Genetic Algorithm – A Sensible Evolutionary Optimization Technique

P Muthulakshmi, E Aarthi

Abstract: The study presents a pragmatic outlook of genetic algorithm. Many biological algorithms are inspired for their ability to evolve towards best solutions and of all; genetic algorithm is widely accepted as they well suit evolutionary computing models. Genetic algorithm could generate optimal solutions on random as well as deterministic problems. Genetic algorithm is a mathematical approach to imitate the processes studied in natural evolution. The methodology of genetic algorithm is intensively experimented in order to use the power of evolution to solve optimization problems. Genetic algorithm is an adaptive heuristic search algorithm based on the evolutionary ideas of genetics and natural selection. Genetic algorithm exploits random search approach to solve optimization problems. Genetic algorithm takes benefits of historical information to direct the search into the convergence of better performance within the search space. The basic techniques of evolutionary algorithms are observed to be simulating the processes in natural systems. These techniques are aimed to carry effective population to the next generation and ensure the survival of the fittest. Nature supports the domination of stronger over the weaker ones in any kind. In this study, we proposed the arithmetic views of the behavior and operators of genetic algorithm that support the evolution of feasible solutions to optimized solutions.

Index Terms: Genetic Algorithm, fitness function, cross over, mutation, optimization technique

I. INTRODUCTION

Genetic Algorithm (GA) is a nature inspired computational metaheuristic model that finds part in most of the problem solving strategies in order to get optimal solutions. This paper discusses the behavior of GA, which supports quick convergence of feasible solutions into optimal solutions.These optimal solutions promise to increase the productivity of any system. This biological evolution method is found useful for solving both constrained and unconstrained optimization problems. In addition, GA techniques could be applied on search problems to find approximate solutions through the principles of biological evolutions [1].

In GA process, the population of candidate solutions is allowed to evolve towards better solutions. The candidate solutions can be a randomly generated population or it can be arrived out of heuristic methods. In the context of GA, the properties of a candidate solution are represented as chromosomes /genotype. Further, these solutions are allowed to get into transitions to form as best candidates. The remaining candidates of the population would be allowed through transitions to take up a chance to turn into best solutions.

P Muthulakshmi, Department of Computer Science, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India-603 203.
E Aarthi , Department of Computer Science, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India-603 203.

The process of feeding the candidate solutions for transitions to get transformed into to better solutions is called iterations. Generally, the iterations will be restricted to one or more of the following cases, (a) certain period of time (b) certain number of times (c) until all the candidate solutions transformed into be better solutions, which is called as convergence point. This time taken for reaching the convergence point influences the complexity of the algorithm. The evolution starts from a population of individuals and the iterative process of evolution proceeds with new population every time called generation. In each generation, the fitness of every individual in the population is evaluated; the fitness is the value of the objective function in the optimization problem. The more fit individuals are selected from the population and are grouped as best solutions. The remaining individual's structure is changed to form a new generation. The new generation is used in the next iteration to identify the best individuals by matching the fitness value.Generally, the problem solving strategies that adopt GA use biologically inspired techniques namely; genetic inheritance, natural selection, crossover and mutation. Inheritance and selection methods involve a general process to arrive at start up solutions whereas, crossover and mutation involves recombination procedures to get better solutions. GA is popular for its ability to convert feasible solutions into optimized solutions in a short span of time, which means the convergence of the solutions happen in very less iterations. The application areas include a broad spectrum of domains like, natural sciences, finance, investment strategies, economics, social sciences, industry, management, biological sciences, bioinformatics, automotive design engineering design, evolution in device/ hardware, robotics, computer science, earth sciences, traffic-trip-shipment routing, encryption and code breaking, marketing, merchandising, telecommunications.

The rest of the paper is organized as follows, section II discusses the fundamentals of genetic algorithm, section III talks on the convergence of solutions and section IV gives the conclusion.

