm in the two cases of crossover operators.

VII. CONCLUSION

The paper has compared two crossover operators namely Partially matched crossover (PMX) and proposed yabu crossover on the benchmark TSP Eil51 problem. It was found that the proposed crossover yielded better results than the PMX. The proposed tabu crossover operator uses the knowledge concept along with the tabu list. It can be stated as knowledge based operator. Proposed tabu crossover can prove to be better for different applications also. Proposed tabu crossover retains the best combination of alleles and avoids the introduction of worst combination of alleles in the offsprings. This concept would surely increase the average fitness of the population and would improve the performance of genetic algorithms in terms of convergence and number of iterations. Proposed crossover can be tested and implemented in different combination of selection and mutation in future to substantiate its performance. Hybridisation of crossover and knowledge has increased the existing technique of genetic algorithms and amplified the search performance of the algorithm.

REFERENCES

[1] D.E. Goldberg, Genetic algorithms in search, optimisation, and machine learning, Addison Wesley Longman, Inc., ISBN 0-201-15767-5, 1989.
[2] D. Fogel, Evolutionary Computation, IEEE Press, 1995.
[3] J.Holland, Adaptation in natural and artificial systems, University of Michigan Press, Ann Arbor, 1975.
[4] C. Ryan, Automatic re-engineering of software using genetic programming, Genetic Programming Series. Kluwer Academic Publishers, ISBN 0-7923-8653-1, 2000.
[5] T. Back, D.B. Fogel, and Z. Michalewicz (Eds), Evolutionary Computation: Basic Algorithms and Operators, Vol. 1 and Vol. 2, Institute of Physics Publishing, Philadelphia, PA, 2000.
[6] K.A. De Jong, An Analysis of the behavior of a class of genetic adaptive systems (Doctoral dissertation, University of Michigan), Dissertation Abstracts International 36(10), 5140B University Microfilms No. 76/9381, 1975.
[7] S.N. Sivanandam and S. N. Deepa. Introduction to Genetic Algorithms, Springer, ISBN 9783540731894, 2007.
[8] A.E. Eiben and J.E. Smith, Introduction to Evolutionary Computing, Springer, Heilderberg, Germany, 2003.
[9] D.H. Ackley, A connectionist machine for genetic hillclimbing. Boston: Kluwer Academic Publishers, 1987.
[10] G. Syswerda, Uniform Crossover in Genetic Algorithms. In Proceedings of ICGA, 1989, pp 2-9.
[11] D.E.Goldberg. and R.Lingle, Alleles, loci and the travelling salesman problem, In Proceedings of an International Conference on Genetic Algorithms, Morgan Kauffman, 1985, pp 10-19.
[12] L. Davis, Handbook of Genetic Algorithms New York, Van Nostrand Reinhold, 1991.

[13] I.M.Oliver, D.J. Smith and J.H. Holland, A study of permutation crossover operators on the travelling salesman problem, In Proceedings of the Third International Conference on Genetic Algorithms, London, Lawrence Eribaum Associates, 1987.

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

[15] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag, 2nd edition, 1994.

[16] H. Muhlenbein, Parallel genetic algorithms, population genetics and combinatorial optimisation, In Proceedings of the Third International conference on Genetic Algorithms, 1989, pp 416-421.

[17] A.E.Eiben, P.E.Raue and Z.Ruttkay, Genetic algorithms with multi-parent recombination, In Davidor et al. (ed.) Proceedings of the 3rd Conference on Parallel Problem Solving from Nature, No. 866 in LNCS Springer Berlin Heidelberg New York, 1994, pp 78-87.

[18] S.Tsutsui, Multi-parent recombination in genetic algorithms with search space boundary extension by mirroring, In Eiben et al. (Ed.) Proceedings of the 5th Conference on Parallel Problem Solving from Nature, 1998, pp 428-437

[19] R.Dawkins,The Selfish Gene, Oxford University Press, Oxford, 1976.

[20] P. Moscato,P. and C. Cotta, A gentle introduction to memetic algorithms, Handbook of metaheuristics, 2003, pp 105-144.

[21] D.T. Pham, D. Karaboga, Intelligent Optimisation Techniques – Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks. Springer-Verlag., 2000

[22] F. Glover, J.P. Kelly, M. Laguna, Genetic Algorithms and Tabu Search: Hybrids for Optimization. Computers and Operations Research. Vol. 22, No. 1, pp. 111 – 134, 1995.

[23] M.Perling , T.S.Gene, A Relational-Functional Genetic Algorithm for the Travelling Salesman Problem", Technical Report, Universitat Kaiserslautern, ISSN 09460071, 1997.

[24] Rakesh Kumar, Jyotishree, Blending roulette wheel selection & rank selection in genetic algorithms, Proceedings of 3rd International conference on machine learning and computing, V4, IEEE catalog number CFP1127J-PRT, ISBN 978-1-4244-9252-7, pp 197-202, 2011