II. FUNDAMENTALS OF GENETIC ALGORITHM

GA simulates the survival of the fittest individuals over a sequence of generations. Each generation has a volume of population. Each individual is a search space in the population and the search space may turn into a solution. Thus, individuals are made to undergo a process of evolution. GA stands on the genetic structure called chromosome and the behavior of chromosomes within the population. The individuals in the population compete for

resource and mates. Better offspring are produced by these population as the genes from good individuals propagate throughout the evolution process and hence competing and successive generation are becoming more suitable to their living environments [2] [3] [4].

A. Chromosome Encoding

A chromosome is a set of parameters defined with functional capabilities that represents solutions of a problem. For example, in resource allocation methods, a pair task/job and resource defines the chromosome, which may have start time, end time, wait time, execution time, resource availability, minimum execution time on the available resources, file size, resource capacity as their functional entities. The parameters play a vital role to determine the efficiency of the computing system. The chromosome may have a string of execution sequences that composed of pairs of task/job, resource. These chromosomes are checked for their feasibility of existence based on precedence constraints, which means the successor task in a path of execution should not begin to execute before its parents finish their execution. Genetic algorithm encodes the solutions of the problem as chromosomes. The functional parameters must be identified as variables to receive inputs and these variables must be encoded into strings. The string structure may vary with respect to the problem and so as its input variables. The length of the string is usually determined according to the accuracy of the desired solution. In order to get better implementation, various encoding methods can be used in a particular problem. Binary encoding, decimal or real number encoding, integer or literal permutation encoding and value encoding are the popularly known encoding methods. Binary encoding uses zeros and ones and hence, the string may consist of combinations of 1's and 0's. It is widely adopted by research methods, as it is easy to implement bit wise operations. However, it might not be suitable for all problems.Decimal or real number encoding is a proven solution for function optimization and constrained optimization problems. Integer or literal permutation encoding is used for combinational optimization problems as it searches for best permutation or combination of solutions.Complicated values of solution that involve real numbers, alphabets, combination of real numbers and alphabets use value encoding. In value encoding, every chromosome is a string of alphanumeric values. From the observations, it is found that value encoding method suits well for most of the real world applications.

B. Fitness Function

Fitness function is the scaling factor used to measure the quality of individuals in the population. Generally, the fitness function would be defined to encourage the formation of solutions that optimizes the objective function. Fitness function helps to identify the best individuals/solutions. A good fitness function results in solutions that might be very close to the desired or amongst set of solutions.The fitness function quantifies the optimality of a solution so that a particular solution may be ranked against all the other solutions. It depicts the closeness of a given solution to the desired result. Figure 1 shows the functioning of Genetic Algorithm
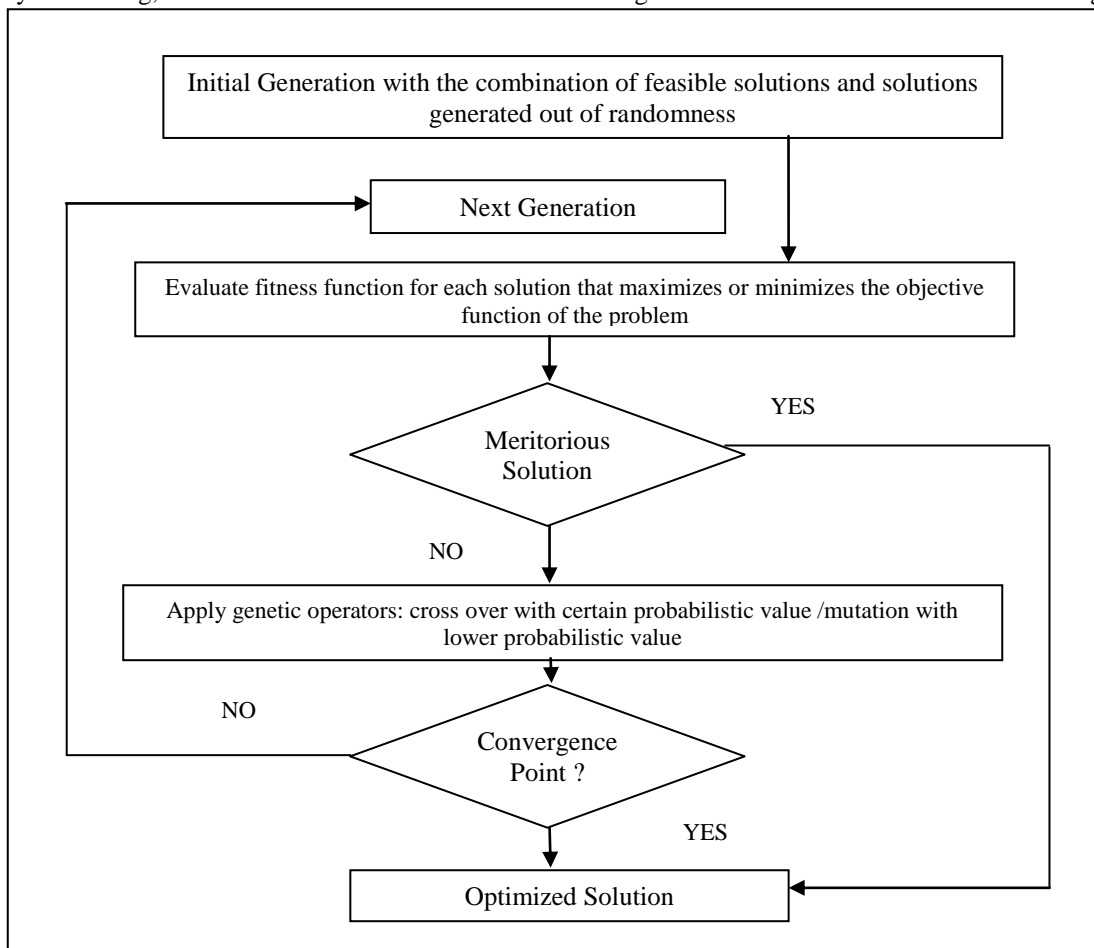
Figure 1. Functioning of Genetic Algorithm

III. THE SELECTION PROCESS

The selection process is to retain the fittest individuals in the population from the successive generations. The selection operator helps to get more copies of better strings and thus ensures good individuals for reproduction through the process of mating. To continue the generation of new population, reproduction of the individuals in the current population is very important. The fittest individuals are identified as best individuals from the current population and are allowed for mating to produce new individuals for the next generation. Roulette wheel selection, rank selection, tournament selection, steady state and elitism are the popular selection mechanism available to choose best individuals for reproduction.

A. Roulette wheel Selection Method

In roulette wheel selection, probabilistic selection of individuals is done. The probabilistic selection is based on the portion of an individual represented on the roulette wheel. Individuals are selected with a probability that is directly proportional to their fitness values. The fitness value of an individual corresponds to a portion of roulette wheel. The selection of individuals for reproduction is similar to the process of spinning roulette wheel for probabilistic results. The roulette wheel contains segments of different sizes. The segment represents an individual and its size could be proportional to the fitness value, it means the fittest individual gets big portion in the roulette wheel whereas, the least fit have correspondingly a smallest segment. However, all individuals would get a chance with a probability that is proportional to its size. The circumference of the roulette wheel is the sum of all fitness values of the individuals.

An individual would be chosen each time to get a fixed population. In order to choose 'n' individuals, the roulette wheel would be spun for 'n' times. Better individuals will be selected more often than the poor thus fulfilling the requirements of survival of the fittest. Let f_1, $f_2,\dots,f_n$ be fitness values of individuals 1, 2,..., n. Then, the selection probability p_i for an individual 'i' is define as,

$$P_i = f_i / (\sum_{j=1}^{n} f_j) \qquad (1)$$

Roulette wheel selection method gives a chance for each individual to be selected. It avoids the problem of frequent selection of certain individuals and it would not reject individuals. Hence, an assort population is generated; also wide range of population diversity is preserved.

B. Rank Selection Method

Rank selection method assigns a numerical rank value for individuals. The rank value is determined by the fitness an individual in the population. In this selection, the individuals are sorted from best to worst according to their ranks. The individual having highest rank is the fittest to get selected for the next generation population. The rank selection method prevents least fit individuals from being selected. Sorting the individuals may consume time that would result in increased time complexity.

C. Tournament Selection Method

Tournament selection method randomly selects a set of individuals and picks out the best individual for reproduction. Tournament size is defined to be the number of individuals in the set. The tournament selection strategy uses selection pressure (the degree to which the better individuals are favored) by holding a tournament competition among 'n' individuals. The selection pressure is nothing but an extent to which the better individuals get favors. Binary tournament method is found to be the common tournament method. The winner of the tournament is observed to be the best individual and it is obvious that the winner would have the highest fitness value out of 'n' individuals. The other individuals are identified to be tournament competent. Then, the winner and competent are allowed mating. The tournament is repeatedly conducted as long as there are places for the new offsprings. An arbitrary selection procedure may be used to get various tournament sizes.

D. Steady State Selection Method

In every generation, steady state selection method allows very few individuals to create offsprings. Good individuals with high fitness are retained for reproduction and this favors the maximization problem. This method eliminates bad individuals and the new offsprings will be replaced to preserve the size of population.

E. Elitism Method

In general, elitism method is carried out after the selection methods as it retains best individuals that could give best offsprings for the next generation. When using elitism method, the quality of solutions in each generation increases over time. There may be chances to lose the best individuals due to uncertain errors in one or more operators viz., crossover, mutation or selection procedure. Elitism method can be incorporated with roulette wheel method and rank selection method. It is proven that the elitism method could improve the performance of genetic algorithm.

IV. THE GENETIC OPERATORS

After the initial generation of population, the generations evolve through genetic operators called crossover and mutation

A. Crossover

It is a recombination operator that could create new offsprings for the next generation. The operator involves few steps. During the initial step, the reproduction operator randomly selects a pair of individuals mating, then a cross site is selected at random places on the strings. Finally, the strings are swapped with respect to the cross site i.e., the first portion of the first string and the second portion of the second string will form a new offspring similarly, the first portion of the second string and the second portion of the first string would be combined to form new offspring.

There are different types of crossovers namely, single-point crossover, two-point crossover, uniform crossover, arithmetic crossover and heuristic crossover. The selection of crossover operators should ensure appropriate search in the genetic space [5].

Single-point crossover randomly selects one crossover point and interchanges the parts of the strings to obtain new offsprings.

Two-point crossover randomly selects two crossover points within an individual and interchanges the portions between the points to produce two new offsprings.

Uniform crossover allows the parent chromosomes to get mixed at the gene level rather at the segment level.

Arithmetic crossover uses the following equation in order to produce new offsprings from the parents,

$$\text{Offspring} = a * parent1 + (1-a) * parent2 \qquad (2)$$

$$\text{Offspring2} = (1-a) * parent1 + a * parent2 \qquad (3)$$

Where, 'a' is a random weight factor used to mark an influential parent.

Heuristic crossover uses the fitness values of the two parent chromosomes in order to determine the direction of the search.

$$\text{Offspring1} = \text{Best Parent} + r * (\text{Best Parent-Worst Parent}) \qquad (4)$$

$$\text{Offspring1} = \text{Best Parent} \qquad (5)$$

Where, 'r' is a random number between 0 and 1.

Crossover Rate

Cross over rate is the probability of crossover. The probability varies from 0 to 1. Crossover rate determines the ratio of the number of pairs involved in swapping the portion of strings to some fixed populations. Generally, the crossover rates range between 0.5 and 1.

B. Mutation

Mutation operator involves the process of flipping i.e., changing 0 to 1 and vice versa in such a way that it accepts a small mutation probability. The bits of the individuals are independently muted. The mutation of a bit does not affect the probability of mutation of other bits. The mutation is simply a protection mechanism against an irreversible loss of genetic material and it is used to maintain variety in the population.

Mutation Rate

Mutation rate is the probability of mutation and it is a value that determines the number of bits to be muted. Mutation probabilities are smaller in natural population. In general, the mutation rates are varying from 0.001 to 0.5.

V. CONVERGENCE OF SOLUTIONS

The genetic operators are found to follow randomness when they are applied. No mathematical proof is available for convergence of GA. GA involves generation of population. The individuals are retained in the population list based on their fitness values and the population proceeds towards the next generation. After certain generations much improvement cannot be seen in the population fitness i.e., best individuals may not be seen for subsequent populations. As the generation progresses, the population gets filled with more fit individuals and there can be small deviation of fitness value for those who are away from the best individuals of maximum fitness values. The convergence of population is achieved when the population is found with best-fit individuals after several generations [6]. It is highly appreciable if the convergence happens in very less generations. The fitness function is responsible for the quick convergence of population. Applying operators on the converged population may lead to generation of weaker solution [7] [8] [9].

VI. CONCLUSION

GA is an evolutionary approach to derive optimal solution from the existing solutions. The initial population may consist of feasible solutions from a generic or hybrid algorithm which could be used for the evolution of generations to produce optimized solutions. Randomness is found to favor most of the situations and therefore a partial population may be filled with solutions generated out of randomness. The random combinations are ensured to be valid by preserving the precedence constraints. The genetic operators are applied suitably on strings to get optimized results. After a considerable number of iterations, the results are found to converge as optimum. Generally, the fitness function is designed to validate the strings that can be considered as an individual in the population for next successive generations. Designing a fitness function plays a vital role applying Genetic Algorithm for optimization. GA helps in finding solutions instead of constructing it and it eases the implementation of further constraints on solutions. GA offers simple ways to integrate different search procedure into to one optimization process and hence results obtained through this optimization yields global optimum for each search instead of local optimal solutions.

REFERENCES

1. Kalyanmoy Deb, "Multi-objective Optimization Using Evolutionary Algorithms", John Wiley & Sons, Inc. New York, USA, 2001.
2. Mitsuo Gen and Xinjie Yu, "Introduction to Evolutionary Algorithms", Springer-Verlag London, 2010.
3. Mitsuo Gen and Runwei Cheng, "Genetic Algorithm and Engineering Optimization", Wiley Series in Engineering Design and Automation, 2000.
4. K.F. Man, K.S. Tang , S. Kwong, "Genetic algorithms: concepts and applications [in engineering design]", IEEE Transactions on Industrial Electronics, 1996.
5. Anju Bala, Aman Kumar Sharma, "A comparative study of modified crossover operators", International Conference on Image Information Processing, 2015.
6. Haldurai Lingaraj, "A Study on Genetic Algorithm and its Applications", International Journal of Computer Sciences And Engineering, Volume 4, Issue., 10, pp. 139-143, 2016.
7. Jian-Jun Hu, Yue-Hong Sun, Qing-Zhen Xu, "The Genetic Algorithm in the Test Paper Generation", International Conference on Web Information Systems and Mining, 2011.

8. Jia Xua , Lang Pei a , Rong-zhao Zhub, "Application of a Genetic Algorithm with Random Crossover and Dynamic Mutation on the Travelling Salesman Problem", 8th International Congress of Information and Communication Technology, 2018.
9. Muhammad Kamal Amjad et. al., "Recent Research Trends in Genetic Algorithm Based Flexible Job Shop Scheduling Problems", Materials Science in Engineering, 2018.

AUTHORS PROFILE

Dr.P.Muthulakshmi, Associate Professor of Computer Science, SRM Institute of Science and Technology is interested in Distributed and High Performance Computing, Algorithms, Wireless Sensor Networks, Java Technologies, Spring Boot Framework

E.Aarthi, Assistant Professor of Computer Science, SRM Institute of Science and Technology is interested in Parallel Processing, Semantic Web, Social Media Analysis, Open Source Technologies